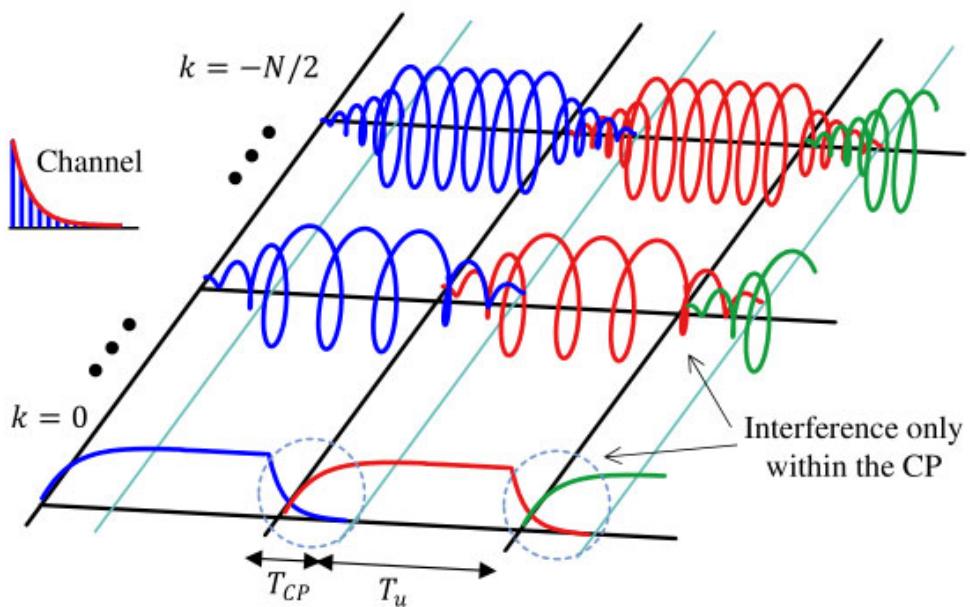


Chapter 9

Orthogonal Frequency Division Multiplexing (OFDM)



"If you want to find the secrets of the universe, think in terms of energy, frequency and vibration."

Nikola Tesla

Since around the start of the 21st century, high data rate wireless communications is often being realized through an Orthogonal Frequency Division Multiplexing (OFDM) system. OFDM is a special case of Multi-Carrier (MC) systems as opposed to a conventional Single-Carrier (SC) system we have discussed so far in the previous chapters.

OFDM is the technology behind many high speed systems such as WiFi (IEEE 802.11a, g, n, ac), WiMAX (IEEE 802.16), 4G Long Term Evolution (LTE) and 5G mobile communication systems. A close cousin, Discrete Multi-tone (DMT), is used in ADSL and powerline communication systems. As a consequence, there has been a tremendous research activity in this field since 1990s.

The concepts on which OFDM is based are so simple that reading from many resources is not required to understand the fundamentals. Furthermore, as opposed to single-carrier systems where the prevalent material everywhere can be too daunting for a non-mathematician radio enthusiast, a large number of excellent tutorials and books are available on OFDM. Nonetheless, I bring my own perspective of viewing the OFDM signals in this chapter.

Single-Carrier vs Multicarrier Systems

A few points regarding single and multicarrier systems are in order.

5 vs 1 chapters: OFDM is a multicarrier technique as opposed to a single-carrier system discussed so far. We have devoted five chapters to understanding the implementation of a single-carrier system, so why just a single chapter on OFDM? There are several reasons for this choice.

First, the material on single-carrier systems enabled us to deeply visualize how the signals behave and subsequently develop neat mental models for understanding communication signals. This is a valuable tool and is useful in any context, even when you go on to read about OFDM or Multiple Input Multiple Output (MIMO) systems not discussed in this text. So it is not really necessary to allocate one chapter to each problem arising in an OFDM system (although it can certainly be done). The tools devised before can help us get to the crux of the problem and solve it right away. For example, many of the insights from symbol timing synchronization in a single-carrier system can be brought into looking at the effect of carrier frequency offset in an OFDM system.

Second, a single-carrier system can be implemented in a relatively diverse manner. For equalization and each type of synchronization block, feedforward and feedback, data-aided/decision-directed and blind solutions are available that can be mixed and matched together. As we will later see, OFDM is implemented through a series of fast processing routines that operate in a process and dump fashion one after another. This ease of processing comes from the particular signal structure that leaves little room for a diverse mix of algorithms. Although the literature available on OFDM far exceeds that on any other technology, it

still mostly consists of modifications around that fixed signal structure and hence less versatile than single-carrier systems.

The role of single-carrier systems: As mentioned above, a number of modern high-speed wireless communication systems are based on OFDM or multicarrier technique in general. It is easy to then focus on understanding OFDM or multicarrier systems only. However, single-carrier systems are just as important and in fact, a multicarrier system can be considered as another kind of a single-carrier system in terms of an elongated symbol described later. So an understanding of single-carrier systems helps solve the issues in multicarrier context as well.

Before we start, I touch upon a bigger picture along with a brief history of OFDM.

9.1 The Big Picture

The basic multicarrier principle of segmenting one bit stream into several parallel streams and using them to simultaneously modulate independent carriers was first used in the Collins Kineplex System. It largely remained in the shadows for almost three more decades with some sporadic developments of significance. This is because an array of parallel sinusoidal generators was quite complex to build through analog modules.

Then in 1971, Weinstein and Ebert [37] showed that a Discrete Fourier Transform (DFT) – a fast version of which is FFT (Fast Fourier Transform) – can be employed to efficiently implement such a system. Later in 1980s, OFDM was proposed as the solution to the challenging problem of digital TV broadcasting through terrestrial networks in Europe in the context of a single frequency network.

Moore's law for the integrated circuits[†] in the meantime was catching up in the radio domain and by the 1980s, real-time digital signal processing advanced to a level that many of the operations performed by the analog circuits could be replaced by a mere play with numbers, thus giving birth to the concept of a Software Defined Radio (SDR). It is good to note that the SDR concept was not born out of thin air but instead was originally pushed forward in 1970s mainly by the efforts of the usual suspect – the US Department of Defense.

Interestingly, an SDR is *the natural partner* to digital communication theory in general. We know that digital communication is mostly about conveying discrete amplitudes and phases, i.e., concrete numbers, from point A to point B. It was a great mismatch then that a major portion of such number manipulation was being performed through analog circuitry and the SDR concept just brought the digital information processing back to where it belonged.

In light of the above, we can say that OFDM is the most prominent flag-bearer of the general SDR concept. This is no surprise then that the two landmark papers announcing the supremacy of OFDM and SDR, respectively, were published very close in time to each other:

[†]Gordon Moore, the co-founder of Fairchild Semiconductor and Intel, predicted in 1965 that the number of transistors in a dense integrated circuit doubles about every 18 months. This scaling down of the transistor size has spawned revolutionary growth in the world of computing, electronics and society in general in the last 50 to 60 years. In fact, if we subtract the effect of Moore's law from the past half a century, humankind has relatively little to show for its progress.

1. "Multicarrier Modulation for Data Transmission: An Idea whose Time Has Come" by J. Bingham [38], and
2. "The Software Radio" by J. Mitola [39].

While SDR was the future of both digital and analog communication in any case, OFDM played a significant role in its traction.

Now even more interestingly, there was a noteworthy development around the same time in computer peripheral communication history. While a serial interface transmits one bit at a time, a parallel port sends multiple bits of data at once through multiple parallel wires in its cable and port connector. The most prominent parallel port, commonly known as a printer port shown in Figure 9.1, was widely used in personal computers from the 1970s to 2000s. At that time, designers found it easier to add extra hardware for high rates such that the ideal aggregate data rate was the individual rate multiplied with the number of parallel links. Again, the integrated technology has progressed to a point that serial links (e.g., a USB) with fewer pins and subsequent device size reduction have become the standard and parallel port has gradually sunk into oblivion.



Figure 9.1: A parallel port (image from Wikipedia) has gradually been replaced by a serial port (USB)

This development has exactly been opposite to the wireless systems where the trend has moved from serial to parallel communication. We will learn in this chapter that OFDM is nothing but a set of several parallel waves in the air instead of one (and hence the term multicarrier). This parallel trend has infiltrated the wireless industry not only through a multicarrier system, but also from multiple antennas at the Tx and/or the Rx. Such a system is known as a Multiple Input Multiple Output (MIMO) system that has added a delightful extension to the Rx algorithm design which unfortunately is outside the scope of this text.

The main highlight of this parallel processing for fast data rates is the emergence of *process and dump* philosophy of the algorithm design, see Figure 8.64 as a reference. Whether it is the timing or frequency synchronization, phase synchronization or (most importantly) equalization, each Rx block takes as an input a set of samples and processes them in a feedforward manner. The natural casualty of such a design are recursive loops which are slowly reducing in scope from the packet based systems. Who wants a time domain equalizer to converge (or to a lesser extent, an x-locked loop to settle) while watching an HD video on the train!

Another aspect of this history appears when we consider that the rise of OFDM almost coincides with the rise of capacity approaching error correcting codes. Also known as a Forward Error Control (FEC), it is a process of detecting and correcting bit errors in digital communication systems by adding some redundancy to the Tx

bit stream in an intelligent manner. Turbo Codes took the communications world by storm (and enough suspicion) when their discovery was first published in 1993. Later on, Low Density Parity Check (LDPC) codes were rediscovered in 1996 that exhibit slightly superior performance in some respects.

Like a trick played by nature, the main feature that propels these codes closer to the channel capacity is their *feedback loop* during the decoding process: several iterations need to be performed within the decoder to *converge* towards a desirable

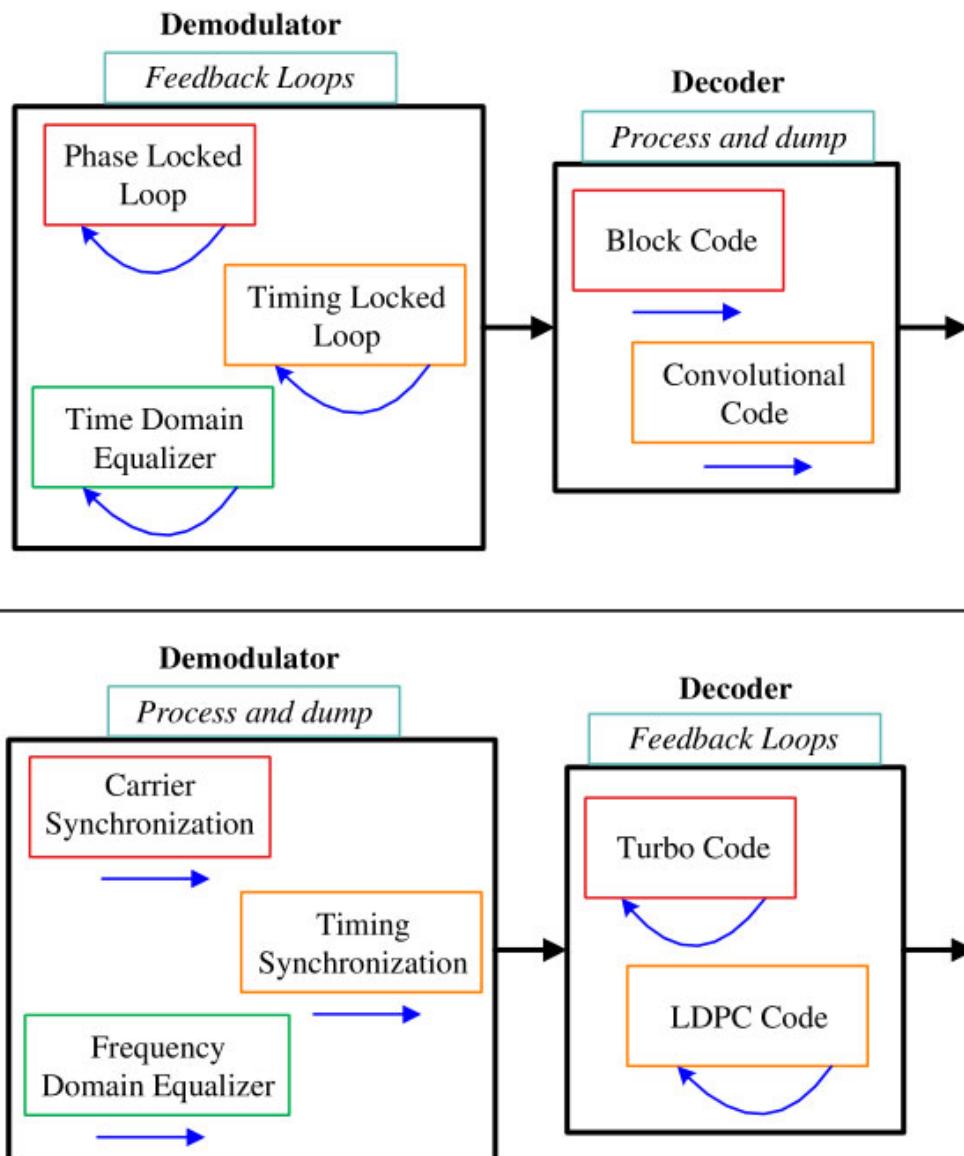


Figure 9.2: During the radio evolution, feedback loops are now less preferred in the demodulator but necessarily implemented in the iterative decoders

point on the BER/SNR curve. As illustrated in Figure 9.2, the feedback loops that were gradually being dismissed from the signal demodulation process have reappeared in the signal decoding procedures! It seems that the radio is evolving like an organism.

This evolution is also manifested in the form of a very large number of antennas in massive MIMO systems. A direct consequence of this transformation is that beam-forming – a luxury for 20th century radios – has become the defining feature of the next generation of wireless systems. This is hardly surprising since a luxury for one generation almost always becomes a necessity for the next.

With this background in place, it seems imperative to have a signal level understanding of how OFDM works.

9.2 How OFDM Works

We start this topic with an analogy that can help us better visualize the transmission and interference processes. I highly recommend reading about the wireless channel in Section 8.1.1 to fully comprehend what comes next.

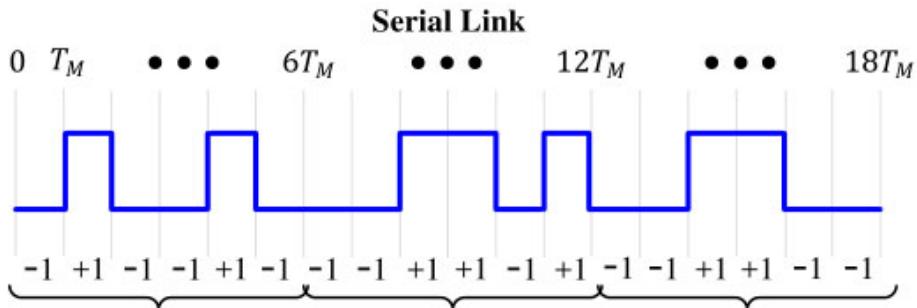
9.2.1 From the Printer Port

First, recall from Section 8.1.1 that the major problem with high speed wireless communication is that the multipath components arriving at the Rx cause different attenuations plus phase shifts and consequently constructive and destructive interference throughout the signal span. This multipath depends on the physical environment around the Tx and Rx. For low data rates, the multipath arrive very close in time to each other with respect to the symbol rate $1/T_M$. However, as we increase the data rate, each symbol starts interfering with dozens of symbols in the future, a phenomenon known as Inter-Symbol Interference (ISI).

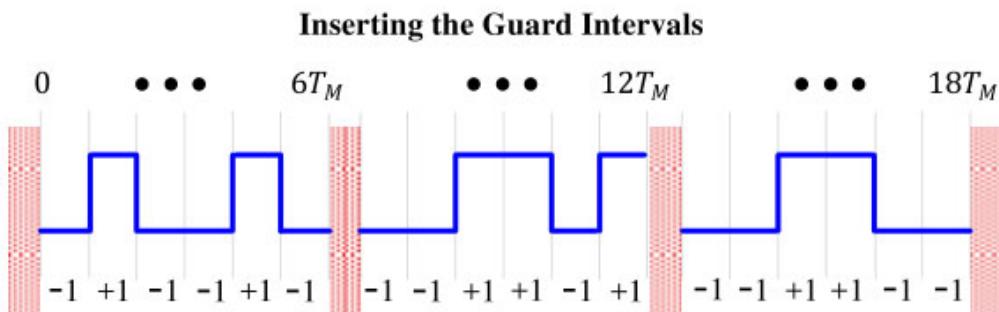
Since this physical environment cannot be changed, a safe method to avoid this long reaching effect of ISI is to reduce our symbol rates. The reduction of symbol rate is then insufficient for meeting our growing needs for faster data rates. What to do in this situation?

Assume that like a printer (parallel) port shown in Figure 9.1 before, the system designer has access to an array of parallel wires in the air that do not interfere with each other during their flight. We will see later how these parallel wires can be elegantly created. It is evident that if N such parallel wires are included in the design, the symbol rate $1/T_M$ can be simply reduced by a factor of N , i.e., the new symbol rate (actually, the block rate) can be $1/NT_M$ without compromising on the overall throughput of the system. This is shown in Figure 9.3 where the block duration $NT_M = 6T_M$ is indicated at the bottom.

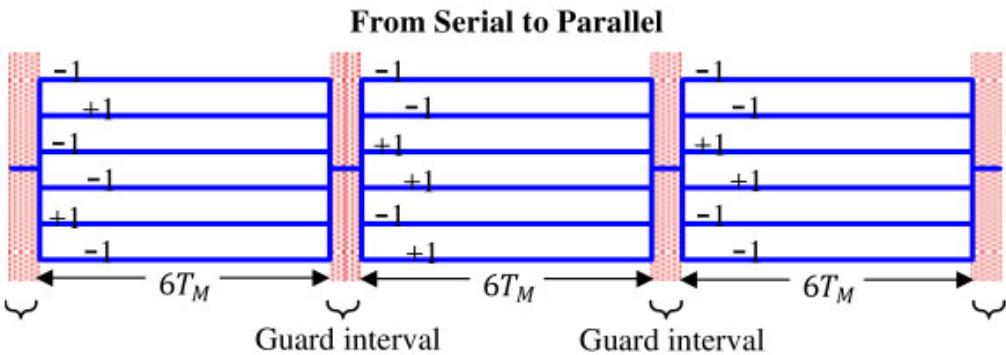
In Figure 9.3a, a serial data stream with symbol duration T_M is shown. Next, it is segmented into blocks of $N = 6$ symbols and a guard interval is inserted within each such block as plotted in Figure 9.3b. If you have read Section 8.3.8 on frequency domain equalization, then this guard interval is actually the Cyclic Prefix (CP). Otherwise, just take this guard interval as a shield for preventing the next block from the multipath interference of the previous block. Next, each symbol in a block of $N = 6$ symbols is sent on a set of $N = 6$ parallel wires shown in Figure 9.3c where each wire is $N = 6$ times longer than the symbol duration T_M . Due to parallelism, there is



(a) A serial symbol stream with symbol duration T_M



(b) A guard interval of duration T_M inserted into the above signal



(c) $N = 6$ symbols collected into one block and sent on N parallel wires

Figure 9.3: A serial data stream is segmented into blocks of $N = 6$ parallel wires where the length of each wire is $N = 6$ times that of the original symbol T_M

no interference within the data symbols on one set. Moreover, the guard interval prevents the multipath of one block smearing into the next as long as the channel length is lesser than the guard interval.

The main theme to focus in Figure 9.3 is how the serial symbol stream of +1s and -1s are mapped on parallel symbol streams and sent on separate wires in the air. For

example, the first six symbols $-1, +1, -1, -1, +1, -1$ are all sent to the first block starting from top to bottom.

A simple example demonstrates what we gain from such a serial to parallel conversion of the data symbols.

Note 9.1 Fooling the Channel

Assume that the wireless channel in which such a communication system is deployed has a maximum delay spread T_{Del} of about a symbol time $1T_M$, i.e., it interferes with one next symbol. Then, the multipath copies of the serial link will arrive T_M seconds later. Temporarily neglecting the impact of carrier wave, each symbol copy will overlap with the next symbol, thus completely randomizing the Rx sequence (since the source generates an independent stream of $+1$ s and -1 s). According to the definition, this is a frequency selective channel and the resultant signal is drawn in the upper part of Figure 9.4.

For the parallel mode, a copy of the elongated symbol of duration $6T_M$ will also be delivered after a delay of $1T_M$, since the multipath channel is the same. However, as opposed to a complete overlap of one symbol into the next, we will have a relatively short overlap of just $1T_M$ which is $1/6$ or 16.7% of the parallel symbol duration $6T_M$. Consequently, this is the case of a frequency flat channel if it is defined as having a delay spread of approximately 15% of the symbol time.

"By introducing parallelism, we have fooled a frequency selective channel into behaving as a frequency flat channel!"

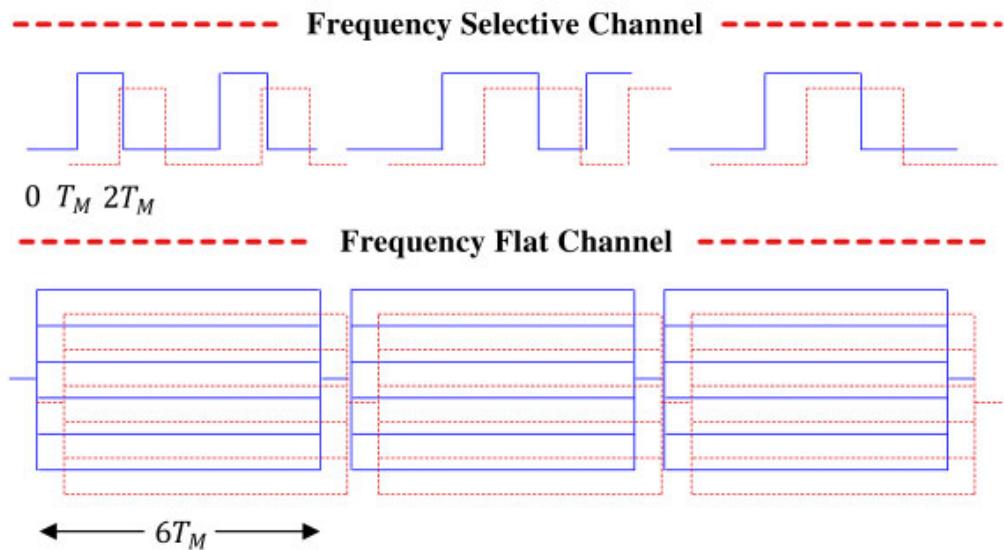


Figure 9.4: By rearranging the symbols from serial to parallel, a frequency selective channel (for symbols of duration T_M) has been converted into a frequency flat channel (for elongated symbols of duration $6T_M$)

This is shown in the lower part of Figure 9.4 where by rearranging the symbols from serial to parallel, a frequency selective channel for symbols of duration T_M has been converted into a frequency flat channel for elongated symbols of duration $6T_M$.

Also observe that due to the guard interval, the delay spread of the first elongated symbol will not bump into the next elongated symbol. By increasing N significantly larger than 6, a severe (much longer) frequency selective channel can be turned into a frequency flat channel as well.

Parallel Wires in the Air

To avoid using the confusing terms of a symbol and an elongated symbol, we will call this *an OFDM symbol* from now onwards. By definition, the duration of an OFDM symbol is NT_M , i.e., N times longer than a conventional symbol duration T_M (later, we will see that its duration increases after inserting the guard interval). Now, the question is how to create these parallel wires in the air.

To create those parallel wires of length NT_M in the air, let us go back to Section 1.7 on the set of discrete frequencies. For N symbols in a serial link, we collect a total of N samples at a rate of $F_S = 1/T_M$, or $L = 1$ sample/symbol, thus spanning a total time duration of NT_M seconds. This description is just to enforce the concept of 1 sample per data symbol which we now change to NT_S seconds below.

In single-carrier systems, we have used the notation $F_S = 1/T_S$ as the sample rate and $R_M = 1/T_M$ as the symbol rate. The relation between them is determined by the number of samples/symbol L .

$$F_S = L/T_M, \quad T_M = LT_S$$

A signal oversampled by L is required in single-carrier systems for Tx and Rx filtering, an example of which is pulse shaping at the Tx and matched filtering at the Rx. Such an oversampled signal is also employed (but not necessary) to perform various tasks such as symbol timing synchronization and fractionally-spaced equalization. In the above description, we saw that the target is to convert a high rate serial symbol stream into a bank of N low rate parallel symbol streams. Since this represents $L = 1$ sample per symbol and most of the blocks (such as synchronization and equalization) in an OFDM Rx design do not require oversampling, we switch to the conventional $F_S = 1/T_S$ notation to denote the sample rate and sample time, respectively.

Now operating on N samples, consider that the lowest frequency that can be represented by these N samples is the one by a sinusoid that completes one full cycle – and no more – during this interval of NT_S seconds. Being an inverse of time period, such a frequency is given by $1/(NT_S)$ Hz and is known as the fundamental frequency F_1 .

$$\begin{aligned} I &\rightarrow & V_I(t) &= \cos 2\pi \frac{1}{NT_S} t = \cos 2\pi F_1 t \\ Q &\uparrow & V_Q(t) &= \sin 2\pi \frac{1}{NT_S} t = \sin 2\pi F_1 t \end{aligned}$$

Let us call the expression NT_S that represents one block of N samples as T_u where u stands for the useful part of the symbol. We will shortly see what useful means in this

context.

$$T_u = NT_S \quad (9.1)$$

which implies that $F_1 = 1/T_u$. Now we consider N integer multiples F_k of this fundamental frequency F_1 .

$$F_k = kF_1 = \frac{k}{NT_S} = \frac{k}{T_u} \quad (9.2)$$

which are all orthogonal to each other in T_u seconds. To see why, let us sample these complex sinusoids at a rate of $F_S = 1/T_S$.

$$\begin{aligned} I &\rightarrow V_I[n] = \cos 2\pi \frac{1}{NT_S} t \Big|_{t=nT_S} = \cos 2\pi \frac{1}{NT_S} nT_S = \cos 2\pi \frac{1}{N} n \\ Q &\uparrow V_Q[n] = \sin 2\pi \frac{1}{NT_S} t \Big|_{t=nT_S} = \sin 2\pi \frac{1}{NT_S} nT_S = \sin 2\pi \frac{1}{N} n \end{aligned}$$

If the integer multiples k/N of the fundamental frequency $1/N$ above are constructed for

$$k = -\frac{N}{2}, -\frac{N}{2} + 1, \dots, -1, 0, 1, \dots, \frac{N}{2} - 1$$

then this is the same set we encountered before in Section 1.7. There, we defined orthogonality in Eq (1.47) reproduced below.

$$\begin{aligned} I &\rightarrow \sum_{n=0}^{N-1} \cos 2\pi \frac{k - k'}{N} n = \begin{cases} N & k = k' \\ 0 & k \neq k' \end{cases} \\ Q &\uparrow \sum_{n=0}^{N-1} \sin 2\pi \frac{k - k'}{N} n = 0 \end{aligned} \quad (9.3)$$

and this concept was illustrated in Figure 1.46. In words, all complex sinusoids having frequencies k/N are orthogonal to each other in a duration of $T_u = NT_S$ seconds. By orthogonality, we imply that over a duration of NT_S seconds (one period of the sinusoid with fundamental frequency), a sum of sample-by-sample products of any two of these sinusoids is zero which is the same as their correlation at lag 0.

There we have a set of N parallel wires or waves that do not interfere with each other when correlated over a duration of N samples. *This set of complex sinusoids acts as our invisible printer port in the air!* Next, we move towards a more visual approach of these parallel waves.

Enter the Correlation (Real Sinusoids)

Although we can directly go to the case of complex sinusoids due to their coverage in the previous chapters, we first consider the inphase components only. This is because it allows us to draw the modulated OFDM signal in the form of these sinusoids in Figure 9.6 later in a simple manner.

Some real sinusoids for $N = 6$ are illustrated in Figure 9.5 where $k = 0, 1, \dots, N - 1$. To verify their orthogonality, take for example $k = 0$ and $k = 1$. The sinusoid

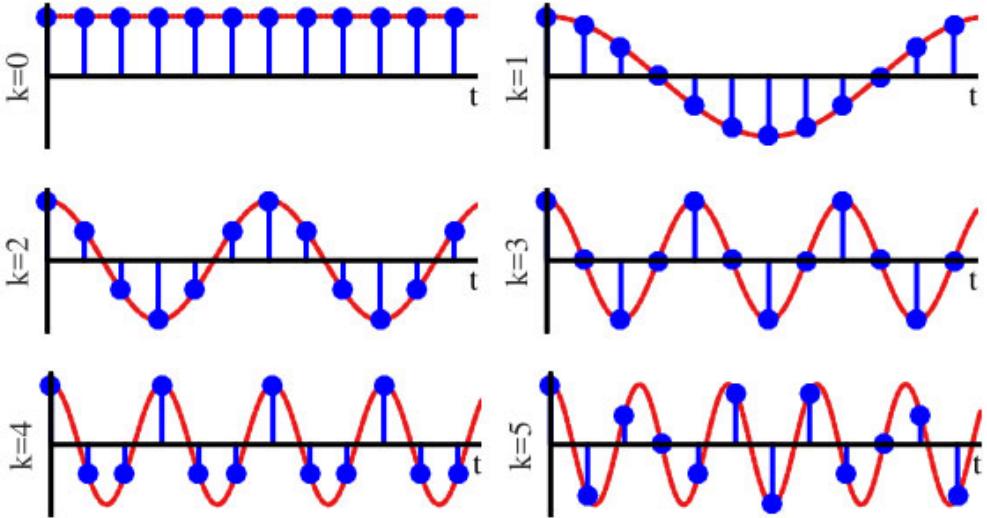


Figure 9.5: Real sinusoids act as a set of parallel waves in the air

with $k = 0$ are all ones, so its sample-by-sample product with $k = 1$ sinusoid yields the same sinusoid back. So for $k = 1$, notice that the sum of all samples are zero.

You can repeat this computation for $k \neq 0$ by simulating in software, the sum of sample-by-sample product between any two sinusoids is zero as long as their frequencies are a multiple of a fundamental frequency, $k = 1$ in this example.

Consequently, as the data symbols $a[k]^{\dagger}$ are produced, we modulate each such sinusoid with one data symbol. For a real sinusoid with frequency k/N ,

$$a[k] \cdot \cos 2\pi \frac{k}{N} n, \quad k = 0, 1, \dots, N-1$$

Finally, all such waveforms are added together to form the cumulative signal as drawn in Figure 9.6. Notice that the symbol stream is the same as the first set of $N = 6$ symbols from Figure 9.3a.

Owing to orthogonality, at the Rx side, a correlation of the above signal is performed with each such sinusoid of frequency k'/N . We know that

$$\sum_{n=0}^{N-1} \left\{ a[k] \cos 2\pi \frac{k}{N} n \right\} \cdot \cos 2\pi \frac{k'}{N} n = \begin{cases} a[k] & k = k' \\ 0 & k \neq k' \end{cases}$$

where a constant scaling factor of 2 has been ignored. This can be proved as follows.

- For $k \neq k'$, the orthogonality condition in Eq (9.3) simply implies the result to be zero. This was proved for a real sinusoid in the discussion after Eq (1.46).
- For $k = k'$, use the identity $\cos^2 \theta = 0.5(1 + \cos 2\theta)$. The first term is a constant and produces $a[k]$ at the output, while the second term is a cosine whose summation over a whole period (or a set of periods) for $n = 0$ to $N - 1$ is zero.

[†]We will shortly see why the data symbols are denoted as $a[k]$ here instead of $a[m]$ as before in the text. This is due to the presence of the iDFT next, the primary task of which is considered as taking a signal from frequency domain into time domain. For this reason, the symbols are assumed to be produced in frequency domain.

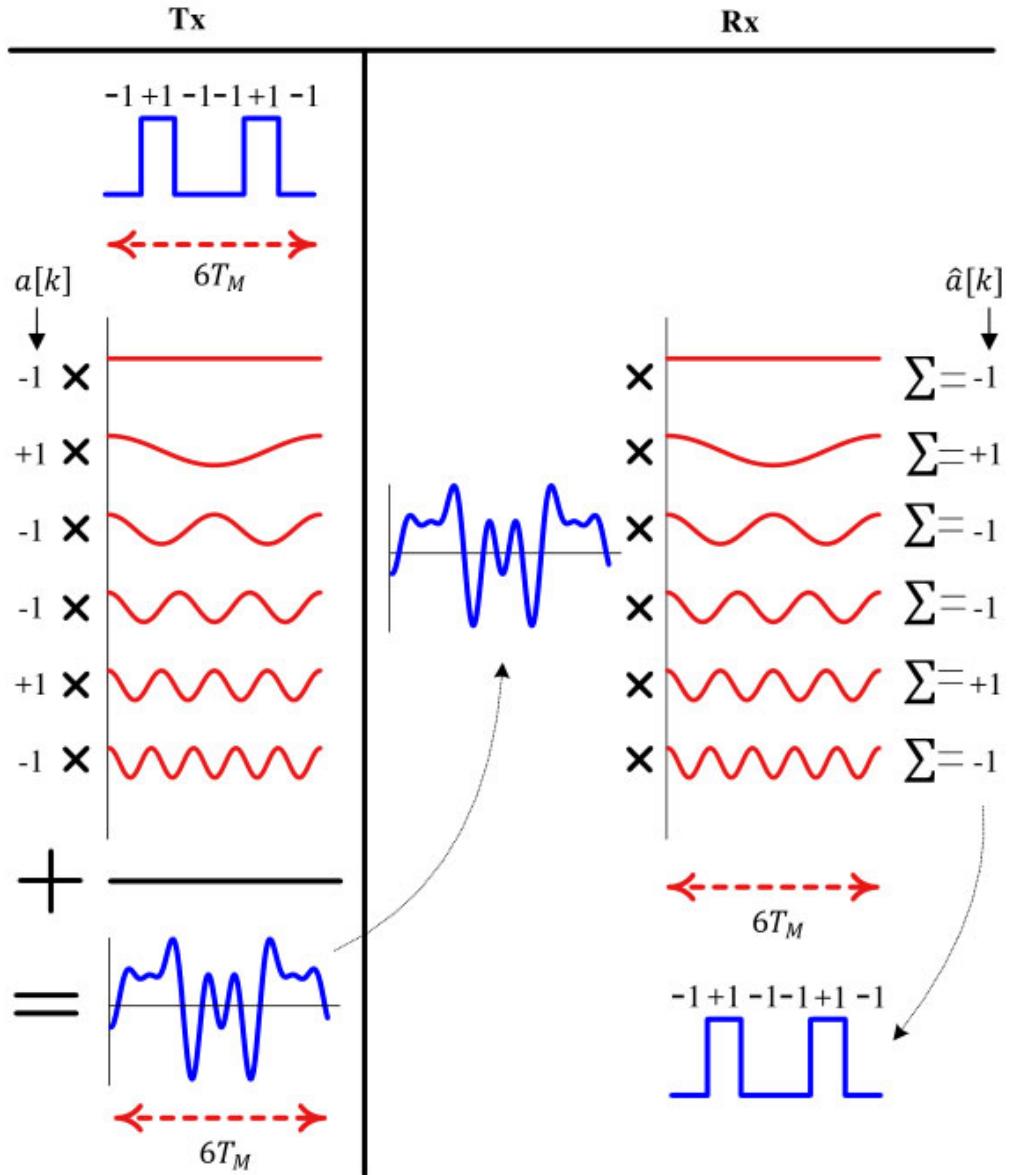


Figure 9.6: Data symbols modulated onto orthogonal sinusoids and added together to form the Tx signal. At the Rx, it is correlated with orthogonal sinusoids to generate symbol decisions

Consequently, if each such sinusoid of frequency k/N was modulated with a different data symbol $a[k]$ and added to all the others, each data symbol would have come out at the Rx side undistorted from the presence of other $N - 1$ symbols in parallel. This is illustrated in Figure 9.6 on the Rx side and can be verified as follows.

Let us form a cumulative signal by adding all N symbols modulated with their

respective sinusoids.

$$\begin{aligned} v[n] &= a[0] \cos 2\pi \frac{0}{N} n + a[1] \cos 2\pi \frac{1}{N} n + \cdots + a[N-1] \cos 2\pi \frac{N-1}{N} n \\ &= \sum_{k=0}^{N-1} a[k] \cos 2\pi \frac{k}{N} n \end{aligned}$$

A correlation with $\cos 2\pi(k'/N)n$ at the Rx would yield

$$\sum_{n=0}^{N-1} \left\{ \sum_{k=0}^{N-1} a[k] \cos 2\pi \frac{k}{N} n \right\} \cdot \cos 2\pi \frac{k'}{N} n = a[k']$$

where a factor of 2 has been ignored again. This can be seen by taking an example of $k' = 1$.

$$\begin{aligned} &\sum_{n=0}^{N-1} \left\{ a[0] \cos 2\pi \frac{0}{N} n \right\} \cdot \cos 2\pi \frac{1}{N} n + \sum_{n=0}^{N-1} \left\{ a[1] \cos 2\pi \frac{1}{N} n \right\} \cdot \cos 2\pi \frac{1}{N} n + \\ &\quad \cdots + \sum_{n=0}^{N-1} \left\{ a[N-1] \cos 2\pi \frac{N-1}{N} n \right\} \cdot \cos 2\pi \frac{1}{N} n \\ &= 0 + a[1] + \cdots + 0 = a[1] \end{aligned}$$

This process is drawn in Figure 9.6 on the right hand side, where the cumulative waveform is multiplied with each orthogonal sinusoid generating the estimate for each data symbol $\hat{a}[k]$. Since these sinusoids carry the data symbols $a[k]$ in an independent fashion, these are known as **subcarriers**. A subcarrier is the word you would hear a lot during any discussion on OFDM.

Enter the Correlation (Complex Sinusoids/Subcarriers)

Having clarified the simple case of real sinusoids, we now focus on the realistic scenario of complex sinusoids known as subcarriers in the context of OFDM. A set of subcarriers for $k = -N/2, \dots, N/2 - 1$ is drawn in a 3D plane in Figure 9.7. Since some people are more comfortable with 2D figures, a set of subcarriers in terms of I and Q components is also illustrated in Figure 9.8 for $N = 4$.

For a subcarrier with frequency k/N and a modulation scheme with no Q component (such as BPSK), we can write

$$\begin{array}{ll} I & \rightarrow a[k] \cos 2\pi \frac{k}{N} n \\ Q & \uparrow a[k] \sin 2\pi \frac{k}{N} n \end{array}$$

More generally, for a QAM scheme with $a_I[k]$ and $a_Q[k]$ as I and Q parts, respectively, we can multiply them with a subcarrier with frequency k/N as follows.

$$\begin{array}{ll} I & \rightarrow a_I[k] \cos 2\pi \frac{k}{N} n - a_Q[k] \sin 2\pi \frac{k}{N} n \\ Q & \uparrow a_Q[k] \cos 2\pi \frac{k}{N} n + a_I[k] \sin 2\pi \frac{k}{N} n \end{array}$$

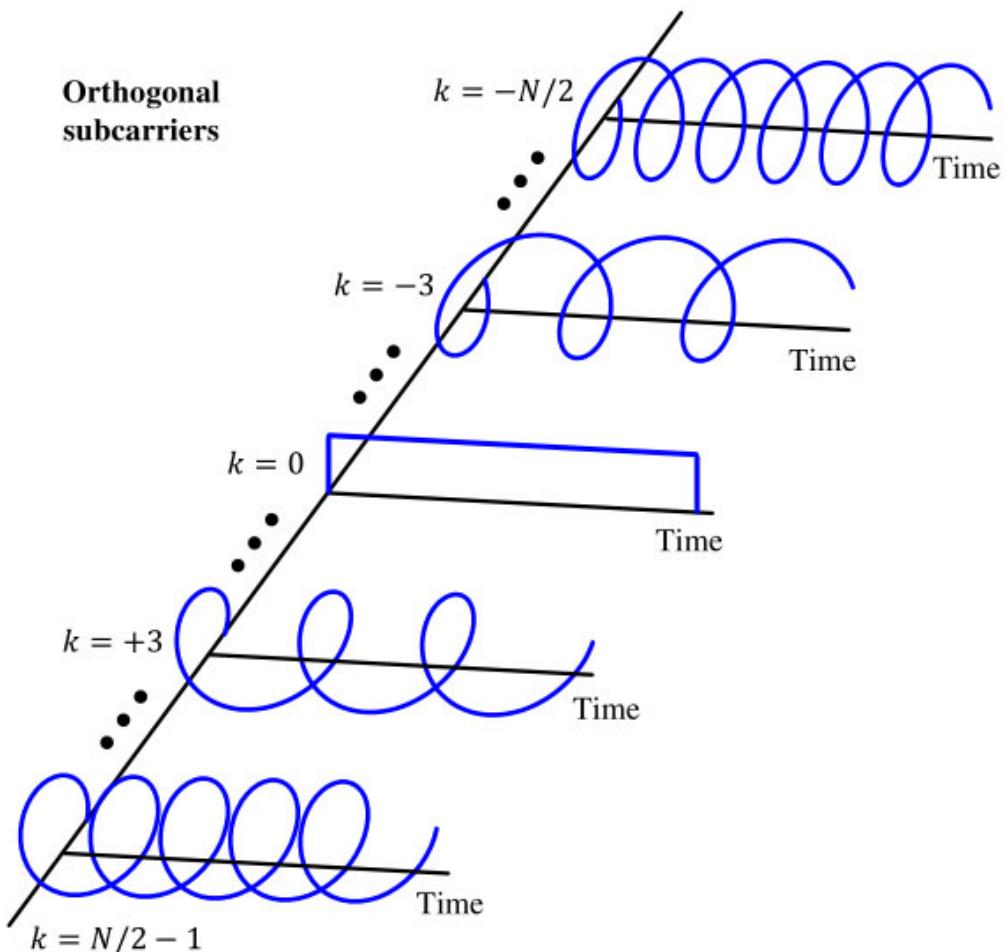


Figure 9.7: Complex subcarriers (which are nothing but complex sinusoids) act as a set of parallel waves in the air. Notice that these are the same sinusoids used in the computation of the DFT

With the concept of subcarriers in place, we can clearly see how this modulation process can be very efficiently implemented: comparing the above expression with the definition of an iDFT in Eq (1.54), *the subcarriers themselves are the DFT sinusoids and the data symbol modulation process is exactly taking the iDFT of the symbol*

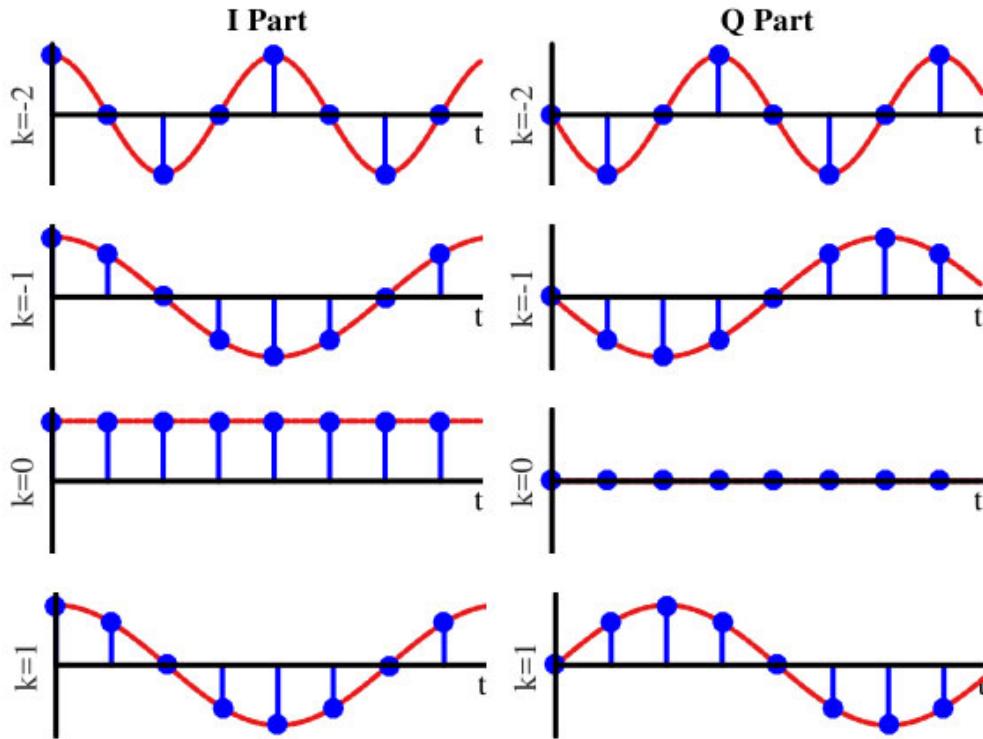


Figure 9.8: I and Q components of the subcarriers (which are nothing but complex sinusoids) that act as a set of parallel waves in the air. Notice that these are the same sinusoids used in the computation of the DFT

sequence $a[k]$. Consequently, OFDM can be implemented through iDFT as[†]

$$\begin{aligned}
 I \rightarrow & v_I[n] = \frac{1}{N} \sum_{k=-N/2}^{N/2-1} \left[a_I[k] \cos 2\pi \frac{k}{N} n - a_Q[k] \sin 2\pi \frac{k}{N} n \right] \\
 Q \uparrow & v_Q[n] = \frac{1}{N} \sum_{k=-N/2}^{N/2-1} \left[a_Q[k] \cos 2\pi \frac{k}{N} n + a_I[k] \sin 2\pi \frac{k}{N} n \right]
 \end{aligned} \tag{9.4}$$

Comparing it with the definition of iDFT which is used to transform a signal from frequency domain to time domain, we can see why the terminology ‘frequency domain symbols’ is used for data symbols denoted by $a[k]$. This has nothing to do with some frequency domain property of data symbols but simply because the Tx signal is naturally considered to be in time domain and any signal before the iDFT block is then considered to be present in frequency domain.

[†]In other resources, you will often see the equivalent expression below.

$$v[n] = \frac{1}{N} \sum_{k=-N/2}^{N/2-1} a[k] \cdot \exp \left(j2\pi \frac{k}{N} n \right)$$

Logically, the reverse operation, i.e., a DFT, is performed at the Rx side to transform the Rx signal into frequency domain for both equalization and detection purpose. The data symbol estimates are given as

$$\begin{aligned} I &\rightarrow \hat{a}_I[k] = \sum_{n=0}^{N-1} \left[v_I[n] \cos 2\pi \frac{k}{N} n + v_Q[n] \sin 2\pi \frac{k}{N} n \right] \\ Q &\uparrow \hat{a}_Q[k] = \sum_{n=0}^{N-1} \left[v_Q[n] \cos 2\pi \frac{k}{N} n - v_I[n] \sin 2\pi \frac{k}{N} n \right] \end{aligned} \quad (9.5)$$

Although a mathematical proof can be constructed by utilizing the subcarriers orthogonality, suffice it to say that the iDFT operation and DFT operation are inverse of each other and hence the data symbols appearing in Eq (9.4) can be perfectly recovered by applying Eq (9.5).

Needless to say, both the iDFT and the DFT are implemented through the iFFT (Inverse Fast Fourier Transform) and FFT (Fast Fourier Transform) procedures, respectively, which is not a separate transform but a very efficient algorithm to compute the iDFT and DFT, respectively. From here onwards, we will refer to going to and coming back from frequency domain through FFT operations interchangeably for the DFT.

Before performing such an FFT operation, however, we need to ensure that the convolution between the Tx signal and the wireless channel happens in a circular fashion instead of a linear one. This is what we explain next.

Inserting the Cyclic Prefix (CP)

An actual OFDM symbol is completed after inserting the Cyclic Prefix (CP) and consists of

$$N_{\text{OFDM}} = N + N_{\text{CP}}$$

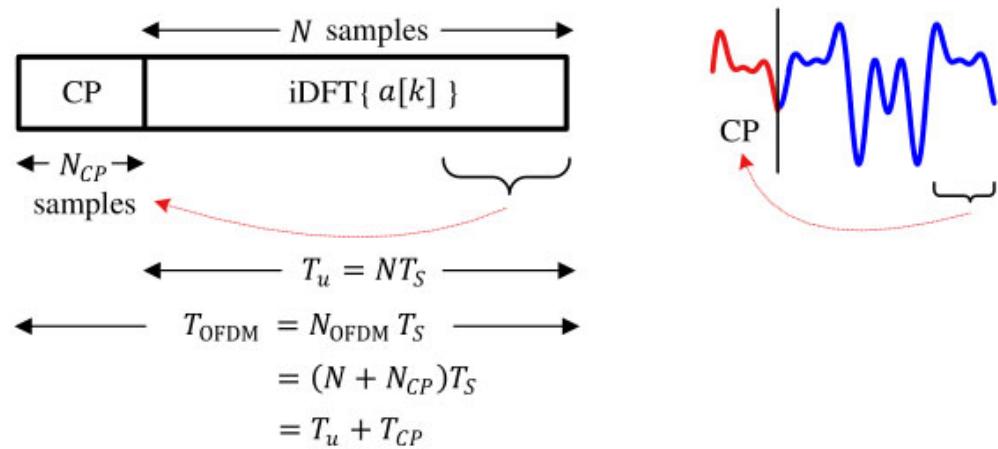


Figure 9.9: A Cyclic Prefix (CP) in the context of an OFDM symbol

samples instead of N samples. We will now see how useful part $T_u = NT_S$ implies the actual iFFT of $a[k]$ while a Cyclic Prefix (CP) is just an added redundancy.

In the discussion on Single-Carrier Frequency Domain Equalization (SC-FDE) in Section 8.3.8, we found that a Cyclic Prefix (CP) consists of the last N_{CP} samples of the Tx sequence (where N_{CP} is determined by the maximum expected channel length). For an OFDM symbol of length N , the process to insert the CP is shown in Figure 9.9. The time duration of the complete OFDM symbol is therefore defined as

$$T_{\text{OFDM}} = N_{\text{OFDM}} T_S = (N + N_{CP}) T_S = T_u + T_{CP}$$

where

$$T_{CP} = N_{CP} T_S$$

It is important to note that as opposed to the last N_{CP} symbols prefixed back in an SC-FDE system, the last $N_{CP} = T_{CP}/T_S$ samples of $\text{iDFT}\{a[k]\}$ are prepended at the start of the OFDM symbol. The job of the CP is threefold in an OFDM system.

Inter-Symbol Interference (ISI): To provide a guard interval for the channel such that no interference from the previous OFDM symbol enters the next OFDM symbol. This happens when the length of the CP is chosen as the maximum expected channel length. This was explained in detail in the context of single-carrier frequency domain equalization in Section 8.3.8 and I recommend revisiting that section to understand this point.

Converting linear convolution into circular convolution: A CP converts linear convolution between the Tx signal and the wireless channel into circular convolution.

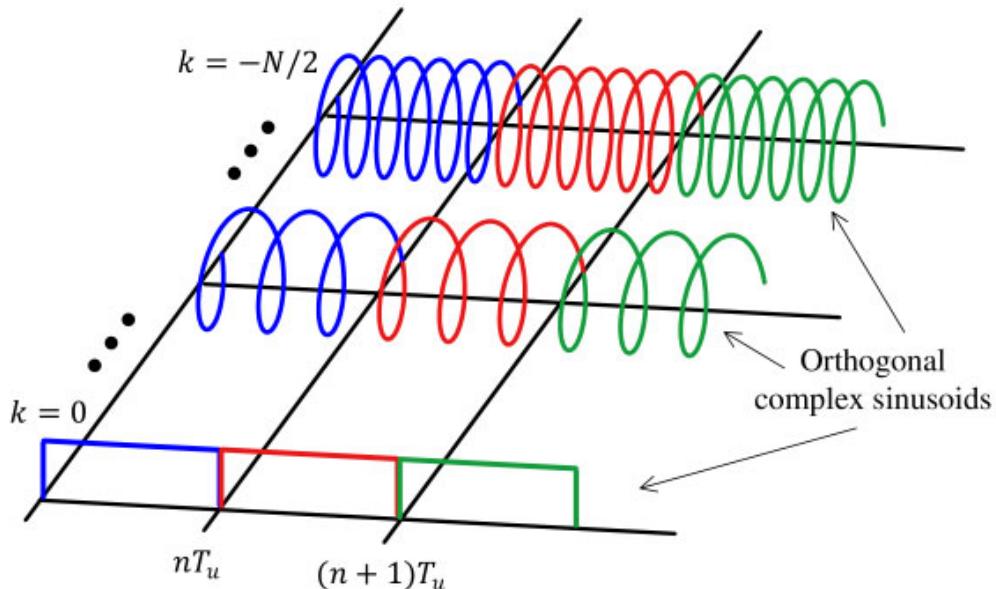


Figure 9.10: Orthogonal subcarriers (complex sinusoids) in an OFDM symbol without a Cyclic Prefix (CP) or guard interval

This circular convolution enables the discrete frequency domain product between their respective transforms. Again, this was explained in significant detail in the context of single-carrier frequency domain equalization in Section 8.3.8.

Inter-Carrier Interference (ICI): In my opinion, a detailed mathematical reasoning on the prevention of ICI through a CP is not necessary and probably too cumbersome for a non-mathematician. For the understanding of the basic mechanism, I just demonstrate this concept with the help of a few figures.

Figure 9.10 depicts components of an OFDM symbol without any CP or guard interval, where *only the negative subcarriers are drawn* to keep the figure size small. The negative sign in the frequency of the last sinusoid, $-N/2$, appears due to its clockwise direction of rotation. These components are the subcarriers that are orthogonal to each other (earlier drawn in Figure 9.8 with I and Q parts separated) and scaled by data symbol $a[k]$ (all assumed +1 here).

The actual OFDM signal is formed through summing all the corresponding samples in these subcarriers. Notice that since the subcarriers are completing their full number of cycles within each OFDM symbol, they are all orthogonal to each other within that OFDM symbol and neither ISI nor ICI occurs.

Next, Figure 9.11 draws the same OFDM signal without any CP but after it has passed through a wireless channel. In this figure, only $k = 0$ subcarrier is shown for all three intervals while the rest of the subcarriers are drawn for OFDM symbol 1 only for clarity. It is straightforward to identify the ISI due to the intruding of one OFDM symbol into the next. The issue of ICI is not as clear here.

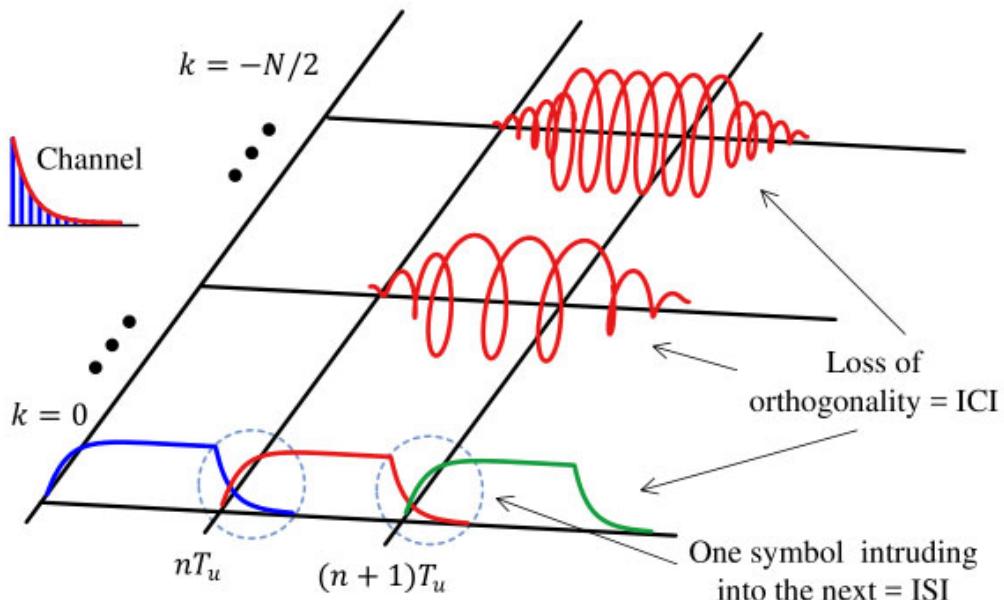


Figure 9.11: After passing through the wireless channel, Inter-Carrier Interference (ICI) arises due to the loss of orthogonality among subcarriers and Inter-Symbol Interference (ISI) arises due to one OFDM symbol intruding into the next

For this purpose, we recall Section 1.10.1 here in which Figure 1.64 and Figure 1.65 hold particular importance. In these figures, we discovered that signals with frequencies other than the set k/N are not periodic in the observation window and are suddenly terminated without completing the final cycle in full. This discontinuity violates the orthogonality relation yielding a non-zero result and becomes responsible for spectral contributions over the entire set of frequencies k/N . Due to the convolution with the wireless channel, a loss of exact complete periods within an OFDM symbol is evident in our current Figure 9.11 which gives rise to ICI among subcarriers. *The data symbols $a[k]$ are not travelling on independent parallel waves anymore!*

On a side note, this ICI was not an issue in single-carrier frequency domain equalization systems because there are no independent subcarriers related to the data symbols $a[k]$. There are subchannels though, which exhibit the impact of each data symbol inherently spread over all of them.

When a CP is inserted with a length greater than the maximum expected channel length, the effect of ISI remains limited to the CP duration. This is drawn in Figure 9.12a. The CP also acts as a buffer against introducing ICI in the system because the loss of orthogonality only happens within the CP duration. When it is removed, it not only helps in getting rid of the ISI in the system but the remaining parts of the subcarriers exhibit a completion of rotation within the useful period T_u , thus restoring the orthogonality among the subcarriers. The OFDM symbol is ICI-free and the data symbols $a[k]$ can then be independently detected. This is illustrated in Figure 9.12b.

Later, we will study that ICI is also caused by a Carrier Frequency Offset (CFO) and the methods to prevent it in that context.

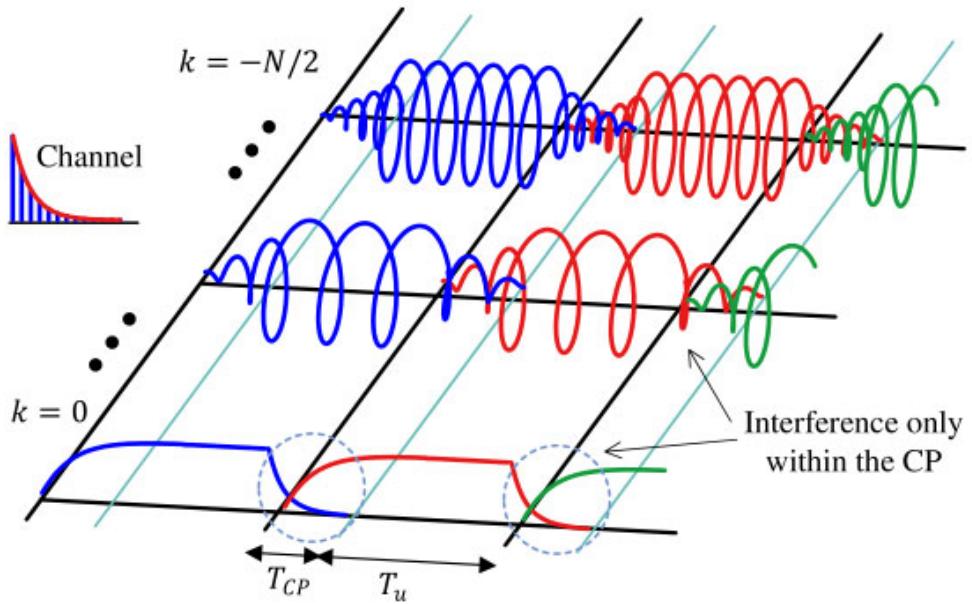
Now we discuss why the equalization becomes simpler in an OFDM system as compared to a single-carrier system.

Why Equalization Becomes Simpler

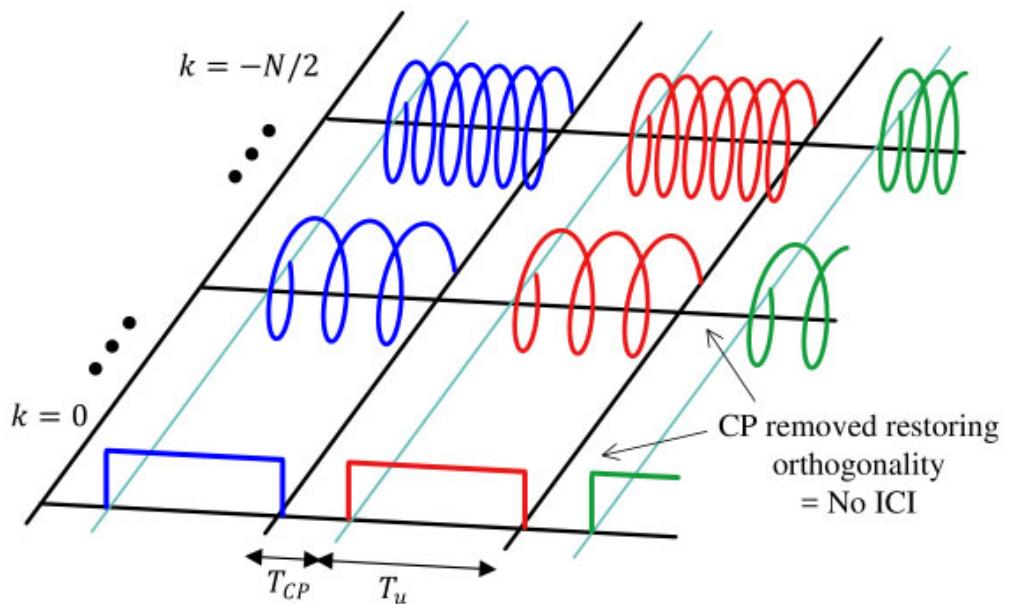
Complexity of equalization has been the bottleneck in successful implementation of high rate single-carrier systems. The introduction of subcarriers in OFDM simplifies this process to a considerable level and has been the major factor in its adoption in high speed wireless systems. Although this process of simple equalization is better explained in frequency domain later, we still have a look at the channel impact in time domain to get an idea.

For this purpose, we continue with $N = 6$ subcarriers shown earlier in Figure 9.6 as an example and see what happens in the presence of a multipath channel. This is drawn in Figure 9.13 where two observations are important.

1. We have explained before that as long as the delay spread of the channel remains less than 10 or so percent of the symbol duration, the channel is frequency flat fading and hence multiplies the signal with a single coefficient, instead of a complete convolution as in the case of a frequency selective fading channel. Now the channel example shown here is a frequency selective channel for the serial symbol stream because the multipath duration is equal or greater than a symbol time T_M . However, due to the elongated symbol length, i.e., NT_M here, it has been converted into a frequency flat fading channel *for each subcarrier*.



(a) ISI is limited to the CP interval



(b) With CP removed, the subcarriers are orthogonal and hence no ICI occurs

Figure 9.12: Effect of CP on ISI and ICI

2. With the addition of a CP, the orthogonality of each subcarrier is ensured and the channel impacts each subcarrier in an individual manner. Notice in Figure 9.13 how each subcarrier will be deteriorated by the channel but discarding the CP

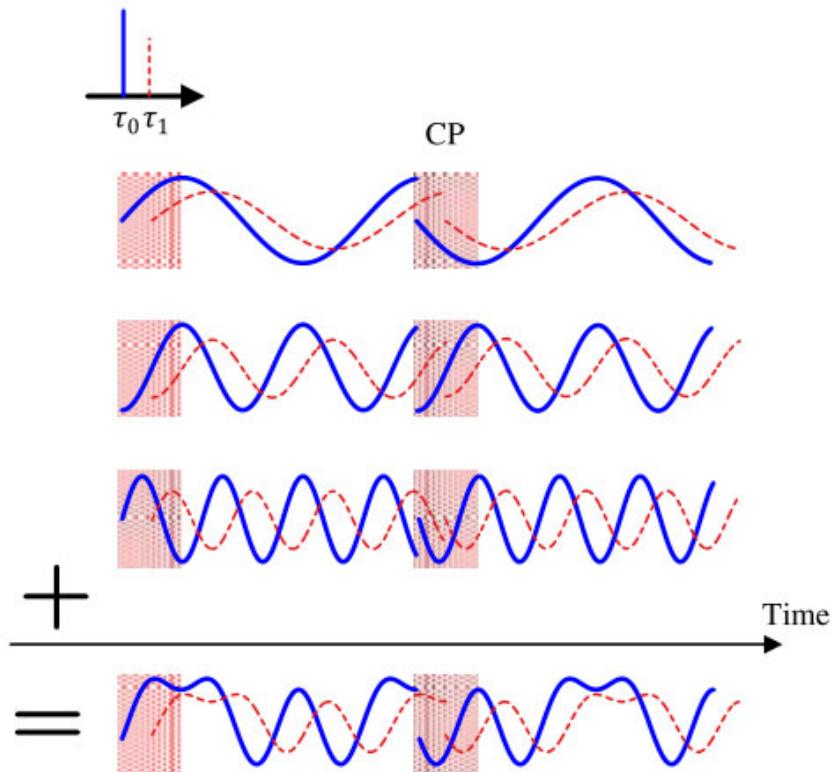


Figure 9.13: In the presence of a CP, multipath in time domain impact each subcarrier individually due to their orthogonality. Moreover, an elongated symbol implies flat fading for each individual subcarrier

gets rid of both ISI from one OFDM symbol to the next as well as ICI among difference subcarriers. Then, for each subcarrier, the channel is just a summation of complex sinusoids *of the same frequency with different amplitudes and delays* which all add up to generate a resultant amplitude and phase without changing the frequency. Let us derive this resultant channel for a subcarrier k .

Referring to Figure 9.13, assume that the delay of the direct path $\tau_0 = 0$ and its amplitude $\rho_0 = 1$. With a single multipath with delay τ_1 (which is $n_1 = \tau_1/T_S$ in terms of samples) and amplitude ρ_1 , the I output can be expressed as below. Recalling that the subcarriers are complex sinusoids, *the actual amplitudes γ_i are complex* and given as a function of carrier frequency F_C in Eq (8.5). Nevertheless, for removing clutter to simplify the derivation, here we take them as real and equal to ρ_i .

$$\begin{aligned}
 & \cos 2\pi \frac{k}{N} n + \rho_1 \cos 2\pi \frac{k}{N} (n - n_1) = \cos 2\pi \frac{k}{N} n + \\
 I \rightarrow & \quad \rho_1 \cos 2\pi \frac{k}{N} n \cdot \cos 2\pi \frac{k}{N} n_1 + \rho_1 \sin 2\pi \frac{k}{N} n \cdot \sin 2\pi \frac{k}{N} n_1 \\
 & = \left(1 + \rho_1 \cos 2\pi \frac{k}{N} n_1\right) \cos 2\pi \frac{k}{N} n + \left(\rho_1 \sin 2\pi \frac{k}{N} n_1\right) \sin 2\pi \frac{k}{N} n
 \end{aligned}$$

where we have used the identity $\cos(A - B) = \cos A \cos B + \sin A \sin B$. On a similar note,

$$\begin{aligned} \sin 2\pi \frac{k}{N} n + \rho_1 \sin 2\pi \frac{k}{N} (n - n_1) &= \sin 2\pi \frac{k}{N} n + \\ Q \uparrow &\quad \rho_1 \sin 2\pi \frac{k}{N} n \cdot \cos 2\pi \frac{k}{N} n_1 - \rho_1 \cos 2\pi \frac{k}{N} n \cdot \sin 2\pi \frac{k}{N} n_1 \\ &= -\left(\rho_1 \sin 2\pi \frac{k}{N} n_1\right) \cos 2\pi \frac{k}{N} n + \left(1 + \rho_1 \cos 2\pi \frac{k}{N} n_1\right) \sin 2\pi \frac{k}{N} n \end{aligned}$$

where $\sin(A - B) = \sin A \cos B - \cos A \sin B$. From the multiplication rule of complex signals $I \cdot I = Q \cdot Q$ and $Q \cdot I + I \cdot Q$, we can see that the subcarrier k is being multiplied with a *single channel coefficient* given by

$$\begin{aligned} I \rightarrow &\quad H_I[k] = 1 + \rho_1 \cos 2\pi \frac{k}{N} n_1 \\ Q \uparrow &\quad H_Q[k] = -\rho_1 \sin 2\pi \frac{k}{N} n_1 \end{aligned} \tag{9.6}$$

For each subcarrier, even when the number of multipath copies increases, the number of terms in the above expression increases but the frequency of their sum signal remains the same (equal to k/N). Subsequently, the important point to remember is that each such sum, for each subcarrier, remains orthogonal to all the other subcarriers $k' \neq k$.

Finally, we need to change the following in the above equation.

- We can write using the useful symbol duration $T_u = NT_S$ from Eq (9.1),

$$2\pi \frac{k}{N} n_1 = 2\pi \frac{k}{N} \frac{\tau_1}{T_S} = 2\pi k \frac{\tau_1}{T_u}$$

- When the actual *complex gain* γ_1 is taken into account, the above expression includes complex multiplication of γ_1 with the complex sinusoid of frequency $-2\pi k \tau_1 / T_u$, see Eq (9.6).
- The actual channel response consists of N_{MP} paths that are summed at the Rx antenna.

Taking all this into account, the channel coefficient becomes

$$\begin{aligned} I \rightarrow &\quad H_I[k] = \sum_{i=0}^{N_{MP}-1} \gamma_{i,I} \cos 2\pi k \frac{\tau_i}{T_u} + \gamma_{i,Q} \sin 2\pi k \frac{\tau_i}{T_u} \\ Q \uparrow &\quad H_Q[k] = \sum_{i=0}^{N_{MP}-1} \gamma_{i,Q} \cos 2\pi k \frac{\tau_i}{T_u} - \gamma_{i,I} \sin 2\pi k \frac{\tau_i}{T_u} \end{aligned} \tag{9.7}$$

which is the very well known single tap channel coefficient for an OFDM system exhibiting flat fading for each subcarrier[†].

[†]In complex notation, these channel gains are given by

$$H[k] = \sum_{i=0}^{N_{MP}-1} \gamma_i \cdot \exp(-j2\pi k \tau_i / T_u)$$

While we will have a better look at this process in frequency domain, the orthogonal subcarriers each experiencing a flat fading channel is the main reason behind simpler equalization required for an OFDM system.

In summary, OFDM increases the symbol duration from T_M to $T_u = NT_M$ such that the information is transmitted in *many low rate parallel streams*. The duration of each low rate stream T_u is significantly longer than the delay spread T_{Del} of the channel and hence no ISI occurs, thanks to the guard interval.

$$T_u \gg T_{\text{CP}} > T_{\text{Del}} \quad (9.8)$$

This eases the equalization part of the Rx processing when we use our Master Algorithm, i.e., correlation, at the Rx to separate the subcarriers.

Having viewed the concept of OFDM in time domain, we look at its frequency domain interpretation.

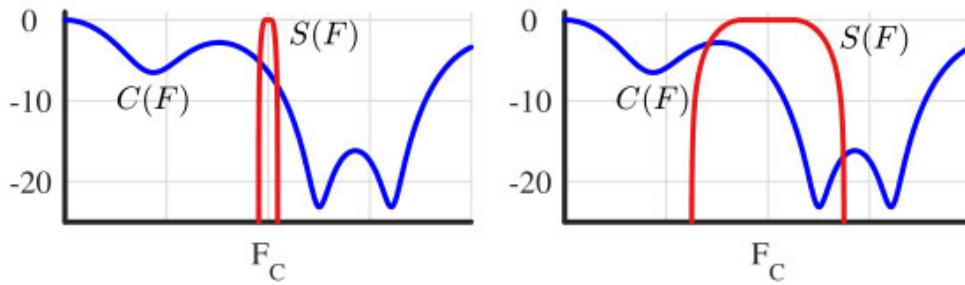
9.2.2 To a Sliced Bread

We start with the problem high rate single-carrier systems face in frequency domain and then explain how OFDM addresses that issue.

Increasing the Symbol Rate

The wireless channel from Chapter 8 has an impulse response derived from the contribution of each multipath component. Its frequency response was also drawn in several figures throughout that chapter, two of which are redrawn here in Figure 9.14a for a frequency flat fading channel and in Figure 9.14b for a frequency selective fading channel. Here, $C(F)$ and $S(F)$ are the frequency domain representations of the channel and Tx signal, respectively.

With respect to frequency domain, the signal at the Rx is a product of the spectra of the Tx signal and the wireless channel. The low data rate signal in Figure 9.14a needs less manipulation by the Rx to get the original data back. Essentially for this kind of signal, the channel acts just as a single multiplier that can be equalized through estimating that channel coefficient and dividing the Rx signal by the estimated value.



(a) A frequency flat fading channel (b) A frequency selective fading channel

Figure 9.14: Our symbol rate determines the signal bandwidth and consequently whether the channel behaves as a frequency flat or selective fading channel

Consequently, the equalization process reduces to a single division operation[†].

Coming to Figure 9.14b, it is evident that due to the high data rate and hence large signal bandwidth, the whole bandwidth extends beyond the coherence bandwidth of the channel and hence the Rx magnitude is distributed across a wide spectral region. As the term selectivity implies, when some part of the signal spectrum is in a deep fade, the rest of the portion is not. This discriminatory treatment of different portions of the signal spectrum by the channel is a form of frequency diversity and hence a good thing if properly equalized.

However, this high data rate signal needs a lot of Rx processing to equalize the channel due to its large bandwidth. Such a computationally complex equalizer is incompatible with the high rate symbol stream as discussed in detail in Section 8.3.7 in the context of the drawbacks of a time domain equalizer.

Subcarriers/Complex Sinusoids in Frequency Domain

To view this process in frequency domain, we need to know how the Fourier Transform of a complex sinusoid looks like.

- If we had infinitely long subcarriers, then their Fourier Transform would just have been an impulse at that frequency. This is shown at the top of Figure 9.15. Recalling the definition of frequency as cycles/second, the frequency of this impulse is $3/T_u$ because there are three complete cycles of the complex sinusoid in our measurement time T_u .
- However, the subcarriers are only T_u wide in time which is the same as multiplying them with a rectangular signal of length T_u . A rectangular signal has a sinc signal as its Fourier Transform shown in the middle of Figure 9.15.
- Finally, a multiplication in time induces a convolution in frequency. For this reason, the sinc signal in frequency gets shifted to the frequency of the subcarrier, $3/T_u$ in our example.

Now we want to inspect all the subcarriers together in frequency domain. For this purpose, the following two questions need to be answered.

Spacing between any two subcarriers: First, remember from Eq (9.1) that the duration of each subcarrier, or complex sinusoid, is

$$T_u = NT_S \quad (9.9)$$

and there are N such subcarriers forming the overall signal. The indices of these subcarriers range from $k = -N/2$ to $k = N/2 - 1$. From this information, we deduced the actual frequency F_k of each subcarrier using $k/N = F/F_S$ as

$$F_k = F_S \cdot \frac{k}{N} = \frac{k}{NT_S} = \frac{k}{T_u}$$

[†]There can be a question of how to recover when this signal encounters a deep channel fade. In that case, nothing but diversity (a signal replica in some form whether in time, frequency, space, etc.) can recover the signal.

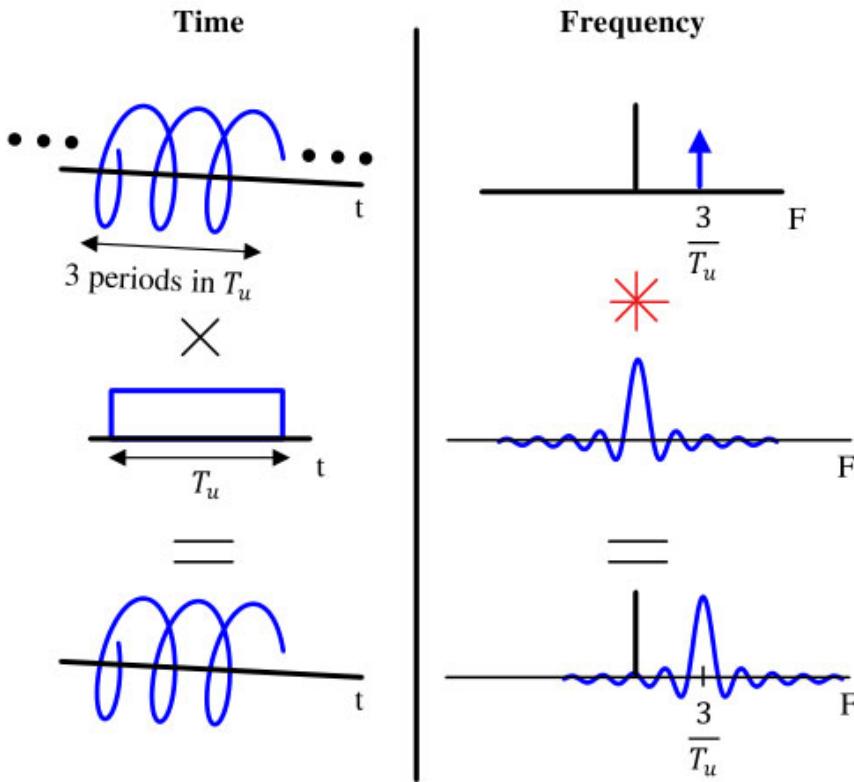


Figure 9.15: Limiting the complex sinusoid or subcarrier in time domain creates its spectrum as a sinc signal in frequency domain

So we have the maximum subcarrier frequency as $-N/(2T_u)$. From $k = 0$ and $k = 1$, we can find the spacing Δ_F between any two subcarriers.

$$\Delta_F = \frac{1}{T_u} - \frac{0}{T_u} = \frac{1}{T_u} \quad (9.10)$$

From Eq (9.9) and Eq (9.10), we have the total bandwidth $B = 1/T_S$ as

$$\frac{1}{\Delta_F} = T_u = NT_S = \frac{N}{B} \quad \rightarrow \quad \text{or} \quad B = N\Delta_F$$

Note 9.2 Bandwidth utilization

By using $T_u = NT_S$, we have derived the subcarrier frequencies as $F_k = k/T_u = k/NT_S$. Here, we have the maximum subcarrier frequency as $-N/(2NT_S) = -1/(2T_S) = -F_S/2$ which is the maximum frequency allowed by the Nyquist's sampling theorem to avoid aliasing. This points to the fact that OFDM efficiently utilizes the whole available bandwidth, although the usage of null subcarriers at the edges reduces this efficiency. Those null subcarriers are required to give

enough transition bandwidth to subsequent simple digital and analog filters thus avoiding costly filters.

Other factors reducing the bandwidth efficiency are the Cyclic Prefix (CP), pilot subcarriers (subcarriers modulated with known data for tracking the channel) and training sequences for initial acquisition.

Location of sinc signal's first zero crossing: To look at the zero crossing of a subcarrier in frequency domain, recall Section 1.10.1 on DFT of a rectangular signal. There, we found in Eq (1.69) that for a full length rectangle (i.e., N samples), the discrete frequency index k_{zc} of the first zero crossing is located at

$$k_{zc} = 1$$

Since the length of the rectangle in our scenario is N samples that cover the duration T_u , we get the first zero crossing frequency F_{zc} as

$$\frac{F_{zc}}{F_S} = \frac{k_{zc}}{N} \rightarrow F_{zc} = \frac{1}{NT_S} = \frac{1}{T_u} \quad (9.11)$$

which interestingly is the same as Eq (9.10).

It is not surprising that the spacing Δ_F between any two subcarriers and the first zero crossing of the sinc signal is the same, equal to $1/T_u$. *This is what orthogonality means in frequency domain!*[†]

With this information at hand, we draw the subcarriers in frequency domain, all at the same time in Figure 9.16a. Notice the zero crossings of the 0th subcarrier passing through

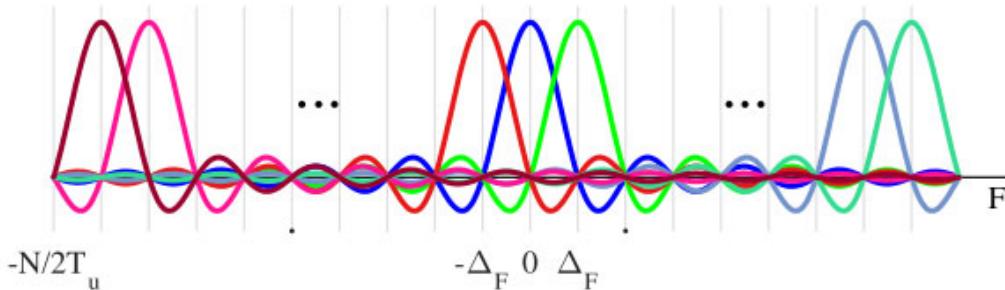
$$\pm\Delta_F = \pm\frac{1}{T_u}, \quad \pm 2\Delta_F = \pm\frac{2}{T_u}, \quad \dots$$

and so on, right at the peaks of the other subcarrier locations $k\Delta_F = k/T_u$.

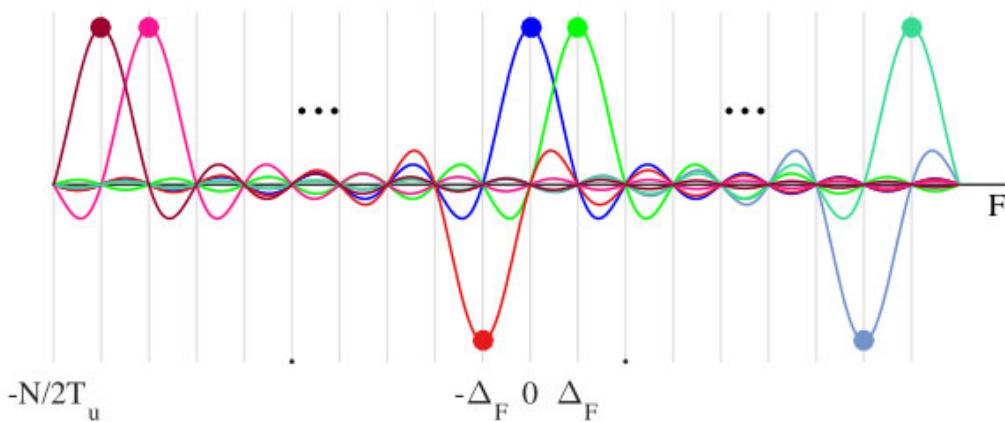
When a linear modulation scheme, such as BPSK, is utilized, the subcarrier amplitudes are modulated as $a[k] = +1$ or $a[k] = -1$ and an iDFT is taken to generate the time domain signal. The DFT at the Rx then needs to be taken at exactly the same set of frequencies k/T_u . In that way, the Rx 'samples' the OFDM waveform in frequency domain at exactly the peaks of subcarriers, exhibiting zero Inter-Carrier Interference (ICI) – maximum contribution comes from the desired subcarrier and zero contribution from all the rest – a phenomenon similar to Inter-Symbol Interference (ISI) and Nyquist's no-ISI criterion in single-carrier systems. This is plotted in Figure 9.16b. Later, we will see how the presence of a CFO causes the 'sampling' in frequency domain at the wrong instants thus introducing ICI in the system.

This is a beautiful dual of the no-ISI phenomenon encountered in time domain in the context of modulated Nyquist pulses and drawn in Figure 3.40 earlier. In time domain of single-carrier systems, the task of the symbol timing synchronization unit is to sample the waveform composed of Nyquist pulses at the peak of the eye diagram (points of no ISI). In frequency domain of multicarrier systems, the task of the carrier

[†]Another kind of orthogonality in frequency domain can be constructed through choosing completely non-overlapping bands for the transmission of two signals. This is the conventional and well known Frequency Division Multiplexing (FDM) system that is too dull for our purpose.



(a) Unmodulated OFDM subcarriers in frequency domain. Notice the zero crossings of each subcarrier passing through other subcarrier frequencies $k\Delta_F = k/T_u$



(b) Modulated OFDM subcarriers in frequency domain. The Rx utilizes exactly the same frequencies $k\Delta_F = k/T_u$ for DFT that ‘samples’ the signal in frequency domain at ideal ICI-free instants

Figure 9.16: Unmodulated and modulated OFDM subcarriers in frequency domain spaced $\Delta_F = 1/T_u$ apart from each other

frequency synchronization unit is to ‘sample’ the waveform composed of subcarriers at the peak of each subcarrier (points of no ICI). Two minor points are in order here.

- Using inverted commas for ‘sampling’ in frequency domain, I refer to the Rx carrier synchronization output. If it contains a zero Carrier Frequency Offset (CFO), we say that the waveform is ‘sampled’ at optimal locations. If there is a residual CFO present, the waveform is ‘sampled’ at a location other than the peaks of the subcarriers.
- Referring to the time frequency duality, a frequency domain eye diagram can also be drawn for an OFDM signal where the peak implies the point of zero CFO, just like the eye diagram in time domain exhibits the point of zero Symbol Timing Offset (STO).

The Puzzle of Violating the Sampling Theorem

From the discussion so far, one might think that these infinitely long sinc signals in frequency domain cause spectral aliasing due to not being bandlimited. This is what the sampling theorem has always promised us. It turns out that the aliasing does occur but in a harmless manner.

Interestingly, sampling at N samples/OFDM symbol implies that an exact replica of the spectrum seen in Figure 9.16 is also present centered at integer multiples of $F = F_S$ Hz or integer multiples of index $k = N$. Nevertheless, the zero crossings of those replica still pass through frequencies $k\Delta_F = k/T_u$ – no matter how far away from their center frequency – and hence inject zero ICI for this sample rate. This could be seen from Figure 9.16 if there was another subcarrier drawn to the left of the subcarrier at $F = -N/2T_u$. That would have been present at frequency

$$F = -F_S + \left(\frac{N}{2} - 1\right) \frac{1}{T_u}$$

and its first right zero crossing would have passed through $F = -N/2T_u$.

In hindsight, this makes perfect sense since this is analogous to Nyquist no-ISI criterion for single-carrier systems, see Note 3.5.

Why Equalization Becomes Simpler

In Section 9.2.1, we saw the time domain picture of the multipath components of the orthogonal subcarriers in Figure 9.13 and claimed that the simpler equalization of OFDM systems is better explained in frequency domain. Let us explore this fact here.

We do not need to denote the DFT of the data symbols $a[k]$ since they are already assumed to be in frequency domain. By convention, the signal before the iDFT at the Tx is assumed to be in frequency domain and after the idFT in time domain. These $a[k]$ modulated the subcarriers in Figure 9.16b. On the other hand, we can denote the DFT of the channel impulse response as $H[k]$.

To find the nature of these channel coefficients $H[k]$ in frequency domain, we first refer to Figure 9.13 where delayed copies of the Tx signal were drawn for each individual subcarrier k . Now we saw in Section 1.9 that a delay in time domain induces a *frequency dependent phase shift* in frequency domain.

$$\text{Time shift } s[(n \pm n_0) \bmod N] \longrightarrow \pm 2\pi \frac{k}{N} n_0 \quad \text{Phase shift}$$

The delay τ_i in seconds becomes τ_i/T_S in samples. Consequently, while each multipath copy of a subcarrier k arrives with the same frequency k/N but an amplitude ρ_i [†], its phase is also changed by an amount $-2\pi(k/N)\tau_i/T_S$. For example, for a single multipath copy arriving with amplitude ρ_1 at τ_1 seconds after the direct path, this phase shift is illustrated in Figure 9.17 along with the direct path which is a graphical manifestation of what we got in Eq (9.6) during time domain description.

The result for several multipath signals can be produced by extending the same concept, i.e., when many of these vectors are added for the same subcarrier frequency

[†]As described before in time domain, remember that the subcarriers are complex sinusoids, *the actual amplitudes γ_i are complex* and given as a function of carrier frequency F_C in Eq (8.5). Nevertheless, for removing clutter to simplify the derivation, here we take them as real and equal to ρ_i .

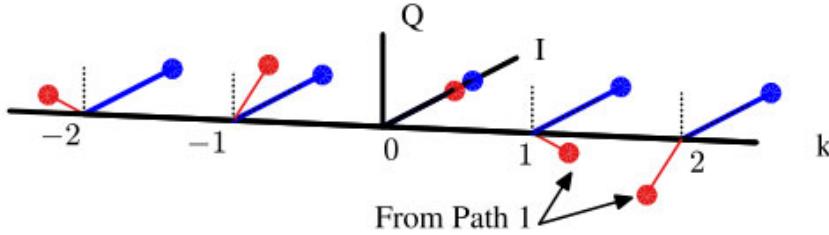


Figure 9.17: Multipath in frequency domain impact each subcarrier individually due to their orthogonality. Moreover, these narrow subcarriers induce flat fading for each k (here, complex channel gains γ_i are assumed real for simplification)

k/N , we get the channel coefficient $H[k]$ for each k as a vector sum of the channel taps, previously expressed in Eq (9.7).

In summary, the output of the channel – a product between the data symbols $a[k]$ and the channel frequency response $H(F)$ – is sampled at N locations in frequency domain by the subcarriers. At the Rx, the time domain signal is sampled at a rate of N samples per OFDM symbol before taking the DFT. The presence of the Cyclic Prefix (CP) converts the linear convolution with the channel into circular convolution. I wanted to write the exact mathematical derivation of this process but then this involves too many indices and instead can become a cause of confusion.

Treading on the visualization path, a signal level diagram was shown in Figure 8.67 in the context of single-carrier frequency domain equalization. There, we also saw how the FFT output $Z[k]$ in Eq (8.72) is a product between the data symbols DFT $A[k]$ and channel frequency response $H[k]$. However, the difference here is that an iDFT has already been taken at the OFDM Tx after defining the data symbols $a[k]$ in frequency domain. Therefore, in this context, the DFT output at the Rx is just the product between the Tx signal DFT $S[k]$ and $H[k]$.

$$Z[k] = S[k] \cdot H[k] \quad (9.12)$$

for each $k = -N/2, \dots, N/2 - 1$

Owing to the orthogonality of the subcarriers which manifests itself in the form of each subcarrier k/T_u passing through zero at other subcarrier frequencies k'/T_u (i.e., zero ICI), $S[k]$ for each k is nothing but the data symbol value $a[k]$. Thus, we can write

$$Z[k] = a[k] \cdot H[k] \quad (9.13)$$

for each $k = -N/2, \dots, N/2 - 1$

Similar to Figure 8.67 in the single-carrier case, Figure 9.18 illustrates this process for our example channel and an all-ones modulated OFDM symbol where the sinc sidelobes in dB are clearly visible. The channel frequency response $H(F)$ is sampled by the DFT at the subcarrier frequencies $k\Delta_F = k/T_u$ to yield $H[k]$. This is shown as the dashed red curve, the Δ_F -spaced samples of which are multiplying each subcarrier data individually. It is evident that such a wideband channel has been converted into many parallel *frequency flat subchannels* through using the subcarriers.

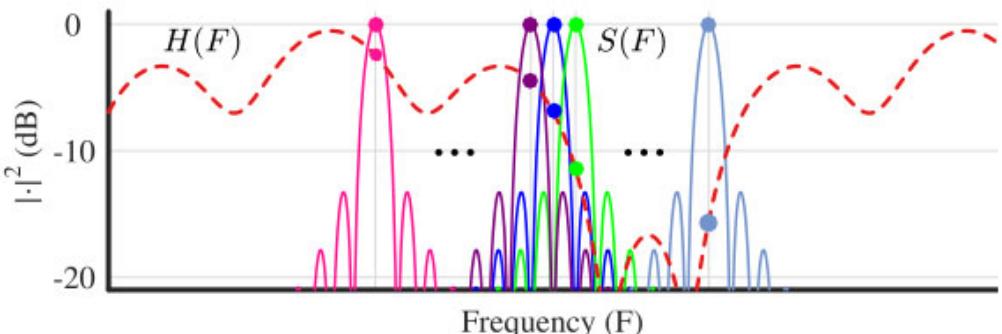


Figure 9.18: The DFT output at the Rx is the spectral product between the data symbol values $a[k]$ and channel frequency response $H[k]$ sampled at subcarrier frequencies $k\Delta_F = k/T_u$

Therefore, to recover the data symbol value when a channel estimate $\hat{H}[k]$ is available, all we have to do is divide each DFT output $Z[k]$ from $k = -N/2$ to $N/2 - 1$ by its corresponding channel coefficient $\hat{H}[k]$.

$$\hat{a}[k] = \frac{Z[k]}{\hat{H}[k]} \quad (9.14)$$

This is the fundamental concept of equalization in an OFDM system. That comes out to be N divisions for N symbols, or **1 division/symbol**, a huge computational saving over a time domain equalizer for a similar frequency selective channel.

Note 9.3 The sliced bread

The OFDM operation is very similar to processing a bread. Long ago, each time a person wanted to eat bread, they had to take a knife and cut a piece of bread for themselves. Then came sliced bread in July 1928 invented by a jeweler Otto Frederick Rohwedder that changed everything. Processing each individual slice got much easier; you could put jam, butter or cheese on different slices, see Figure 9.19.

It was difficult to process the whole bread before that invention. Similarly, it is difficult to process the collective spectrum for communication purposes before OFDM came on the scene. By slicing the spectrum, OFDM not only made it easier to equalize the wireless channel but also made it possible to send different modulation signals on different subcarriers, e.g., subcarriers experiencing a 'good' channel can be used to transmit a higher-order modulation signal (e.g., 16 or 64-QAM) that translates into more bits within the same time. It also made possible to assign different subcarriers to different users for transmission or receive different subcarriers from different users by instructing them beforehand. This kind of spectral slicing and manipulation helped in adoption of OFDM over many transmission schemes, even ahead of those with a slightly superior performance in some aspects.

On a lighter note, now we have a formal proof that OFDM is the best thing that happened since sliced bread.

In summary, what OFDM does in frequency domain is fairly simple. It just segments the available bandwidth of a frequency selective channel into **many parallel**



Figure 9.19: Just like a whole bread needs to be sliced for eating convenience, OFDM slices the spectrum for communication convenience. Compare with Figure 9.18

frequency flat channels through utilizing those sinusoidal subcarriers. This is because the segmentation factor N is chosen such that the bandwidth of these frequency flat channels Δ_F is within the coherence bandwidth B_C of the channel. From Eq (9.8) and using $\Delta_F = 1/T_u$,

$$\frac{1}{T_u} < \frac{1}{T_{\text{Del}}} \quad \Rightarrow \quad \Delta_F < B_C \quad (9.15)$$

Hence, equalization for each narrow slice requires just a single division operation, rendering the computational load of the equalizer to a total of N divisions. A relevant analogy is that of a train versus a fleet of trucks. While the train carriages might look quite similar, they are units in one complete set but the fleet of trucks comes with an independence of handling and manipulating each truck in an individual manner.

9.3 An OFDM Transceiver

Having covered the fundamental concepts, we now discuss the Tx and Rx structure for an OFDM system. We will see that it is very different to a Tx and Rx structure for a single-carrier system described in detail in Chapter 3. The most interesting thing to observe is the reversal of roles in time and frequency domains which will allow us to later exploit the time frequency duality for construction of OFDM synchronization algorithms.

Tx and Rx Structure

With all the background information in place, we can easily draw a block diagram for the Tx and Rx structure of an OFDM system shown in Figure 9.20. The information bits are mapped to the symbols in accordance with the chosen modulation scheme, e.g., PSK or QAM. Next, the symbols are grouped in blocks of N by a serial-to-parallel converter and an N -point iFFT is taken to generate the time domain waveform of the complex signal. The last N_{CP} samples of this time domain waveform are prepended back to the start of the sequence as a Cyclic Prefix (CP) for both as a guard interval and for converting the linear convolution with the channel into circular convolution. A parallel-to-serial converter then produces the serial sequence that is upconverted to the carrier frequency with the help of a Local Oscillator (LO). This signal is input to a DAC that outputs the continuous-time OFDM waveform.

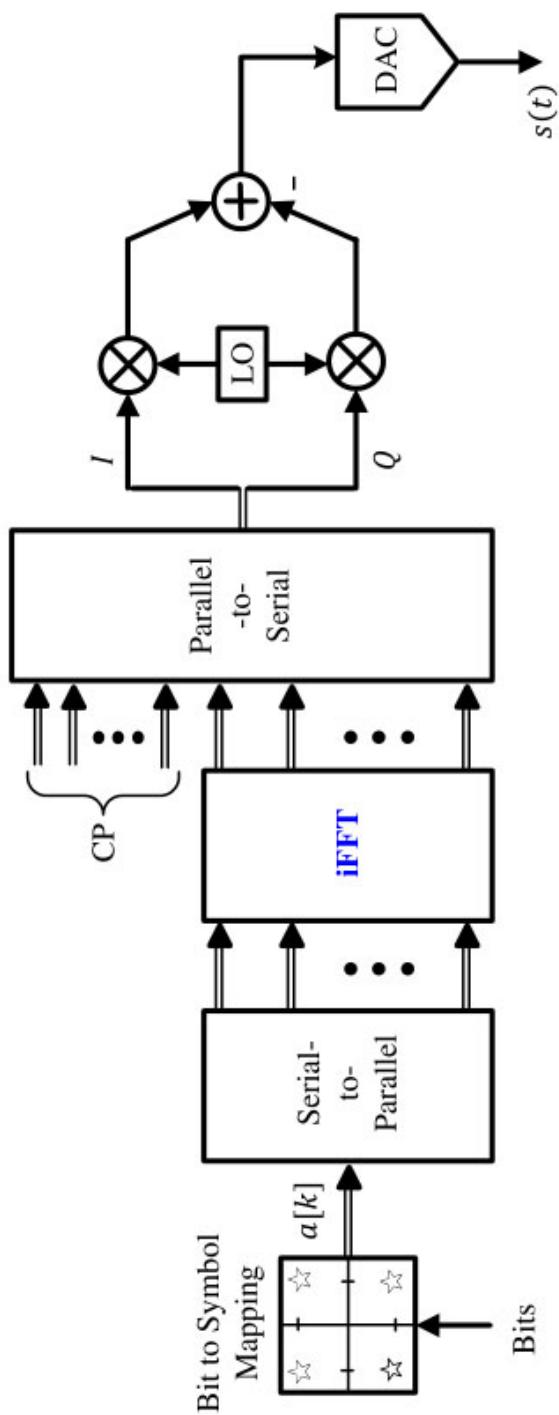


Figure 9.20: Generation of an OFDM signal using an N -point iFFT

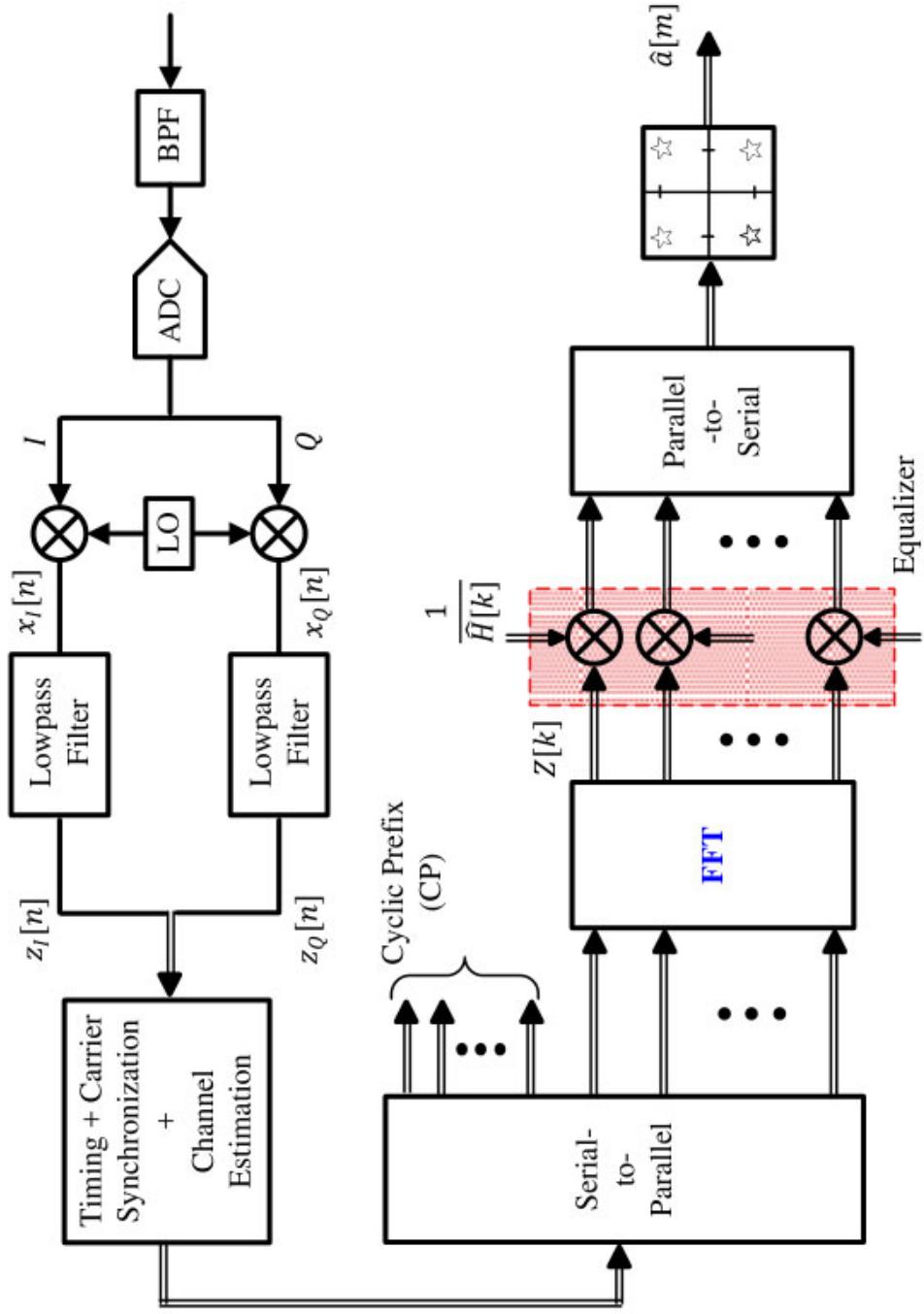


Figure 9.21: Detection of an OFDM signal using an N -point FFT before equalization

A block diagram of such a Rx is drawn in Figure 9.21. The OFDM Rx implements similar operations as the Tx but in reverse order. A continuous-time waveform is first bandpass filtered and downconverted with the help of a Local Oscillator (LO) where the choice of the ADC placement depends on the Rx architecture. This is lowpass filtered to reduce interference and noise as well as for sample rate conversion. Timing synchronization, carrier synchronization and channel estimation are performed at this stage. Then, a serial-to-parallel converter prepares the input for DFT computation. However, the first N_{CP} samples are first discarded since they are contaminated from the ISI of the previous block. Next, an N -point DFT is taken by using its efficient version, namely FFT, and multiplied with the channel inverse $1/\hat{H}[k]$ to equalize in frequency domain. Since the data symbols $a[k]$ were produced in frequency domain (which is the same as saying that an iFFT has already been taken) at the Tx, the equalized signal is ready for symbol detection which is parallel-to-serial converted and input to the decision device.

Turning to an example, we can now understand the relevant parameters of the well known IEEE 802.11a standard.

Example 9.1

In IEEE 802.11a standard, there are $N = 64$ subcarriers in total that can be scaled by the modulation symbols. In fact, only 48 out of 64 subcarriers are used for data transmission.

$$k = \{-26, \dots, -22, -20, \dots, -8, -6, \dots, -1, \\ +1, \dots, +6, +8, \dots, +20, +22, \dots, +26\}$$

Out of the remaining 16 subcarriers, 4 are used as pilot symbols for synchronization and channel tracking purposes (usually transmitted with a boosted power) while nothing is transmitted on 12 subcarriers. Subcarrier 0 is not used due to the DC offset problem described in Chapter 10 whereas 6 subcarriers on the left and 5 subcarriers on the right are used as guard bands to ease spectral and filtering requirements (recall that an OFDM signal consists of sinc signals in frequency domain which have a slow rolloff). In summary, the data modulated subcarriers are as below^a.

The subcarriers $\{-21, -7, +7, +21\}$ are used for inserting pilot subcarriers. After all these assignments, the subcarrier arrangement is shown in Figure 9.22.

After the mapping, an iDFT (or iFFT) is taken to convert the subcarriers to time domain, a cyclic prefix is inserted and finally the resulting waveform is truncated to a single OFDM symbol length by applying time domain windowing. The timing parameters of such an OFDM signal are given in Table 9.1.

^aThese subcarriers for later generations of 802.11 systems kept increasing for higher data rates. For example, an 802.11n OFDM signal contains 28 subcarriers on each side of the spectrum in a 20 MHz channel and 57 subcarriers on each side in a 40 MHz channel. On the other hand, 802.11ac utilizes 121 subcarriers on each side for an 80 MHz channel.

A Comparison with SC-FDE

Recall that in case of Single-Carrier Frequency Domain Equalization (SC-FDE) discussed in Section 8.3.8, both the FFT and iFFT were performed at the Rx. In an OFDM system on the other hand, the iFFT is implemented at the Tx and the FFT at the Rx. The respective block diagrams of the two systems are drawn in Figure 9.23.

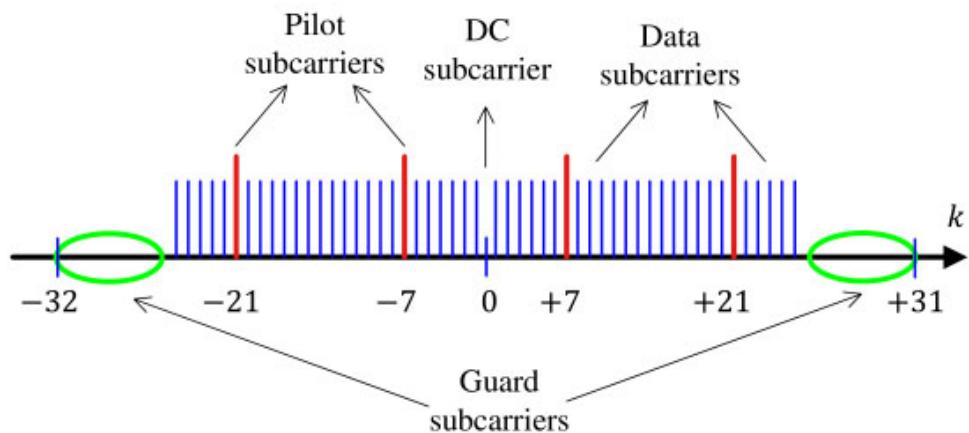


Figure 9.22: Mapping of the DC, pilot, data and guard subcarriers in IEEE 802.11a

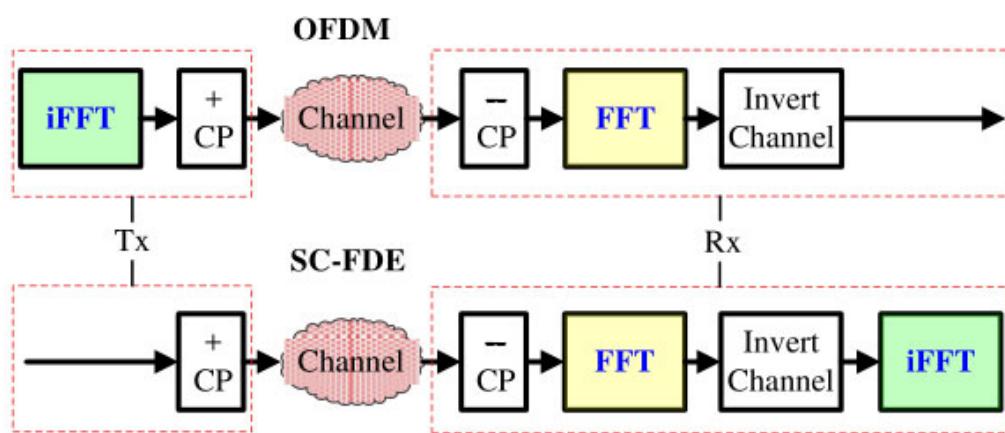


Figure 9.23: A block diagram comparison of an OFDM and an SC-FDE system

Table 9.1: Timing for IEEE 802.11a OFDM System

Parameter	Value
Bandwidth $B = 1/T_S$	20 MHz
Data subcarriers	48
Pilot subcarriers	4
Null subcarriers	12
Total subcarriers	$48 + 4 + 12 = 64$
Subcarrier spacing $\Delta_F = B/N$	$20\text{MHz}/64 = 312.5 \text{ kHz}$
Useful symbol duration (FFT period) $T_u = 1/\Delta_F$	$3.2\mu\text{s}$
Cyclic prefix length T_{CP} (25% of T_u)	$T_u/4 = 0.8\mu\text{s}$
Total symbol duration T_{OFDM}	$3.2 + 0.8 = 4\mu\text{s}$

The overall effect of this change is that OFDM transmits the symbols on orthogonal subcarriers whereas an SC-FDE system spreads the information of each data symbol across all subchannels, thus providing a little extra frequency diversity. Consequently, once a subcarrier in OFDM gets in a deep fade, the information about that data symbol is completely gone and noise enhancement varies with different subcarriers. On the other hand, an SC-FDE Rx still gathers some of that information from other subchannels and each symbol sees the same effective noise variance. Hence, the performance of an SC-FDE system is slightly better than an OFDM system in general.

This remains true in an uncoded system. Channel coding is a process in which extra redundancy is added to the information bits to make them more resilient to errors occurring in the transmission process. It turns out that in a system with channel coding and interleaving included, the performance of both systems is comparable to each other.

In other aspects, OFDM is more sensitive to RF impairments than SC-FDE systems. For example, it is sensitive to power amplifier non-linearities because the ratio between the peak of the OFDM signal and its average value – known as *peak to average power ratio* – is higher. Furthermore, OFDM systems suffer more in the presence of a carrier frequency offset and phase noise as compared to single-carrier systems.

Why is OFDM still the dominant technology for high rate wireless communication so far when an SC-FDE Rx can also be implemented? Besides the fact that OFDM balances the computational load through an iFFT at the Tx, the flexibility of having a sliced spectrum at hand open to manipulation of any kind cannot be overlooked. As stated before, a different modulation and coding scheme can be allocated to each

subcarrier based on their individual channel gains. In addition, different users can be assigned a set of subcarriers to perform multiplexing known as Orthogonal Frequency Division Multiple Access (OFDMA).

It should be noted that although SC-FDE was proposed decades earlier, it was revived in 1995 by an article from Hikmet Sari in [40]. I must add that *the rediscovery of SC-FDE did take the original magic away from the beauty of OFDM transmission.* When the idea of OFDM was treated as a new revolution, Sari's article slightly shook this excitement and told us that we already had something equivalent we ignored during all this time.

This is the time we briefly touch on the concept of pulse shaping in OFDM systems.

Pulse Shaping in OFDM Systems

This discussion heavily relies on the excellent article by Keysight Technologies in Ref. [41]. Originally, OFDM is a block processing scheme and hence there is no extra pulse shaping required after the iDFT at the Tx. Instead, a certain number of subcarriers, both on the positive and negative edges of the spectrum, are left unmodulated to cater for the decay of sinc sidelobes thus alleviating the subsequent digital and analog filtering tasks. In Example 9.1 on IEEE 802.11a standard, 6 subcarriers on the negative edge and 5 subcarriers on the positive edge of the spectrum were assigned as null subcarriers.

Nevertheless, many systems still employ pulse shaping in OFDM systems to smooth the edges at the OFDM symbol boundaries that reduces the out of band power to meet the regulatory requirements. Recall that the pulse shape employed in a single-carrier system is a Raised Cosine in frequency domain. Therefore, from time frequency duality, we already know that the Raised Cosine pulse shape in an OFDM system is in time domain. We have covered the topic of pulse shaping in depth in Section 3.6 and hence briefly describe this operation in the current context.

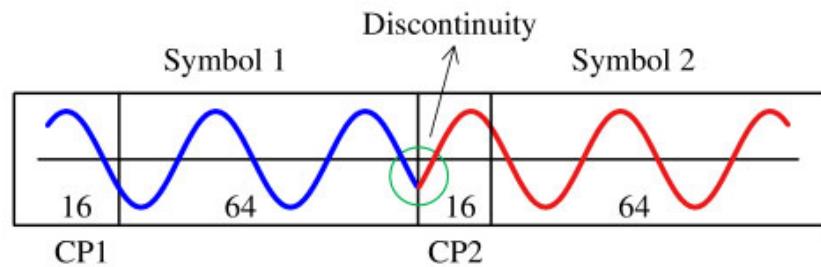
We created the Cyclic Prefix (CP) by appending the last N_{CP} samples from the iFFT output to the beginning of the OFDM symbol. The total symbol duration thus becomes $N + N_{CP}$ samples. For obvious reasons, the next OFDM symbol begins with a different amplitude and phase at the boundary of the current symbol, as shown in Figure 9.24a for $N = 64$ and $N_{CP} = 16$. This discontinuity causes spectral regrowth that violates the spectral mask and creates interference for adjacent channels. Therefore, our task is to assemble consecutive OFDM symbols in a smooth fashion.

To introduce a smooth transition between the current and next OFDM symbol, a cyclic suffix (also known as cyclic postfix) is introduced. Just like a cyclic prefix is the last N_{CP} samples in an N sample symbol, a cyclic suffix is the first N_{CS} samples out of N that are attached at the end.

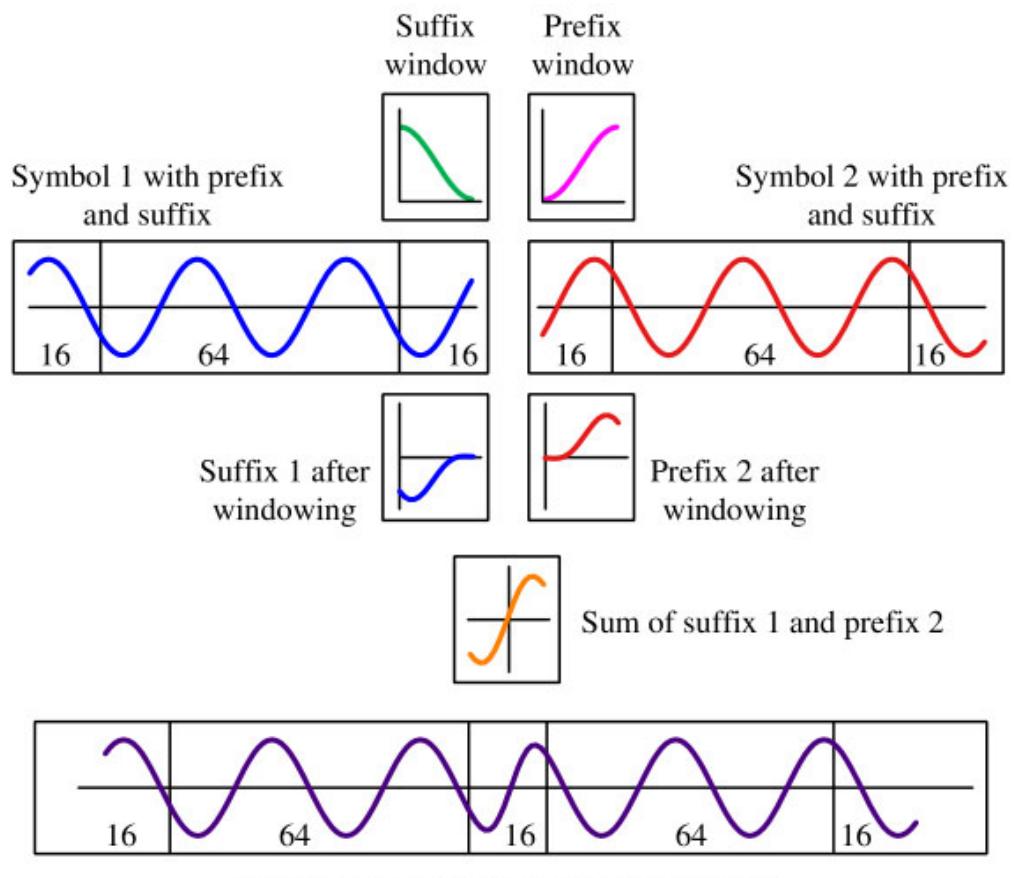
Let us create an example for $N = 64$ and $N_{CP} = 16$. The total number of samples in this OFDM symbol is equal to $N + N_{CP} = N_{OFDM} = 80$ as in Figure 9.24a and it should remain the same after the windowing operation. Here, we assume a full length cyclic suffix of length $N_{CS} = 16$ samples to understand the concept.

When the first 16 samples of OFDM symbol 1 are appended in the end as a cyclic suffix, the symbol becomes 96 samples long which violates the $N_{OFDM} = 80$ sample condition of the original signal. Instead, an alternative scheme is applied as follows for which a graphical representation is illustrated in Figure 9.24b.

- Length 16 cyclic suffix of symbol 1 is windowed with the transition part of a Raised Cosine that rolls off from 1 to 0.



(a) Discontinuity arising without pulse shaping



(b) Pulse shaping produces a smooth waveform

Figure 9.24: At the boundary of two OFDM symbols, pulse shaping smoothes the edges, resulting in the avoidance of spectral regrowth

- Length 16 cyclic prefix of symbol 2 is windowed with the transition part of the Raised Cosine that rolls on from 0 to 1.

- These two segments are overlapped and summed together to generate a length 16 portion that is attached between both symbols.

Notice that the waveform is now continuous going from one symbol to the next while the total OFDM symbol duration is still 80.

Recall that one purpose of the cyclic prefix is to convert linear convolution with the channel into circular convolution. This circular convolution is a result of the periodicity that appears due to the cyclic prefix being the same as the last part of the symbol. However, now a full length cyclic suffix of the first symbol interferes with the cyclic prefix of the next symbol, thus leaving no room for the periodicity to appear.

The solution is to reduce the length of the window, i.e., cyclic suffix, to as less number of samples as possible. For the current example, a length 4 cyclic suffix implies that the multipath immunity has reduced from $N_{CP} = 16$ samples to $N_{CP} - N_{CS} = 12$ samples. Only a maximum channel length of $N_{Tap} = 12$ can now induce a single tap flat fading at each subcarrier of the Rx DFT output. This is the price we pay for the spectral smoothing of the waveform.

A variant of this technique is known as Windowed-OFDM (W-OFDM) in which windowing is applied in a manner that the window overlaps with the complete OFDM symbol as well as the CP and the cyclic suffix. Moreover, the window amplitude is -3 dB at the start and end of the OFDM symbol. For a perfect reconstruction of the OFDM symbol, the cyclic suffix and cyclic prefix are not discarded and instead are added to the start and the end, respectively.

OFDM Time-Frequency Grid

In single-carrier systems, the frequency domain is not defined in an isolated sense as in OFDM systems. Instead, the spectrum of the signal contains contributions from all the modulation symbols. From the discussion so far, we come to know that OFDM is particularly attractive due to its ability to treat the frequency domain in a sliced manner, thanks to the orthogonal subcarriers. This gives rise to the concept of time-frequency grid where the index k defines individual subcarriers on the frequency axis while the index m defines successive OFDM symbols on the time axis. This is plotted in Figure 9.25 where the indices m and k lie on horizontal and vertical axis, respectively.

The concept of time-frequency grid brings much freedom in manipulating the frequency axis. For example, it makes tasks such as synchronization in general and channel estimation in particular quite straightforward. Furthermore, interpolation between those subcarriers produces reliable channel estimates for the whole spectrum. This is what we will discuss in Section 9.7.

Next, we discuss the synchronization techniques for OFDM systems which is mainly influenced by Ref. [42] and [43].

9.4 Timing Synchronization

We have arrived here by covering enough scenarios of time frequency duality that synchronization issues in an OFDM system can be easily understood. The two major impairments discussed above are

- Inter-Symbol Interference (ISI) where one OFDM symbol can leak into the other OFDM symbol, and

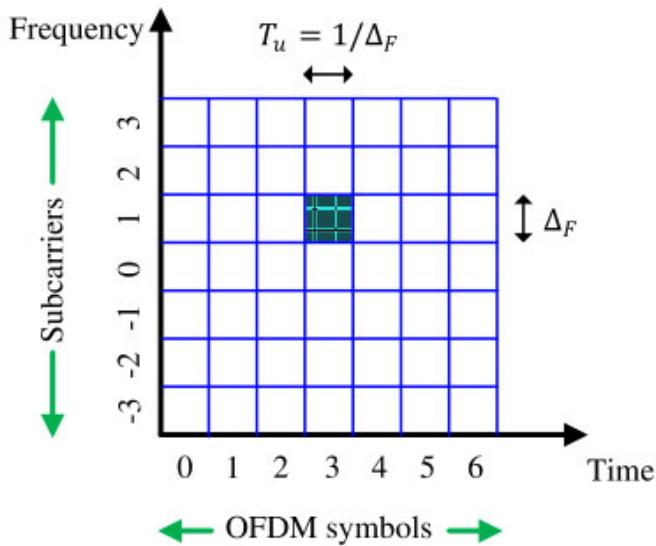


Figure 9.25: OFDM time-frequency grid

- Inter-Carrier Interference (ICI) where the orthogonality among the subcarriers is destroyed.

While we have discussed that ISI can happen when the channel length is longer than the CP, it can also happen if we choose the wrong window of N samples before taking the DFT at the Rx, i.e., misalignment in timing synchronization. Although this same phenomenon also causes ICI, the subcarriers also suffer from ICI due to a misalignment in carrier frequency synchronization (or to a minor extent, sampling clock frequency synchronization).

The choice between a feedforward and a feedback implementation depends on the performance requirements and system specifications. By virtue of block processing in OFDM, feedforward approaches naturally complement the burst mode transmissions. On the other hand, most continuous transmissions like Digital Audio Broadcasting (DAB) and Digital Video Broadcasting (DVB) can implement a closed loop approach. There are also many hybrid techniques where the acquisition is done in a feedforward manner while the pilot-based estimates are used to drive loops that slowly correct the radio impairments.

In this chapter, we focus on feedforward techniques, as feedback methods such as a PLL, FLL and TLL are discussed in enough details in the previous chapters and the reader can easily carry over similar feedback techniques from single-carrier to OFDM systems.

Next, we discuss the effects of a timing impairment without help of many equations.

Effect of Timing Mismatch

To avoid using many indices, we skip the OFDM symbol index m in the following expressions to focus on each OFDM symbol. Moreover, upconversion and downconversion through a carrier at frequency F_C is also bypassed and complex baseband

representations are employed. Finally, no additive white Gaussian noise is present to study the effects of each impairment in isolation.

With these settings in place, a baseband OFDM signal at the Tx is given in Eq (9.4) and reproduced below.

$$\begin{aligned} I \rightarrow \quad v_I[n] &= \frac{1}{N} \sum_{k=-N/2}^{N/2-1} \left[a_I[k] \cos 2\pi \frac{k}{N} n - a_Q[k] \sin 2\pi \frac{k}{N} n \right] \\ Q \uparrow \quad v_Q[n] &= \frac{1}{N} \sum_{k=-N/2}^{N/2-1} \left[a_Q[k] \cos 2\pi \frac{k}{N} n + a_I[k] \sin 2\pi \frac{k}{N} n \right] \end{aligned} \quad (9.16)$$

Keep in mind that the data symbols $a[k]$ arise from a linear modulation scheme in exactly the same manner as in single-carrier systems but are assumed to be in frequency domain here due to the iFFT operation involved.

After the DAC, what is the form of the continuous-time signal? Recall the expression for subcarriers in continuous domain in Eq (9.2) given by $2\pi F_k t$ where each subcarrier frequency F_k in Hz is given by

$$F_k = \frac{k}{NT_S} = \frac{k}{T_u} = k\Delta_F \quad \text{for each } k = -N/2, \dots, N/2 - 1$$

where we have used $k/N = F/F_S$. The above relation reinforces the concept that as the subcarrier frequencies F_k should be integer multiples of subcarrier spacing Δ_F , see Figure 9.16a. Now using $(k/N)n = (F_k/F_S)n = F_k n T_S$ which translates to $F_k t$ in continuous time domain, the OFDM signal in continuous-time becomes

$$\begin{aligned} I \rightarrow \quad v_I(t) &= \frac{1}{N} \sum_{k=-N/2}^{N/2-1} [a_I[k] \cos 2\pi F_k t - a_Q[k] \sin 2\pi F_k t] \\ Q \uparrow \quad v_Q(t) &= \frac{1}{N} \sum_{k=-N/2}^{N/2-1} [a_Q[k] \cos 2\pi F_k t + a_I[k] \sin 2\pi F_k t] \end{aligned}$$

The spectrum of such an OFDM signal is drawn in Figure 9.26. Notice the rectangular structure within the spectrum and sinc like sidelobes at the edges. To constrain the spectrum, a time domain Raised Cosine (RC) pulse shape can also be used which was described earlier.

With zero noise, the same signal is received at the Rx convoluted by the channel and the baseband version, following our conventions from the previous chapters, is denoted by $x(t)$. Before we sample $x(t)$ to go to the discrete domain, we need to know how a Carrier Frequency Offset (CFO) F_Δ and a Symbol Timing Offset (STO) ε_Δ affects the Rx signal. For this purpose, let us focus on the expression $F_k t$. After the synchronization errors, this becomes

$$2\pi F_k t \longrightarrow (F_k + F_\Delta)(t + \varepsilon_\Delta)$$

As opposed to introducing a negative ε_Δ as in $x(t - \varepsilon_\Delta)$ done in single-carrier systems before, there is a reason to choose a positive ε_Δ here that will become clear in the discussion on timing synchronization. Now after sampling at a rate of $F_S = 1/T_S$, we

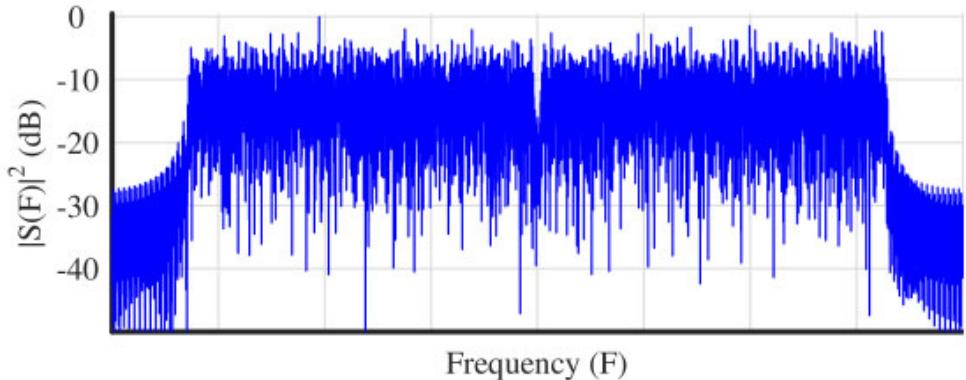


Figure 9.26: Spectrum of a typical OFDM signal

can write

$$\begin{aligned}
 2\pi(F_k + F_\Delta)(t + \varepsilon_\Delta)|_{t=nT_S} &= 2\pi(F_k + F_\Delta)(nT_S + \varepsilon_\Delta) \\
 &= 2\pi(F_k + F_\Delta)\left(n + \frac{\varepsilon_\Delta}{T_S}\right)T_S \\
 &= 2\pi\frac{F_k + F_\Delta}{F_S}\left(n + \frac{\varepsilon_\Delta}{T_S}\right) \\
 &= 2\pi\left(\frac{F_k}{F_S} + \frac{F_\Delta}{N\Delta_F}\right)\left(n + \frac{\varepsilon_\Delta}{T_S}\right) \\
 &= \frac{2\pi}{N}\left(k + \frac{F_\Delta}{\Delta_F}\right)\left(n + \frac{\varepsilon_\Delta}{T_S}\right)
 \end{aligned}$$

where we have used $F_k = k\Delta_F$ and

$$\frac{1}{\Delta_F} = T_u = NT_S = \frac{N}{F_S} \Rightarrow F_S = N \cdot \Delta_F$$

Also, F_Δ/Δ_F [†] and ε_Δ/T_S are normalized CFO and STO, respectively, as encountered before in the context of single-carrier systems. We denote them by

$$v_0 = \frac{F_\Delta}{\Delta_F} \quad \varepsilon_0 = \frac{\varepsilon_\Delta}{T_S}$$

With these settings in place, the above expression can be modified as

$$2\pi(F_k + F_\Delta)(t + \varepsilon_\Delta)|_{t=nT_S} = \frac{2\pi}{N}(k + v_0)(n + \varepsilon_0) \quad (9.17)$$

Now we can use the above relation to get the sampled baseband version of the Rx signal $x[n]$ which after lowpass filtering as in Figure 9.21 produces the signal $z[n]$. Skipping the mathematical details of this process, all we need to do is include the

[†]As a reminder, F_Δ here is the CFO and Δ_F is the subcarrier spacing.

product of a channel coefficient $H[k]$ with data symbol $a[k]$ for each subcarrier k into Eq (9.16). Defining $Z[k] = a[k]H[k]$ from Eq (9.13),

$$I \rightarrow z_I[n] = \frac{1}{N} \sum_{k=-N/2}^{N/2-1} \left[Z_I[k] \cos \frac{2\pi}{N} (k + v_0)(n + \varepsilon_0) - Z_Q[k] \sin \frac{2\pi}{N} (k + v_0)(n + \varepsilon_0) \right] \quad (9.18a)$$

$$Q \uparrow z_Q[n] = \frac{1}{N} \sum_{k=-N/2}^{N/2-1} \left[Z_Q[k] \cos \frac{2\pi}{N} (k + v_0)(n + \varepsilon_0) + Z_I[k] \sin \frac{2\pi}{N} (k + v_0)(n + \varepsilon_0) \right] \quad (9.18b)$$

For a zero CFO and STO, the above equation is just an iDFT[†] of $Z[k]$. Hence after the DFT at the Rx, we get our $Z[k]$ back which are the modulation symbols $a[k]$ scaled by the channel coefficients $H[k]$.

We conclude that the timing synchronization problem in OFDM systems is slightly different than single-carrier systems. There, a symbol timing offset is the fractional interval between 0 to symbol time T_M due to the necessity of just 1 sample/symbol for symbol detection. Here, while the sample/symbol is still 1, these symbols are grouped into N samples of an OFDM symbol. Consequently, there is an integer part of the Symbol Timing offset (STO) and a fractional part.

Integer STO: The integer part of the STO problem corresponds to locating the right sample. What exactly is the right sample? A major portion of the processing at the Rx is composed of taking the N -point DFT (actually, FFT) of the incoming signal. For this purpose, the Rx must know the starting sample of the sequence which will form the input to this FFT routine. This samples coincides with the end of the CP and constitutes of the useful portion of the OFDM symbol T_u . As a result, all N samples participate in estimating the location of the right sample. This availability of N samples/OFDM symbol implies that the (OFDM) symbol timing synchronization problem is quite similar to the frame synchronization part in the single-carrier systems. This is shown in Figure 9.27 where the boundaries of the OFDM symbols in an OFDM frame are drawn for clarification purpose. The task of the timing synchronization unit is to locate these boundaries.

Fractional STO: As far as the fractional part of the STO is concerned, it is a bit surprising but we do not really need to estimate it. Recall from Section 1.9 that a delay in time domain induces a *frequency dependent phase shift* in frequency domain.

$$\text{Time shift } s[(n \pm \varepsilon_0) \bmod N] \longrightarrow \pm 2\pi \frac{k}{N} \varepsilon_0 \quad \text{Phase shift} \quad (9.19)$$

Consequently, after the DFT, the fractional timing offset appears as a phase rotation at each individual subcarrier k and *hence becomes part of the phase of*

[†]If you are familiar with complex notations in OFDM, you can recognize this equation as

$$z[n] = \frac{1}{N} \sum_{k=-N/2}^{N/2-1} a[k]H[k] \exp \left(j \frac{2\pi}{N} (k + v_0)(n + \varepsilon_0) \right)$$

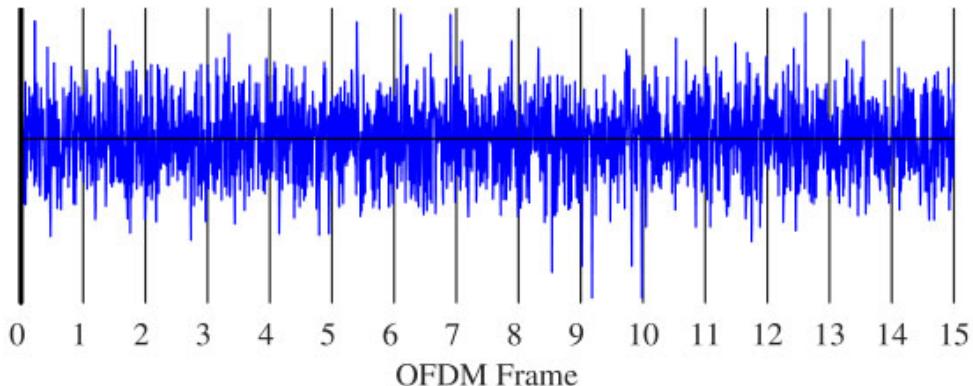


Figure 9.27: The task of the timing synchronization unit is to locate the OFDM symbol boundaries (i.e., their starting samples) in an OFDM frame

the channel gain $H[k]$. Since we were not slicing the spectrum in separate subcarriers in single-carrier systems, we could not apply this result there for rate $1/T_M$ sampling. But the timing phase offset does become part of the channel phase for a fractionally-spaced equalizer, enabling it to compensate for this timing phase. A timing recovery loop, however, is still required to track the waveform periodicity in the form of clock frequency.

The question now is the following: how far away the timing offset can be from the ideal sampling instant such that no ISI or ICI is contributed in the DFT input. For this purpose, refer to Figure 9.28 that shows two OFDM symbols in succession.

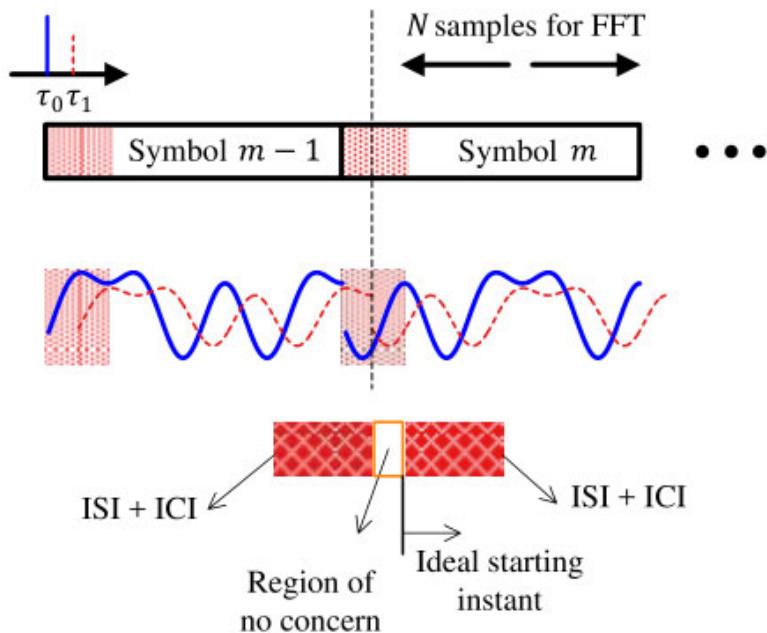


Figure 9.28: The effect of STO ε_0 on an OFDM symbol

- The ideal sampling instant is the sample where the CP ends. However, even if our first sample is behind this location, there is no harm done as long as it lies after the contamination from the previous symbol (i.e., beyond the channel length). Carefully look at the waveform portion of Figure 9.28. In this region of no concern, *only the multipath copies of the same subcarrier* are being summed up for each subcarrier and this can only induce a change in phase. This change in phase merges into the channel phase $\angle H[k]$ which is compensated by the equalizer later.

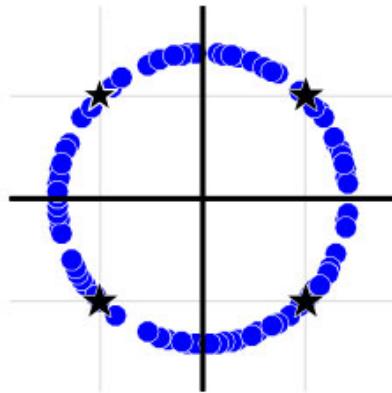
If the STO ε_0 is within this region, and a DFT is computed at the Rx, we know from our knowledge of time frequency duality that the effect of a timing error on an OFDM symbol should cause a spinning of the symbols defined in frequency domain, just like a CFO in a single-carrier systems rotated the constellation of our symbols defined in time domain, see Figure 6.3. Therefore, after the DFT and owing to Eq (9.19), the Rx symbols will be rotating in circles with an inverse period ε_0 (whether it is an integer or a fraction), as long as ε_0 is within the region of no concern. This is drawn in Figure 9.29a.

To confirm our intuition, we manipulate the argument of the subcarriers at the Rx FFT input in Eq (9.17), reproduced below, as follows.

$$\begin{aligned} 2\pi(F_k + F_\Delta)(t + \varepsilon_\Delta)|_{t=nT_S} &= \frac{2\pi}{N} (k + \nu_0) (n + \varepsilon_0) \\ &= \frac{2\pi}{N} \left\{ \underbrace{kn}_{\text{Term 1}} + \underbrace{\nu_0 n}_{\text{Term 2}} + \underbrace{k\varepsilon_0}_{\text{Term 3}} + \underbrace{\nu_0 \varepsilon_0}_{\text{Term 4}} \right\} \end{aligned}$$

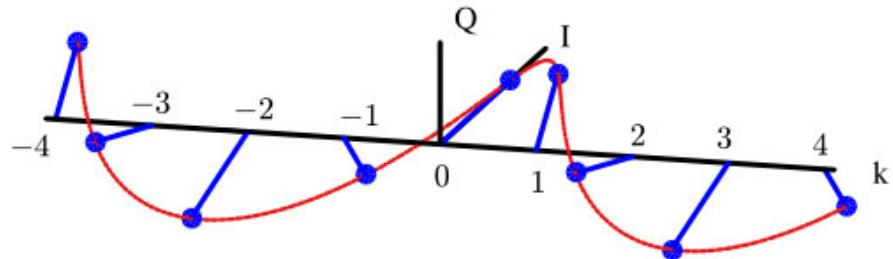
- From the DFT definition in Eq (1.53), we know that the FFT at the Rx consists of subcarriers with the arguments $-kn$ which simply cancels out Term 1 above.
- Term 2 appears as a normalized CFO at the FFT output because it is a function of time n .
- Term 4 is neither a function of n nor a function of k . Hence, it appears as a constant phase that gets merged into the channel phase $\angle H[k]$.
- This leaves Term 3 as a function of STO and clearly it is being multiplied with the subcarrier index k . As a result, the data symbols at each subcarrier k get rotated by the product of this index k varying from $-N/2$ to $N/2 - 1$ with the STO ε_0 and this is what causes the constellation rotation. In other words, these data symbols are multiplied with a complex sinusoid $2\pi(k/N)\varepsilon_0$ with inverse period ε_0 and we see this complex sinusoid rotating in frequency domain in Figure 9.29b, which is exactly the same as drawn in Figure 9.29a in the form of a scatter plot.

- Referring back to Figure 9.28, when the starting sample of the OFDM symbol is before the region of no concern, the subcarriers from the previous OFDM symbol interfere with the current OFDM symbol inducing both ISI and ICI.
- Finally, when the input window is after the ideal sampling instant, then a similar ISI and ICI come from the next OFDM symbol. If you are not clear on why ICI occurs in such a case, refer to Figure 9.11.



(a) A small STO in the DFT input spins the DFT constellation output in circles. Compare this scatter plot with that in Figure 6.3a

OFDM Data Symbols with a Timing Offset



(b) Frequency domain representation of the data subcarriers are modulated with a complex sinusoid with inverse period ε_0

Figure 9.29: Effect of a timing offset ε_0 on the data subcarriers: scatter plot and frequency domain subcarriers

Having established the necessity to have synchronization in place, we discuss some of the relevant estimation techniques. In the single-carrier systems, our Master Algorithm, the correlation, was mostly employed in an implicit manner in the sense that the fundamental expression was derived from the correlation and then various approximations were found to estimate the desired parameter. In the case of OFDM, the correlation is overwhelmingly utilized in an explicit manner for time domain estimation of synchronization parameters. The discussion is mostly based on the synchronization techniques in Ref. [44].

Training Correlation

First, consider the correlation of a pseudo-random sequence of +1s and -1s with itself (i.e., autocorrelation) for several time shifts[†]. The summation in the correlation of such a long stream of opposite signs will mostly cancel out and the resultant value will be large only when they exactly coincide. From Section 2.8, we know that the autocorrelation of white noise is a unit impulse which was drawn in Figure 2.39. When we draw the autocorrelation of a pseudo-random sequence in Figure 9.30 for a range of negative and positive time shifts, we can see that such a sequence approximately behave like noise. The important result is that *the autocorrelation of this sequence has a sharp peak when completely aligned*, much higher than any other time shift.

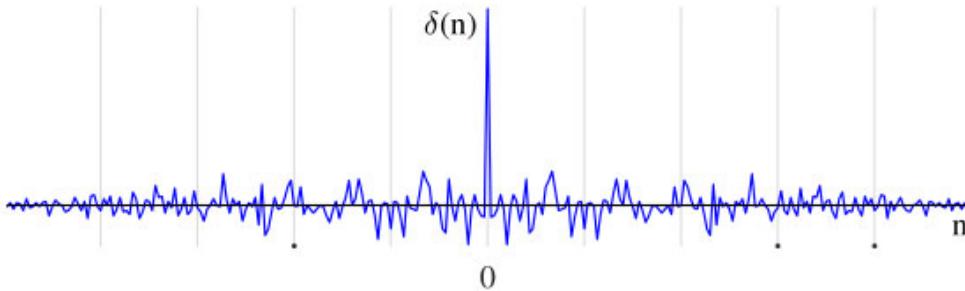


Figure 9.30: Autocorrelation of a pseudo-random sequence is similar to an impulse, just like white noise

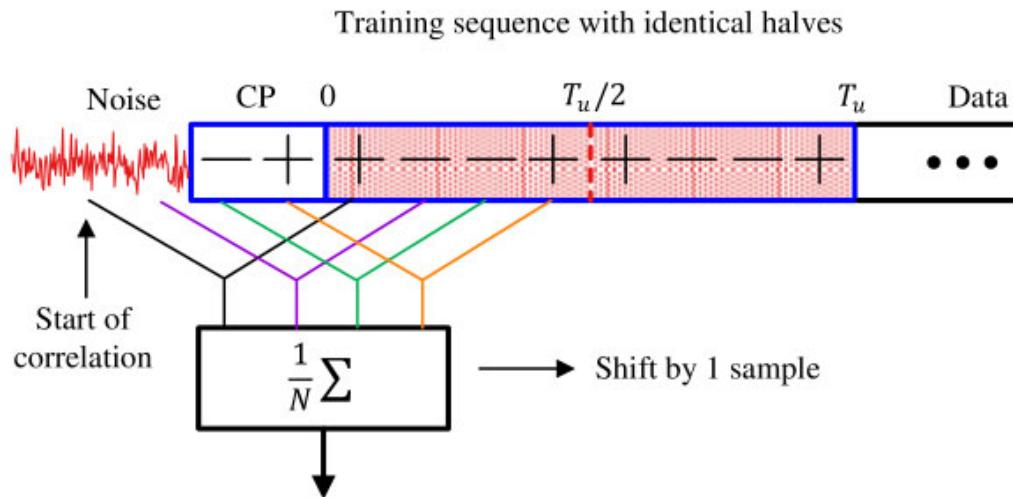
Second, two copies of such a pseudo-random sequence is prepended before the data symbols without any iFFT at the Tx, i.e., the training sequence is in time domain. The Rx is sampling the signal in the expectation of the OFDM frame. Here, the initial samples are composed of just noise, followed by the above training sequence and then the OFDM symbols as illustrated in Figure 9.31a. We choose an initial sample and compute the correlation as follows. If a sample in $z[n]$ is multiplied with the conjugate of another sample spaced $N/2$ apart (i.e., $T_u/2$ seconds), and $N/2$ such multiplications are carried out and summed together, then this will yield a large value only when this starting sample coincides with the start of the training sequence. An example of such a correlation is shown in Figure 9.31 for a length $N = 4$ training sequence composed of $\{+, -, -, +\}$.

- In Figure 9.31a which illustrates a wrong location for the correlation start, assume that the two noise samples have a - and + sign, respectively. Then, the correlation result here (normalized by $1/N$) becomes

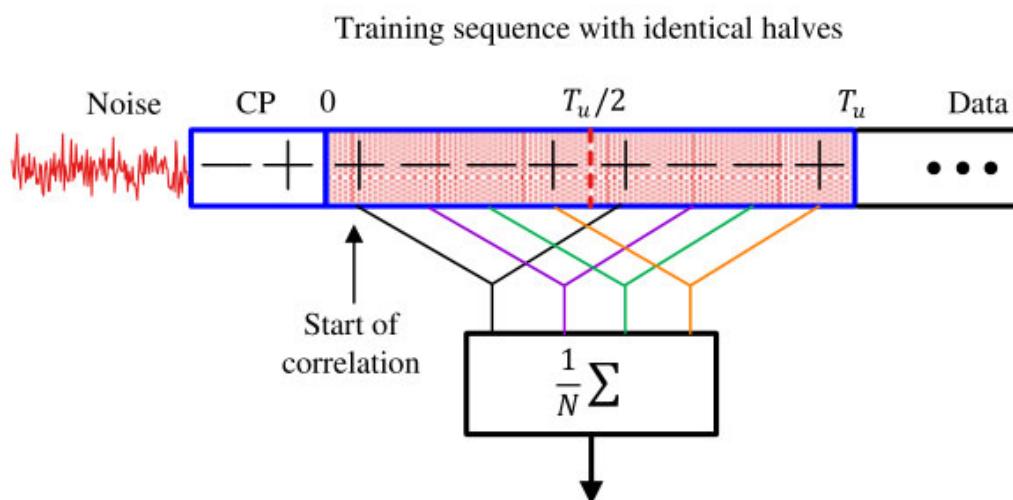
$$\frac{1}{4} \sum \{ (- \cdot +), (+ \cdot -), (- \cdot -), (+ \cdot +) \} = 0$$

One could argue that the noise samples could have opposite signs and hence generate a large value. But remember that we are considering an example for a very small $N = 4$. For a large value of N , it is almost improbable that so many noise samples have exactly the same signs in succession as the training sequence. Furthermore, the example is taken near the training sequence where

[†]There are many specialized sequences available for this purpose such as Gold codes and Kasami codes but since optimization is not our purpose, we consider a random stream of +1s and -1s here.



(a) The correlation start is located before the time domain training



(b) The correlation start is located at the initial sample of the time domain training

Figure 9.31: A time domain training sequence with two identical halves containing a pseudo-random sequence. The correlation is a large value only when the correlation start coincides with the initial sample of the training

the last two values in the window are already the same due to the presence of a CP. This partial similarity gives rise to a gradually increasing slope of the correlation result, instead of a sharp peak.

- Next, we shift this correlation window by 1 sample to the right and compute the $N/2$ -spaced correlation among the samples falling within this window. And continue repeating this process.

- When this correlation window reaches the actual frame start, i.e., the initial sample of the training sequence, the following result is produced.

$$\frac{1}{4} \sum \left\{ (+\cdot+), (-\cdot-), (-\cdot-), (+\cdot+) \right\} = 1$$

which corresponds to the exact alignment of the length $N/2$ training sequence. Connect this result with Figure 9.30 illustrating $\delta[n]$ as its autocorrelation.

The Coarse Timing Metric

To form such a signal, we construct a single OFDM symbol that consists of the length $N/2$ iFFT of a pseudo-random sequence, i.e., the training sequence is in frequency domain now and consists of complex samples. Then, we repeat it twice in time domain which makes the second half of this OFDM symbol the same as the first half. This forms our training sequence prepended at the start of an OFDM frame consisting of two identical halves with length $N/2$ each.

Note that such a training sequence can also be generated by assigning the length $N/2$ sequence to even subcarriers and zeros to odd subcarriers. This is equivalent to upsampling by 2 the length $N/2$ sequence in frequency domain. Therefore, its length N FFT is composed of two copies in the time domain, just like upsampling by P in time domain consists of P spectral replicas in frequency domain.

After the addition of a CP, the convolution of the channel will affect both halves of the training sequence in a similar manner. The samples received are denoted as $z[n]$ and let us compute a correlation in this sequence in the following manner. Utilizing the definition of correlation, we define the index ε_0 as the sample where the correlation starts and a metric $P(\varepsilon_0)$ based on this index as

$$P(\varepsilon_0) = \sum_{i=0}^{N/2-1} z \left[\left(\varepsilon_0 + \frac{N}{2} \right) + i \right] z^* [\varepsilon_0 + i] \quad (9.20)$$

This correlation involves a window of N samples starting at ε_0 that slides to the right while searching for the training symbol. The conjugate operation is necessary to rotate the length $T_u/2$ subcarriers in the first half in an opposite direction to cancel them out and add the signs of the pseudo-random sequence in a coherent manner. This is drawn in Figure 9.32 where a length N correlation window is also visible for the ideal correlation start. Although this could be seen from plugging the value of $z[n]$ from Eq (9.18), we avoid this route to keep the discussion simple.

There are two factors that can cause a wrong timing offset if only $P(\varepsilon_0)$ is used for this purpose.

Phase/frequency rotation: Recalling that OFDM is a passband and hence complex modulation scheme and the subcarriers are composed of both I and Q components, $P(\varepsilon_0)$ in Eq (9.20) above consists of complex samples. How should we compare this metric for different values of ε_0 , through the I part, Q part or the magnitude? If there is a phase rotation between the two identical halves of the training symbol, the inphase part will not produce a correct amount of correlation. For a worst case phase offset of $\pi/2$ between them, a zero appears at the I output while all the correlation magnitude appears in the Q output. Similar is the case when a frequency offset is present in the Rx signal that spins the modulation symbols and makes the two identical halves non-identical. Observe

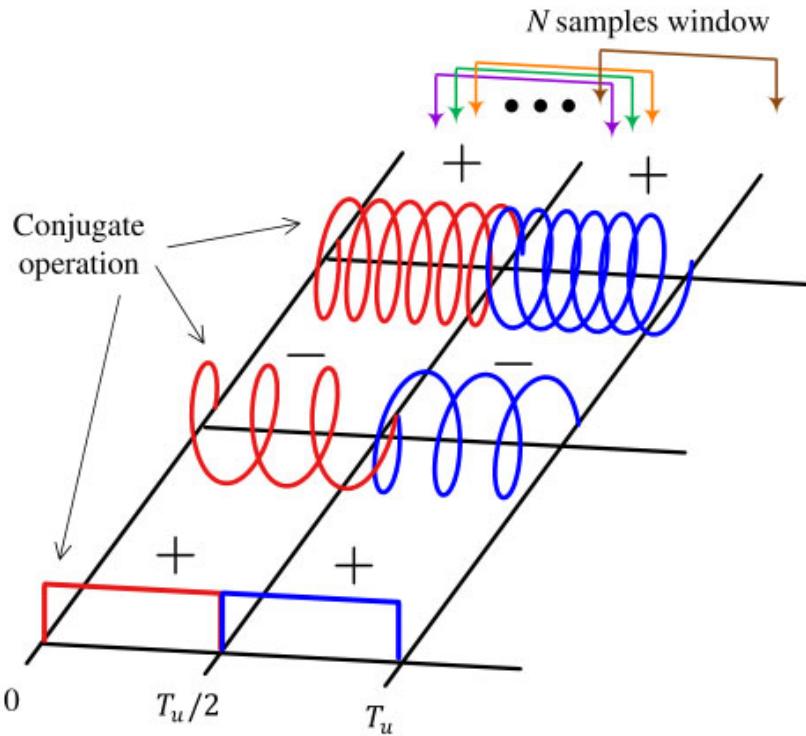


Figure 9.32: Computation of the metric $P(\varepsilon_0)$ involves a correlation sum at a time difference of $T_u/2$ seconds. Notice the opposite rotation of each subcarrier in the first half (after scaling by + or - sign)

that if this was not the case, then the best method to acquire the timing is to correlate the incoming signal (in which the training is embedded) with a locally stored copy of the training at the Rx which is free of noise and other distortions. The presence of a frequency offset ruins the correlation of two otherwise identical sequences. One simple remedy for this problem is to focus on the magnitude squared of the metric $P(\varepsilon_0)$.

Signal energy: Since the channel is not AWGN, the Rx signal energy can vary throughout the duration of the transmission. Consider a case where the energy is high in the first half and low in the second half. Consequently, the correct peak will exhibit a low value and hence can be easily missed even for the correct ε_0 . The solution here is to normalize the metric $P(\varepsilon_0)$ with the energy of the samples within the correlation window. However, each sequence in the correlation consists of $N/2$ samples, so only $N/2$ samples need to be taken into account. While energy in any of the two halves can be measured, the second half is usually chosen as a precaution against an unexpectedly long channel.

$$E(\varepsilon_0) = \sum_{i=0}^{N/2-1} \left| z \left[\left(\varepsilon_0 + \frac{N}{2} \right) + i \right] \right|^2$$

Taking into account the above two factors, a new timing synchronization metric can

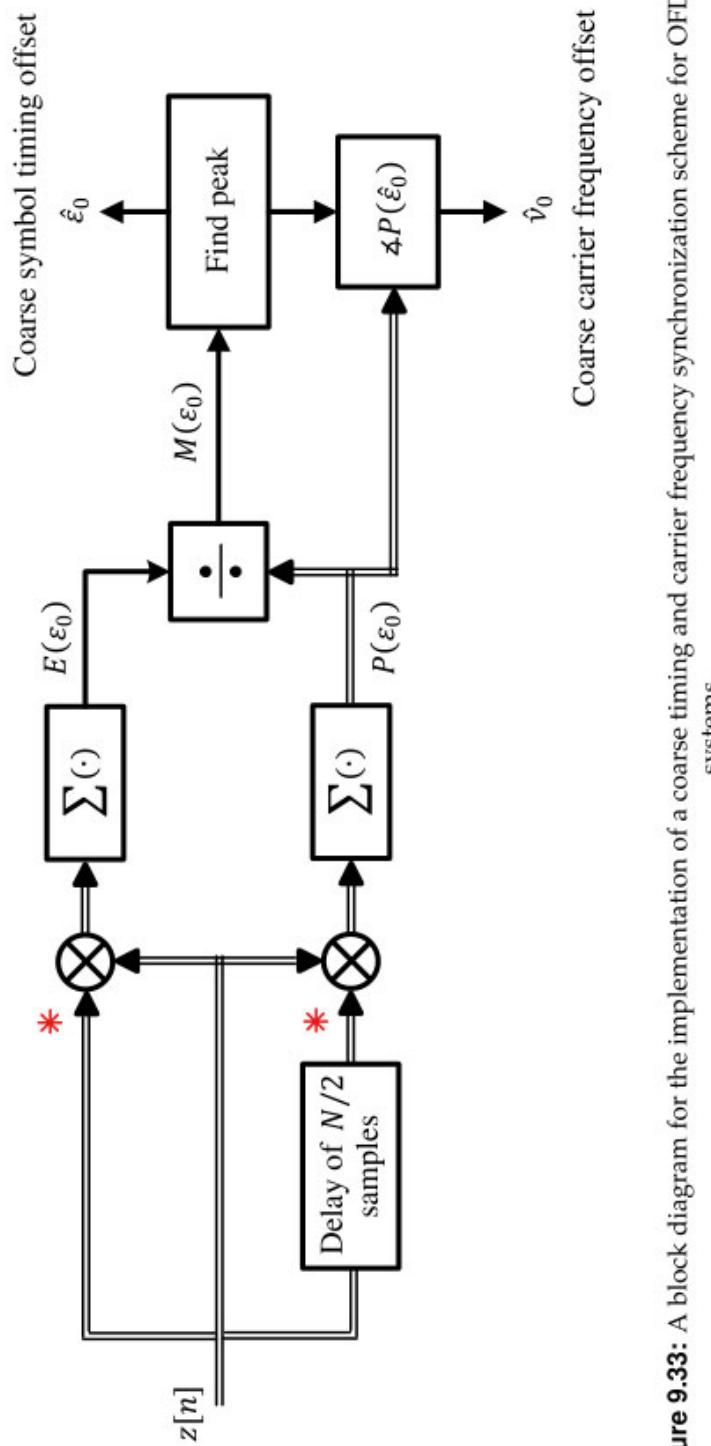


Figure 9.33: A block diagram for the implementation of a coarse timing and carrier frequency synchronization scheme for OFDM systems

be defined as

$$M(\varepsilon_0) = \frac{|P(\varepsilon_0)|^2}{[E(\varepsilon_0)]^2} \quad (9.21)$$

And the optimal timing instant $\hat{\varepsilon}_0$ (recall that timing synchronization in OFDM corresponds to locating the correct sample, i.e., an integer offset) is given as

$$\hat{\varepsilon}_0 = \max_{\varepsilon_0} M(\varepsilon_0) \quad (9.22)$$

Here, the denominator $E(\varepsilon_0)$ can be considered as a virtual AGC to provide a suitable signal gain for accurate timing metric detection. A block diagram of such an implementation is illustrated in Figure 9.33. We will later come back to this scheme to find out how it helps in coarse CFO correction.

Recursive Implementation

The main attraction of such a strategy is the correct result even in the presence of the CFO due to the magnitude squared operations involved. Furthermore, from a hardware point of view, such a timing metric can be implemented in a recursive manner thus simplifying its implementation. For this purpose, write $P(\varepsilon_0 + 1)$ from Eq (9.20) as

$$P(\varepsilon_0 + 1) = \sum_{i=0}^{N/2-1} z \left[\left(\overline{\varepsilon_0 + 1} + \frac{N}{2} \right) + i \right] z^* [\varepsilon_0 + i]$$

This is the same correlation window as computed for ε_0 but shifted by 1 sample to the right. Therefore, the leftmost sample (corresponding to $i = 0$) can be excluded from $P(\varepsilon_0)$ and an additional sample from the right (corresponding to $i = N/2$) can be added to $P(\varepsilon_0)$ to get $P(\varepsilon_0 + 1)$.

$$P(\varepsilon_0 + 1) = \underbrace{\sum_{i=0}^{N/2-1} z \left[\left(\varepsilon_0 + \frac{N}{2} \right) + i \right] z^* [\varepsilon_0 + i]}_{P(\varepsilon_0)} - z \left[\varepsilon_0 + \frac{N}{2} \right] z^* [\varepsilon_0] + z [\varepsilon_0 + N] z^* \left[\varepsilon_0 + \frac{N}{2} \right]$$

Similarly, the denominator $E(\varepsilon_0)$ can be recursively computed as

$$E(\varepsilon_0 + 1) = E(\varepsilon_0) - \left| z \left[\varepsilon_0 + \frac{N}{2} \right] \right|^2 + |z [\varepsilon_0 + N]|^2 \quad (9.23)$$

For these reasons, such a correlation window is implemented in most of the OFDM receivers for timing synchronization purpose.

Improvements on the Timing Scheme

We have seen above an implementation of a coarse timing estimation scheme through autocorrelation, i.e., correlation between the samples of the same sequence. When

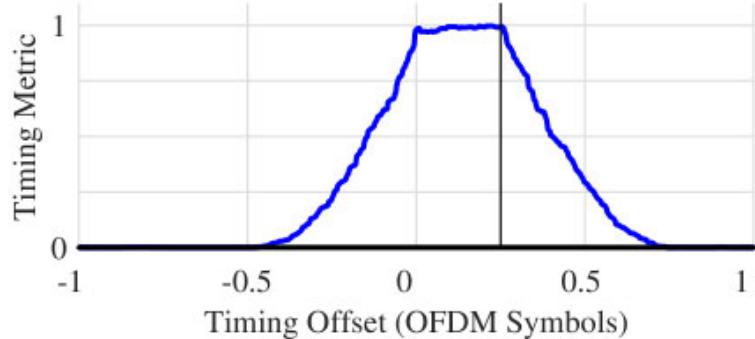


Figure 9.34: The timing metric $M(\varepsilon_0)$ of Eq (9.21) drawn for $N = 512$ subcarriers with a 25% CP, an AWGN channel and a random training sequence

such a timing synchronization scheme is implemented for an OFDM frame in which there is a training sequence prepended at the start, we get the timing metric $M(\varepsilon_0)$ of Eq (9.21) drawn in Figure 9.34 for $N = 512$ subcarriers, a 25% CP, a random training sequence and an AWGN channel. Notice that as the correlation window slides past the OFDM signal, the timing metric gradually rises near the training, reaches a plateau when the window hits the CP, remains constant for the duration of the CP (25%) and then slides down gradually after the training begins.

Since this is an AWGN channel, there is no ISI within this plateau to distort the signal and the first sample can be taken anywhere within this region. This was shown earlier in Figure 9.28 as a region of no concern. In a frequency selective channel, the length of this plateau is actually given as

$$\text{Plateau length} = \text{CP length} - \text{channel length}$$

In an AWGN channel, the plateau is as long as the CP.

The next question is how to utilize the timing metric $M(\varepsilon_0)$ for synchronization purpose. The most straightforward way is to raise a flag of preamble detection whenever $M(\varepsilon_0)$ rises past a certain threshold. However, the metric possesses a flat timing plateau instead of exhibiting a sharp peak. Moreover, due to this flat plateau, the variance of the timing synchronization estimate is quite high. For reliable operations and exact analysis, we like to have a repeatable experience with as less a variance as possible.

Many variations on this timing synchronization scheme have been proposed, some of which we discuss next.

Peak detection through 90% points: While a naive solution is to find the maximum point $\varepsilon_{0,\max}$ of $M(\varepsilon_0)$, this is not a reliable solution as well since it will move around in each realization due to the random noise. A better solution [44] is to detect the peak by finding this maximum point, then locate the points to the left and right of this maximum value that are 90% of this maximum value, and average the time points corresponding to these two 90% values.

Smoothing with a moving average filter: With the knowledge that the plateau has a maximum length equal to the CP in an AWGN channel, the flatness of this region can be turned into a linearly rising curve by convolving it with a moving average filter having a length equal that of the CP. The rationale is that a flat

signal resembles a rectangular signal and convolution of a rectangle with another rectangle in the moving average filter is a triangle that exhibits a relatively pronounced peak.

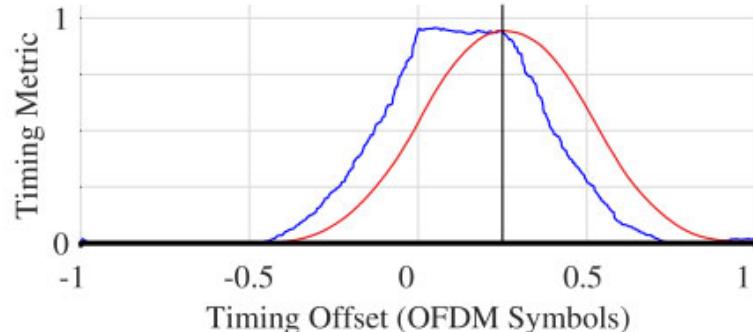


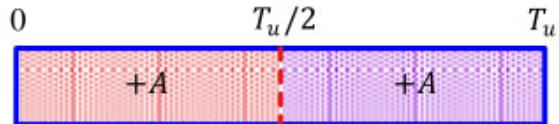
Figure 9.35: The timing metric $M(\epsilon_0)$ of Figure 9.34 convolved with a moving average filter of length equal to the CP length

Such a modified metric is plotted in Figure 9.35 where the single point of the sequence start can be isolated. Note that the convolution result is not a triangle because the plateau is not a rectangular signal in isolation. In any case, with the peak of the timing metric $M(\epsilon_0)$ in hand, many peak finding strategies – similar to the symbol centric and zero crossing techniques in Chapter 7 – can also be applied if a long frame of OFDM symbols is available and convergence time is not much of a concern. This happens for example in continuous transmission systems.

Repetitions with the same or different signs: Ref. [45] proposed an interesting alternative technique to improve the timing metric $M(\epsilon_0)$ in Eq (9.21). Consider the training sequence with 2 identical halves as a special case of a training sequence with L repetitions as well as different signs. For example, denote the length $N/2$ training sequence by A . Then, until now, we have worked on a sequence A repeated twice and with $++$ signs, as illustrated in Figure 9.36a. By $++$ sign, we do not mean the signs of the individual training symbols before the iFFT but that the whole length $N/2$ sequence A after the iFFT is multiplied with a $+1$ to generate $\{+A, +A\}$.

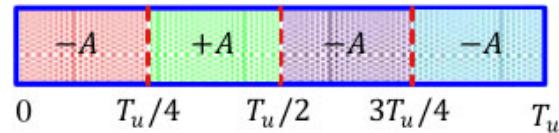
Now consider Figure 9.36b where a training sequence with $L = 4$ identical parts is drawn. While the underlying training before the length $N/4$ iFFT is the same, each iFFT output quarter is multiplied with a different pattern, $\{-A, +A, -A, -A\}$ in this case, and concatenated together. The advantage of such an approach is that even after the insertion of the CP, there is no plateau left due to the presence of *only one unique time shift* for which the whole correlation window exactly matches. The timing metric $M(\epsilon_0)$ is still given by the ratio of $P(\epsilon_0)$ and $E(\epsilon_0)$ but both of these metrics are modified to implement the summation of products between the samples of the repetitive parts. This is shown in Figure 9.37 where the logic of sign inversion responsible for the steep rolloff of the timing metric is also evident. Later, we will see how such a strategy is useful for CFO estimation as well.

Training sequence with 2 identical parts
with signs + +



(a) 2 identical parts

Training sequence with 4 identical parts
with signs - + - -



(b) 4 identical parts

Figure 9.36: A training sequence with L identical parts multiplied with different signs is a generalization of 2 identical parts with ++ signs

Training sequence with 4 parts

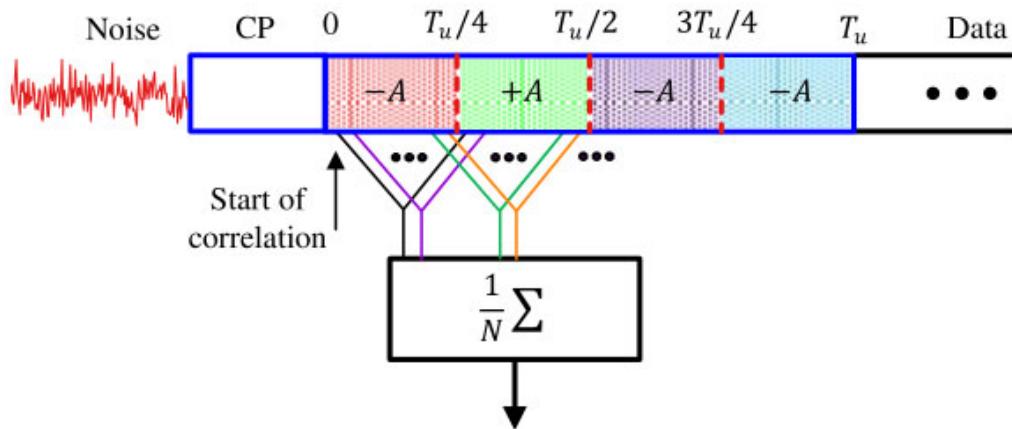


Figure 9.37: The logic behind the correlation window involving L identical parts

It must be kept in mind that during the computation of $P(\varepsilon_0)$, the actual repetitive pattern (e.g., $\{-A, +A, -A, -A\}$) must be multiplied with the corresponding training part for canceling its sign. Figure 9.38 compares the timing metric $M(\varepsilon_0)$ for $L = 4$ repetitions with the pattern $\{-A, +A, -A, -A\}$ and original $L = 2$ repetitions employing the pattern $\{+A, +A\}$. It is evident that the plateau

has disappeared and a steep rolloff of the timing metric helps in locating the exact timing sample for synchronization purpose. The concept demonstrated here for 4 repetitions can be easily generalized to L repetitive parts.

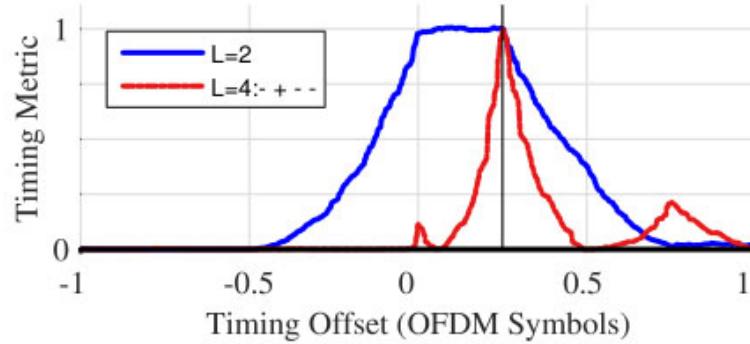


Figure 9.38: The timing metric $M(\varepsilon_0)$ for $L = 4$ repetitions with the pattern $\{-A, +A, -A, -A\}$ and original $L = 2$ repetitions with the pattern $\{+A, +A\}$

Let us now return towards the preamble of IEEE 802.11a standard discussed in Example 9.1.

Example 9.2

In 802.11a WiFi standard, the preamble consists of a *Short Training Field (STF)* with 10 repetitions and a *Long Training Field (LTF)* with 2 repetitions, as drawn in Figure 9.39 where S denotes the STF. Other than coarse timing synchronization, the STF is also utilized for several different purposes (e.g., signal detect, AGC, diversity selection and coarse frequency synchronization). The LTF consists of two identical halves used for fine frequency synchronization as well as channel estimation while a GI2 represents a guard interval of twice the normal length.

The STF in 802.11a is generated by taking an iFFT of the sequence below. Notice that only a few subcarriers from $k = -26$ to $k = +26$ are employed for this purpose which are multiples of 4, the effect of which we shortly see. From $k = -26$ to $+26$,

$$a_{\text{STF}}[k] = \sqrt{\frac{13}{6}} \left\{ 0, 0, \underbrace{+V}_{-24}, \underbrace{0, 0, 0, -V}_{-20}, \underbrace{0, 0, 0, +V}_{-16}, \underbrace{0, 0, 0, -V}_{-12}, \right. \\ \left. \underbrace{-V}_{-8}, \underbrace{0, 0, 0, +V}_{-4}, \underbrace{0, 0, 0, 0}_{0}, \underbrace{0, 0, 0, -V}_{+4}, \underbrace{0, 0, 0, -V}_{+8}, 0, \right. \\ \left. 0, 0, \underbrace{+V}_{+12}, \underbrace{0, 0, 0, +V}_{+16}, \underbrace{0, 0, 0, +V}_{+20}, \underbrace{0, 0, 0, +V}_{+24}, 0, 0 \right\}$$

where V is a complex number given by

$$\begin{array}{ll} I \rightarrow & V_I = 1 \\ Q \uparrow & V_Q = 1 \end{array}$$

The scaling factor $\sqrt{13/6}$ comes from normalizing the average power of the OFDM symbol which employs 12 out of 52 available subcarriers. Combined with $\sqrt{1/2}$ that is

used to make V above a unit energy number,

$$\sqrt{\frac{1}{2} \cdot \frac{52}{12}} = \sqrt{\frac{13}{6}}$$

What is the implication of only the subcarriers in STF being multiples of 4 utilized for modulation purpose? This can be seen as a case of upsampling a signal consisting of 13 values by 4. From Section 2.7.2, we know that upsampling by inserting L zeros in one domain causes a repetition by L in the other domain. In frequency domain, this repetition was visible in the form of the L spectra becoming a part of the upsampled signal spectrum. Here, the zero insertion is in frequency domain and hence $L = 4$ repetitions appear in time domain. The time duration of the iFFT output is the FFT size N multiplied with the sample time $T_S = 1/20 \times 10^6 = 50$ ns. Thus, the time period of one repetition consisting of $64/4 = 16$ samples is given by

$$\frac{NT_S}{4} = \frac{64 \times 50 \times 10^{-9}}{4} = 0.8\mu\text{s}$$

and the total duration of the STF field is $10 \times 0.8 = 8\mu\text{s}$. As another example, the LTF is generated through the iFFT of the following symbols.

$$a_{\text{LTF}}[k] = \left\{ 1, 1, -1, -1, 1, 1, -1, 1, -1, 1, 1, 1, 1, 1, -1, -1, 1, 1, -1, 1, 1, 1, 1, 1, 1, 0, 1, -1, -1, 1, 1, -1, 1, -1, 1, -1, 1, -1, -1, -1, -1, 1, 1, -1, -1, 1, -1, -1, 1, -1, 1, 1, 1, 1, 1, 1 \right\}$$

Observe that all 52 subcarriers are utilized. There are numerous other metrics and even more variations on those metrics that have been discovered in the enormous research literature on OFDM systems. Nevertheless, the fundamental concepts remain similar to what has been discussed above. An interested reader should study the intriguing training sequences defined in various standards implementing the OFDM systems such as WiFi (802.11a/g/n/ac), WiMax (802.16), LTE (4G), NR (4.5G) and 5G systems.

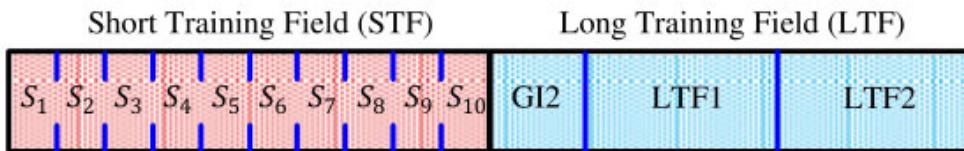


Figure 9.39: Preamble for IEEE 802.11a PHY

Once a coarse timing estimate $\hat{\epsilon}_0$ is located, it is also customary to advance it by a predetermined number of samples as a safety margin against channel dispersion. Recall from the discussion on the effect of STO that taking the FFT between the ideal sampling instant and the point where the channel ends has no effect on the demodulator performance because any residual phase becomes part of the channel phase $\angle H[k]$ for each k . This safety margin introduces a cushion in synchronization design where an advance of a few samples can be afforded but not a delay of even 1 sample.

As a final remark, we have established the necessity of some kind of periodicity in the Rx signal to mark the timing synchronization instant. For OFDM systems consisting of long packets or continuous transmissions such as DAB or DVB, the timing

synchronization can be accomplished through using the cyclic prefix which is essentially a short repetition in an OFDM symbol. Nevertheless, the underlying concepts related to building useful metrics and defining thresholds remain unchanged. For example, the timing metric using a CP can be written as

$$P_{CP}(\varepsilon_0) = \sum_{i=-N_{CP}}^{-1} z[(\varepsilon_0 + N) + i] z^* [\varepsilon_0 + i] \quad (9.24)$$

where the repetition happens at a time difference of N samples and gives rise to a high correlation. The other two metrics, namely $R(\varepsilon_0)$ and $M(\varepsilon_0)$, can also be defined in a similar manner. Keep in mind that for one length N OFDM symbol, the sample indices including the CP are defined as follows.

$$-N_{CP}, \dots, -1, 0, 1, 2, \dots, N-1$$

Fine Timing Synchronization

The coarse timing offset above was computed through using the autocorrelation, i.e., correlation between the samples of the same sequence. Advantages of this approach are its computational complexity due to recursive structure in Eq (9.23) and that it is immune to any impairment caused by the CFO. Nonetheless, the disadvantage of such an approach is that both sets of samples are noisy and hence the performance remains suboptimal.

One remedy to this problem is that we already have a ‘clean’ copy of the training sequence stored at the Rx. Why not correlate the noisy Rx signal with the stored clean sequence? Recall that the noisy version of the training is affected by the CFO and hence even its correlation with the clean sequence will be rotating instead of generating a single sharp peak.

To solve this issue, a coarse frequency offset is also computed after the coarse timing estimation and compensated for which we discuss in the next section. Once the noisy Rx sequence is mostly free of the coarse CFO, we denote it by $\tilde{z}[n]$ and a fine timing estimate is computed by cross-correlating it with the clean training sequence $s[n]$ stored at the Rx. The new metric $\tilde{M}(\varepsilon_0)$ is given by

$$\tilde{M}(\varepsilon_0) = \sum_{n=0}^{N-1} \tilde{z}[n + \varepsilon_0] s^*[n]$$

Such a strategy is also known as matched filtering because the correlation is implemented with the known training sequence $s[n]$. In practice, such a cross-correlation produces highly accurate results. To reduce the acquisition time, this operation can be carried out partially with a portion of $s[n]$ as well. For IEEE 802.11a example, after the coarse timing computation from the Short Training Field (STF), a partial correlation with 32 or 64 samples of the Long Training Field (LTF) can be carried out instead of the complete length 128 sequence for a fine timing estimate.

On the other hand, in the continuous or frame-based OFDM systems, fine timing can be accomplished through averaging over a number of OFDM symbols. This is because the acquisition time requirements are relevantly much more relax as compared to packet based systems and a simple tradeoff between extending the averaging interval for an increased accuracy is feasible.

Next, we turn our attention towards the problem of CFO estimation in an OFDM system.

9.5 Carrier Frequency Synchronization

In OFDM systems, the Carrier Frequency Offset (CFO) is also corrected in multiple stages, a coarse and a fine stage. Before the synchronization procedures, we need to understand how a CFO impacts the demodulated symbols.

Effect of Carrier Frequency Mismatch

Just like a single-carrier signal, the baseband OFDM signal at the Tx is upconverted by a local oscillator to a carrier frequency F_C and downconverted at the Rx by its local oscillator at a frequency slightly different than F_C . Along with any Doppler shift encountered in the channel path, this imparts a Carrier Frequency Offset (CFO) equal to F_Δ in the Rx signal[†].

As we saw before in the discussion on the timing mismatch, we can divide the CFO F_Δ into an integer and a fractional part.

Integer CFO: What happens in a single-carrier system when a symbol timing offset is an integer multiple of symbol time, say $2T_M$, away from the ideal instant? It would not induce any ISI but our symbol numbers will become wrong. For an STO of $2T_M$, our symbol 0 would appear at symbol instant 2 and a similar shift will appear for the rest of the symbols.

Exactly in a similar way, and by virtue of time frequency duality, an integer CFO will not induce any ICI in the Rx signal and the orthogonality among the subcarriers is maintained. However, the symbols after the DFT will appear at wrong locations and an absence of ICI does not indicate a successful transmission. These wrong symbol indices either cause a signal decoder failure or failed header checks in a higher layer.

Fractional CFO: Similar to a fractional STO in a single-carrier system, a fractional CFO would ‘sample’ the Rx waveform in frequency domain at a shifted location thus inducing ICI. This is plotted in Figure 9.40 for a 25% CFO where the subcarrier demodulation is not being performed at optimal instants. Compare this frequency domain ICI with the time domain ISI in Figure 7.2.

On the same note, a scatter plot isolating the CFO distortion in OFDM resembles the scatter plot isolating the STO distortion in single-carrier systems and drawn in Figure 7.4 where the demodulated symbols form a cloud around the ideal constellation points. A little difference can arise from the presence of a CP that causes a phase rotation of those clouds away from those constellation points.

Coarse Frequency Synchronization

First, recall Section 6.4.1 on a delay and multiply technique to estimate the CFO in a feedforward manner. The mathematical details of this section were elaborated in the context of a single-carrier system. A similar method can be applied to understand coarse frequency synchronization in an OFDM system. The main idea is that the frequency of any complex sinusoid can be estimated by taking a phase difference

[†]Here, we ignore another major source of carrier distortion, namely the phase noise.

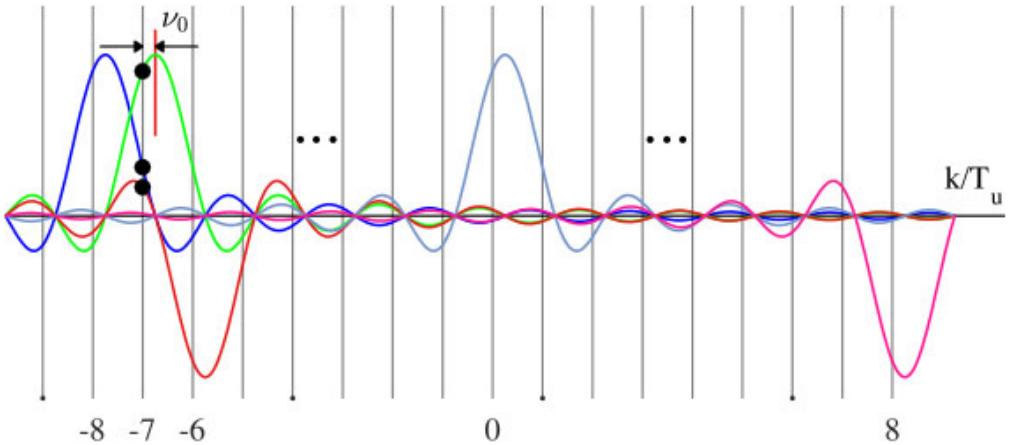


Figure 9.40: A fractional Carrier Frequency Offset (CFO), shown here as 25% of subcarrier spacing $\Delta F = 1/T_u$ causing ICI in the neighbouring symbols

between any of its two samples at known time instants, say T_2 and T_1 . This is because by definition, the phase difference is given by

$$\Delta\phi = 2\pi F(T_2 - T_1)$$

Since T_2 and T_1 are known and $\Delta\phi$ is estimated through some method, the underlying frequency is

$$F = \frac{\Delta\phi}{2\pi(T_2 - T_1)} \quad (9.25)$$

The good thing about OFDM is that *we have N such complex sinusoids (embedded in noise) already available.*

For this purpose, consider Figure 9.41 which illustrates a single subcarrier (out of N) with index $k = 2$ that is rotating at a frequency of $2/N$ cycles/sample (i.e., it completes 2 cycles in N samples, or T_u seconds). The first cycle is shown in a solid blue line in the first half of the OFDM symbol while the second cycle in a dashed red line in the second half of the OFDM symbol.

Also shown in a dotted line is another complex sinusoid with a slightly different frequency that is the same subcarrier shifted by a normalized CFO v_0 . The actual subcarrier frequency k/N is known but the Rx signal contains N such subcarriers shifted in frequency by v_0 and our task is to estimate this unknown CFO v_0 . If we implement a conjugate product of the actual subcarrier at k/N and this Rx subcarrier at frequency $(k + v_0)/N$, then we will be left with a complex sinusoid rotating at v_0/N .

We already have the relevant samples $z[n]$ in the form of the Rx OFDM signal, so the only question before we can apply Eq (9.25) is that which two time instants T_2 and T_1 should be chosen? We will later see that this choice establishes the acquisition range of the CFO estimation technique. If we recall the definition of the metric $P(\varepsilon_0)$ from Eq (9.20), we find that the phase difference between two samples has already been computed during the timing synchronization procedure if

$$T_2 = T_u/2, \quad T_1 = 0$$

Not only that, but computation of $P(\varepsilon_0)$ (reproduced below) involves a summation of $N/2$ such phase differences, the next between $T_2 = T_u/2 + 1$ and $T_1 = 1$, and so on

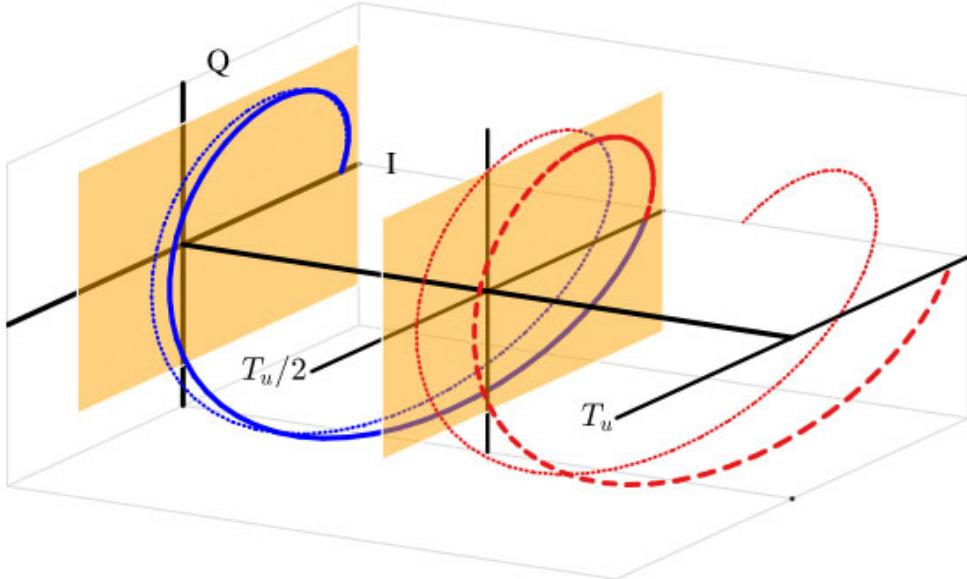


Figure 9.41: Phase difference of a complex sinusoid at two time instants gives its frequency

until it reaches $T_2 = T_u$ and $T_1 = T_u/2$.

$$P(\varepsilon_0) = \sum_{i=0}^{N/2-1} z \left[\left(\varepsilon_0 + \frac{N}{2} \right) + i \right] z^* [\varepsilon_0 + i]$$

In all these terms, the time difference is equal to $T_u/2$ while the summation is simply a moving average filter. As shown previously in Figure 9.32, the conjugate operation rotates the length $T_u/2$ subcarriers in the first half in an opposite direction to cancel them out as well as the modulation signs of the pseudo-random sequence. As a result of this procedure, it produces the phase differences at each sample of that window equal to

$$\Delta\phi = 2\pi F_\Delta \frac{T_u}{2} = \angle P(\varepsilon_0)$$

Using the definition of normalized frequency offset $\nu_0 = F_\Delta/\Delta_F = F_\Delta T_u$, the above equation yields

$$\begin{aligned} \hat{\nu}_0 &= \frac{1}{\pi} \angle P(\hat{\varepsilon}_0) \\ &= \frac{1}{\pi} \angle \left\{ \sum_{i=0}^{N/2-1} z \left[\left(\hat{\varepsilon}_0 + \frac{N}{2} \right) + i \right] z^* [\hat{\varepsilon}_0 + i] \right\} \end{aligned} \quad (9.26)$$

where $\hat{\varepsilon}_0$ is the coarse timing offset that marks the beginning of the training sequence. A block diagram of such a scheme to estimate the coarse frequency offset $\hat{\nu}_0$ along with the timing offset $\hat{\varepsilon}_0$ was drawn in Figure 9.33. This is the most widely used coarse frequency synchronization method in OFDM systems due to its simplicity and robustness.

Note that if the CP was used for the coarse timing synchronization procedure, then from Eq (9.24), we get

$$\vartheta_0 = \frac{1}{2\pi} \angle P_{CP}(\hat{\varepsilon}_0) = \frac{1}{2\pi} \angle \left\{ \sum_{i=-N_{CP}}^{-1} z[(\hat{\varepsilon}_0 + N) + i] z^* [\hat{\varepsilon}_0 + i] \right\}$$

The factor 2π instead of π in the denominator arises due to N sample spacing between the two correlation participants, instead of $N/2$ in the case of the training sequence with two identical halves. Furthermore, in OFDM systems with continuous transmissions, the imaginary part of the above expression can be used as a frequency error detector in a frequency locked loop.

Acquisition Range

One drawback of the above coarse frequency synchronization technique is its narrow acquisition range. First, recall the definition of phase $\Delta\phi = 2\pi F(T_2 - T_1)$ and the uniqueness of phase within $\{-\pi, +\pi\}$. Then, for $T_2 - T_1 = T_u/2$, we get the maximum frequency $|v_{0,\max}|$ that can be uniquely determined.

$$2\pi |F_{\Delta,\max}| \frac{T_u}{2} < \pi, \quad \text{or} \quad |F_{\Delta,\max}| < \frac{1}{T_u} = \Delta_F \\ |v_{0,\max}| < 1 \quad (9.27)$$

where $v_{0,\max} = F_{\Delta,\max} T_u$. There will be an ambiguity when the CFO F_Δ approaches **one subcarrier spacing** Δ_F either from the positive or the negative side.

Another route to understand this concept is through the sampling theorem. For samples taken at a rate of $F_S = 1/T_S$, the maximum frequency to avoid aliasing is $F_S/2 = 1/2T_S$. Here, the expression

$$z \left[\left(\hat{\varepsilon}_0 + \frac{N}{2} \right) + i \right] z^* [\hat{\varepsilon}_0 + i]$$

in $P(\varepsilon_0)$ is essentially 'sampling' the waveform at two time instants $N/2$ samples (or $T_u/2$ seconds) apart, see Figure 9.41 (the summation in $P(\varepsilon_0)$ is just to average several such samples to reduce the additive noise). Thus, according to the sampling theorem, the maximum frequency before aliasing is $1/2(T_u/2) = 1/T_u = \Delta_F$. Consequently, this is the largest unique CFO F_Δ that can be determined through this technique.

What is the most straightforward method to increase the maximum allowable CFO from the perspective of the sampling theorem? This is simply to increase the sampling rate, i.e., taking the two samples closer in time to each other. In this context, we can increase the desired samples by taking the correlation at closely spaced intervals than $T_u/2$. This can be done by using a training sequence that is identical not only for two halves but for L identical parts. We saw one such scenario in the preamble of IEEE 802.11a standard in Example 9.2 where the short training field consists of 10 identical repetitions, each consisting of 16 samples. Although the FFT size in IEEE 802.11a is $N = 64$, the total length of the short (as well as long) training field is 160 samples, i.e., there are $L = 64/16 = 4$ repetitions within $N = 64$ samples.

For a general training sequence with L identical parts, the time difference between the two correlation participants is reduced from $N/2$ to N/L samples. Then, the

correlation expression above can be changed to

$$z \left[\left(\hat{\varepsilon}_0 + \frac{N}{L} \right) + i \right] z^* [\hat{\varepsilon}_0 + i]$$

There are N/L such terms in the correlation window, so we can add them together to average out the noise. Moreover, when the full correlation window of N samples is taken into account, we get

$$\begin{aligned} \hat{\nu}_0 = \frac{L}{2\pi} \angle & \left\{ \sum_{i=0}^{N/L-1} z \left[\left(\hat{\varepsilon}_0 + \frac{N}{L} \right) + i \right] z^* [\hat{\varepsilon}_0 + i] + \right. \\ & \sum_{i=N/L}^{2N/L-1} z \left[\left(\hat{\varepsilon}_0 + \frac{N}{L} \right) + i \right] z^* [\hat{\varepsilon}_0 + i] + \dots \\ & \left. \sum_{i=(L-2)N/L}^{(L-1)N/L-1} z \left[\left(\hat{\varepsilon}_0 + \frac{N}{L} \right) + i \right] z^* [\hat{\varepsilon}_0 + i] \right\} \end{aligned} \quad (9.28)$$

where the factor $L/2\pi$ is a generalization of $2/2\pi = \pi$ in $L = 2$ identical parts from Eq (9.26). From the previous analysis that lead to Eq (9.27), the new acquisition range is easily seen to be

$$\begin{aligned} 2\pi |F_{\Delta,\max}| \frac{T_u}{L} < \pi, \quad \text{or} \quad |F_{\Delta,\max}| < \frac{1}{2T_u/L} = \frac{L}{2} \Delta_F \\ |\nu_{0,\max}| < \frac{L}{2} \end{aligned} \quad (9.29)$$

With this modification, the CFO acquisition range increases as L increases. On the other hand, the number of samples available for computing one correlation term gets reduced in proportion to L . Next, we turn our attention towards improving the frequency estimate through applying finer techniques.

Fine Frequency Synchronization

The preamble in many OFDM systems consists of not one but two training sequences. The first training sequence is used for coarse synchronization strategies while the second preamble is used for fine frequency synchronization and channel estimation.

We discussed in Example 9.2 that the Short Training Field (STF) in IEEE 802.11a standard consists of 10 repetitions of the same length 16 sequence. After the signal detection, AGC convergence and coarse timing synchronization, the last repetitions of the STF can be employed for a coarse frequency estimate. Next, the Long Training Field (LTf) consists of a training sequence with 2 identical halves, each 64 samples in length.

- Since the residual CFO is now small, acquisition range and ICI distortion are not significant issues anymore. Therefore, it can be compensated for by employing the same correlation strategy over these long periods.
- Since this is not a timing synchronization problem anymore, the metric in the denominator to normalize the samples energy is not required.

Assuming that by virtue of successful timing synchronization, $z_{\text{LTF}}[n]$ refers to the portion of the Rx signal corresponding to the LTF. Then, we estimate $\hat{\nu}_0$ as

$$\hat{\nu}_0 = \frac{1}{2\pi} \angle \left\{ \sum_{n=0}^{64-1} z_{\text{LTF}}[64+n] z_{\text{LTF}}^*[n] \right\}$$

There is no other factor with 2π because the normalized CFO is defined with respect to T_u which corresponds to $N = 64$ samples in IEEE 802.11a. This is the same as the length of the LTF and hence $T_2 - T_1$ in Eq (9.25) is T_u as well.

Since the conjugate operation cancels the data modulation, the resulting signal turns out to be a complex sinusoid of frequency F_Δ embedded in noise. Notice that there is only one correlation being computed at a certain lag. We can also follow a strategy similar to the estimators in Section 6.3.4 that compute multiple correlations at all possible lags and subsequently generate a much accurate CFO estimate. Algorithms solving such a problem can then certainly be applied in the context of OFDM as well. Nevertheless, it defeats the purpose of fast synchronization manifested by a block processing configuration in general and OFDM systems in particular.

Finally, as far as the *integer CFO correction* is concerned, it is usually corrected by taking the DFT after completing the timing and frequency synchronization procedures. Then, the demodulated subcarriers in a training sequence or embedded pilots within an OFDM symbol are compared with the expected locations of those subcarriers in the Rx signal. If the integer CFO is zero, the training or pilots in the Rx signal are present at the correct locations. Otherwise, the shift in the pattern can be found by examining the Rx signal. Let us find out how.

Until now, we ignored the OFDM symbol index m in our expressions to avoid index cluttering. Now this index needs to be introduced to compute the integer CFO $[\nu_0]_{\text{int}}$. As described before, when an integer CFO $[\nu_0]_{\text{int}}$ is present, each subcarrier k appears at a location $k + [\nu_0]_{\text{int}}$, including the pilot subcarriers. This spectral shift can be found by exploiting the following facts about the Tx pilots.

- They remain the same for each OFDM symbol $m - 1, m$, and so on, i.e.

$$a_{m-1}[k] = a_m[k] \quad \text{if } k \text{ is a pilot subcarrier}$$

- They are boosted in power by a factor of B^2 .

Let us denote this set of pilot indices by k_p . Recall from Example 9.1 regarding an IEEE 802.11a system that the set $k_p = \{-21, -7, +7, +21\}$ is used for inserting the pilot subcarriers drawn in Figure 9.22. This implies that correlating the samples of the FFT output for two consecutive OFDM symbols $m - 1$ and m and each subcarrier k_p yields

$$\text{corr}_0[m] = \sum_{k_p} Z_m[k_p] \cdot Z_{m-1}^*[k_p] \quad (9.30)$$

Assuming a slowly varying channel, the channel coefficients remain almost the same for these two OFDM symbols.

$$H_{m-1}[k] \approx H_m[k]$$

Using $Z[k] = a[k]H[k]$ and ignoring other phase distortions due to the oncoming absolute value operation of the correlation metric, the correlation output thus becomes

$$\begin{aligned} |\text{corr}_0[m]| &= \left| \sum_{k_p} a_m[k_p] H_m[k_p] \cdot a_{m-1}^*[k_p] H_{m-1}^*[k_p] \right| \\ &= \begin{cases} B^2 \sum_{k_p} \left\{ |a_m[k_p]|^2 \cdot |H_m[k_p]|^2 \right\} & [\hat{\nu}_0]_{\text{int}} = 0 \\ \left| \sum_{k_p} \left\{ a_m[k_p] a_{m-1}[k_p] \cdot |H_m[k_p]|^2 \right\} \right| & [\hat{\nu}_0]_{\text{int}} \neq 0 \end{cases} \end{aligned}$$

Notice that if $[\hat{\nu}_0]_{\text{int}} = 0$, k_p corresponds to the true pilots and the output magnitude is proportional to $B^2 |a_m[k_p]|^2$ which is a large number. On the other hand, if $[\hat{\nu}_0]_{\text{int}} \neq 0$, the correlation output is proportional to $\sum_{k_p} a_m[k_p] a_{m-1}[k_p]$ which is a small number owing to the data randomness and different data modulation symbols for each m [†]. This is drawn in Figure 9.42 for $[\hat{\nu}_0]_{\text{int}} = -2$ where you can observe the correlation sum being performed at shift $I = 0$. Due to the random data, this summation turns out to be a small value.

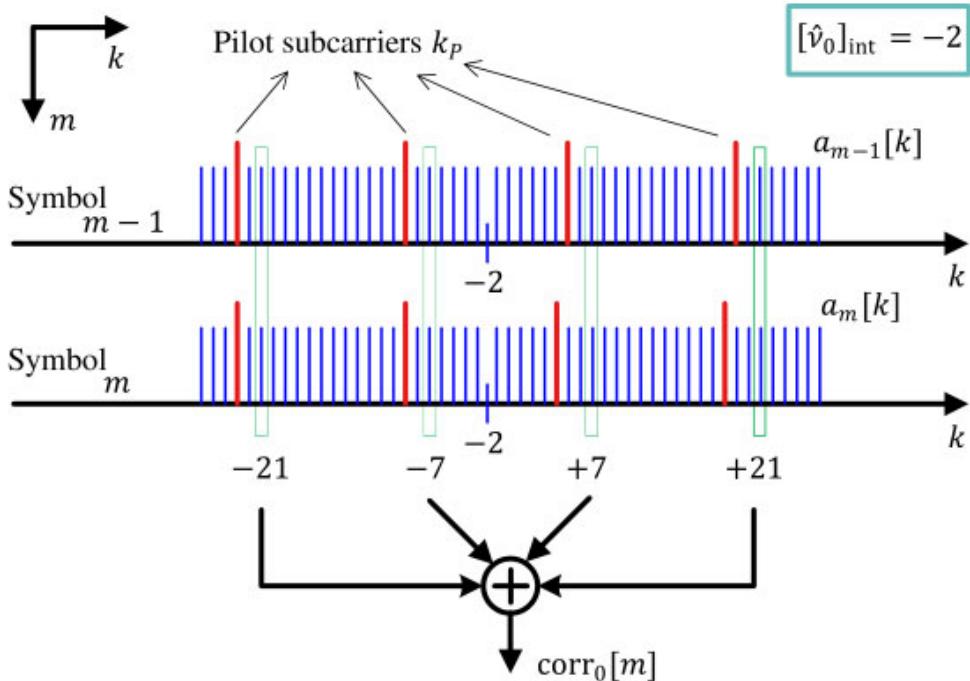


Figure 9.42: Estimation of an integer CFO $[\hat{\nu}_0]_{\text{int}} = -2$ through pilot subcarriers

Next, a similar procedure is carried out for the set of indices $k_p - 1$ to compute $|\text{corr}_{-1}[m]|$, indices $k_p + 1$ to compute $|\text{corr}_{+1}[m]|$, and so on, up to a prescribed limit

[†]Recall Eq (7.77) on data randomness, reproduced below.

$$\text{Mean}\{a[m] \cdot a[i]\} = \begin{cases} A^2 & i = m \\ 0 & \text{otherwise} \end{cases}$$

to which the integer CFO can vary. The magnitude of the largest such correlation $|\text{corr}_I[m]|$ gives the estimate $[\hat{\nu}_0]_{\text{int}}$.

$$[\hat{\nu}_0]_{\text{int}} \Rightarrow \max_I |\text{corr}_I[m]| \quad (9.31)$$

This procedure is usually known as post-FFT synchronization due to its implementation after the FFT block. In a similar fashion, the previous approaches are known as pre-FFT synchronization. Both time domain (pre-FFT) and frequency domain (post-FFT) synchronization strategies are crucial for correct data symbols demodulation. Target specifications and existing constraints guide the system designers to choose a combination of appropriate approaches.

Note 9.4 Timing and frequency synchronization

In a study of single-carrier systems, you will find a large number of solutions for timing synchronization, far exceeding those on frequency synchronization. The situation in multicarrier systems is its dual and we find many more algorithms on carrier frequency rather than timing synchronization. This seems a bit puzzling because a single-carrier system will simply not work if even a small uncompensated frequency offset is present in the signal that steadily spins the Rx constellation. On the other hand, a small timing offset degrades the SNR by spreading the Rx constellation but the system will still keep delivering many correct bits.

This is exactly why timing synchronization in single-carrier systems is more important. Once we solve the problems that could completely stop the show, we want to iron out the last bit of inefficiency that remains. So even if the timing is *almost* right, 1 error in 100 bits is too bad and 1 error in 1 million bits is barely good enough. Probably this is the same reason why many of the problems for those living in an affluent part of a city are almost non-existent for those living in an impoverished one.

Now we explain a sampling clock synchronization strategy that simultaneously corrects for any residual CFO as well and hence can be categorized as fine tracking.

9.6 Sampling Clock Synchronization

The nature of a Sampling Clock Offset (SCO), i.e., a difference in sampling clock frequency and also known as a symbol timing frequency offset, has been explained in Section 7.10 before. It was also visually demonstrated through Figure 7.58 the meaning of a sampling clock offset between the Tx and Rx waveforms. Here, our purpose is to devise a strategy to correct this error in OFDM systems.

We start with the effect of an SCO on the Rx signal.

Effect of Timing/Clock Frequency Mismatch

We derived Eq (9.18) to understand how the normalized CFO and the normalized STO affect the FFT input at the Rx. Now assuming a successful timing synchronization $\varepsilon_0 = 0$, we study the effect of the residual normalized CFO and a normalized Sampling Clock Offset (nSCO). One question that arises here is that why a perfect CFO cannot be assumed in the system model like the STO. Recall from the timing synchronization

problem in single-carrier systems that timing needs to be continuously adjusted and can almost never be set and done. Similar is the case with the frequency offset in an OFDM systems and a residual CFO still needs to be included.

For a Rx sampling the waveform at sampling intervals of \tilde{T}_S instead of T_S , an SCO and its normalized versions are defined as

$$\xi = \tilde{T}_S - T_S, \quad \xi_0 = \frac{\tilde{T}_S - T_S}{T_S} = \frac{\xi}{T_S}$$

Next, taking the same route used to derive Eq (9.17) for $\varepsilon_\Delta = \varepsilon_0 = 0$ but sampling at $t = n\tilde{T}_S$, we get

$$\begin{aligned} 2\pi(F_k + F_\Delta)t|_{t=n\tilde{T}_S} &= 2\pi(F_k + F_\Delta)n\tilde{T}_S \\ &= 2\pi(F_k + F_\Delta)(n\tilde{T}_S + nT_S - nT_S) \\ &= 2\pi(F_k + F_\Delta)\left(nT_S + n\frac{\tilde{T}_S - T_S}{T_S}T_S\right) \\ &= 2\pi(F_k + F_\Delta)(n + n\xi_0)T_S \\ &= 2\pi\frac{k\Delta_F + F_\Delta}{F_S}(1 + \xi_0)n \\ &= \frac{2\pi}{N}(k + v_0)(1 + \xi_0)n \end{aligned}$$

where we have used $F_k = k\Delta_F$, $F_S = N\Delta_F$ and $F_\Delta/\Delta_F = v_0$. In the presence of the CFO and STO, the argument of the subcarriers in the FFT input of Eq (9.18) was derived in Eq (9.17). In the current settings of an SCO and zero STO, this argument becomes

$$\begin{aligned} 2\pi(F_k + F_\Delta)t|_{t=n\tilde{T}_S} &= \frac{2\pi}{N}(k + v_0)(1 + \xi_0)n \\ &= \frac{2\pi}{N} \left\{ \underbrace{kn}_{\text{Term 1}} + \underbrace{v_0(1 + \xi_0)n}_{\text{Term 2}} + \underbrace{(k\xi_0)n}_{\text{Term 3}} \right\} \end{aligned} \quad (9.32)$$

Notice that the index n is common to all the terms above, as opposed to the situation we encountered in timing offset synchronization before. As we saw in that case, these terms are handled as follows.

- From the DFT definition in Eq (1.53), the FFT at the Rx consists of subcarriers with the arguments $-kn$ which simply cancels out Term 1 above.
- Being a function of time n , term 2 appears as a CFO equal to $v_0(1 + \xi_0)$ at the FFT output. The part $v_0 \cdot \xi_0$ is in addition to the original CFO v_0 and is very similar to the Doppler shift (actually superimposed over the CFO by a moving or drifting Rx clock).
- This leaves Term 3 as a function of both k and n . The data symbols at each subcarrier k get rotated by the product of this index k varying from $-N/2$ to $N/2 - 1$ with the SCO ξ_0 and this is what causes the constellation rotation. However, in the timing offset case, this rotation was equal to $k\varepsilon_0$ and hence the same for each OFDM symbol. Here, it is different from one OFDM symbol to

the next due to the presence of the factor n and depends on the OFDM symbol number in the frame. For example, symbol 3 will experience a different amount of SCO induced rotation as compared to symbol 100. Considering the relation $N_{\text{OFDM}} = N + N_{\text{CP}}$, sample number n is equal to $mN_{\text{OFDM}} + N_{\text{CP}}$ at the start of m^{th} OFDM symbol. Plugging $n = mN_{\text{OFDM}} + N_{\text{CP}}$ in Eq (9.32) after removal of Term 1 through a DFT at the Rx, we get[†]

$$\text{Phase rotation} = \frac{2\pi}{N} \left\{ v_0(1 + \xi_0) + k\xi_0 \right\} (mN_{\text{OFDM}} + N_{\text{CP}})$$

We can simplify our view of this effect if we only consider this phase rotation from the previous OFDM symbol to the present and denote it by $\varphi_{\Delta}[m]$. Assuming the same data symbols $a_{m-1}[k] = a_m[k]$ (which is the case for pilot subcarriers) and slowly varying channel $H_{m-1}[k] \approx H_m[k]$,

$$\begin{aligned} \varphi_{\Delta}[m] &= \frac{2\pi}{N} \left\{ v_0(1 + \xi_0) + k\xi_0 \right\} \left\{ mN_{\text{OFDM}} + N_{\text{CP}} \right\} - \\ &\quad \frac{2\pi}{N} \left\{ v_0(1 + \xi_0) + k\xi_0 \right\} \left\{ (m-1)N_{\text{OFDM}} + N_{\text{CP}} \right\} \\ &= \frac{2\pi}{N} \left\{ v_0(1 + \xi_0) + k\xi_0 \right\} N_{\text{OFDM}} \\ &= 2\pi \left\{ v_0(1 + \xi_0) + k\xi_0 \right\} \frac{N + N_{\text{CP}}}{N} \end{aligned}$$

Let us denote

$$\lambda = \frac{N + N_{\text{CP}}}{N}$$

so that the above equation becomes

$$\varphi_{\Delta}[m] = 2\pi\lambda \left\{ v_0(1 + \xi_0) + k\xi_0 \right\} \quad (9.33)$$

i.e., $\varphi_{\Delta}[m] \propto k\xi_0$. While this looks similar to the STO case where the phase rotation was equal to $k\varepsilon_0$, the difference is that *$k\xi_0$ is just the rotation between two consecutive OFDM symbols* and keeps increasing with time. Ref. [42] draws the data symbols rotation at a subcarrier level corresponding to $v_0(1 + \xi_0) + k\xi_0$ in an elegant manner which is the inspiration behind Figure 9.43. Observe that the constant frequency shift is given by $v_0(1 + \xi_0)$ while maximum rotation is experienced by the subcarriers at the edges of the spectrum, namely $k = -N/2$ and $k = N/2 - 1$.

In addition to the above, there is a sliding of the Rx FFT window due to the SCO between the Tx and Rx clocks. The details of such a window drift are similar to what was encountered in the context of single-carrier systems in Section 7.10. A similar timing locked loop can stuff or delete an extra sample while slowly correcting for the SCO. For this purpose, an SCO error detector is needed which is what we discuss below. In the documentation of QIRX software defined radio, Ref. [46] presents an excellent description of OFDM in general and a real world example of sampling clock misalignment in particular.

[†]In this complete discussion on OFDM synchronization topics, we are ignoring extra noise arising from ICI and/or ISI for the sake of simplicity.

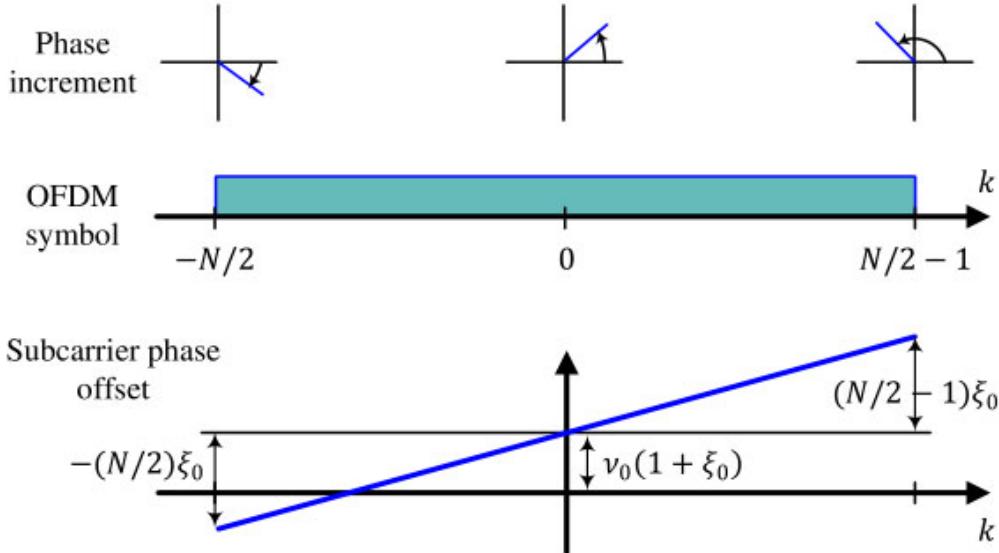


Figure 9.43: Data symbols riding on the subcarriers experience a rotation due to residual CFO and sampling frequency offset

Joint Sampling Clock and Residual CFO Synchronization Strategy

Now we are in a position to devise an SCO compensation technique that also helps in correcting any residual CFO in the FFT output. Recall from Eq (9.30) (reproduced below) that the *correlation magnitude* between m^{th} OFDM symbol and $(m-1)^{\text{th}}$ OFDM symbol at the FFT output was exploited to estimate the integer CFO.

$$\text{corr}[m] = \sum_{k_P} Z_m[k_P] \cdot Z_{m-1}^*[k_P]$$

where k_P denotes the set of pilot subcarriers. In the current context, we will use the *correlation phase* but in a slightly different manner.

First, notice that the phase of $\text{corr}[m]$ above for a single k_P is just the phase difference between two consecutive OFDM symbols, m and $m-1$. Fortunately, we already have the phase for one term of this correlation from Eq (9.33) when we considered the phase rotation from one OFDM symbol to the next and denoted it by $\varphi_\Delta[m]$.

$$\varphi_\Delta[m] = 2\pi\lambda \left\{ v_0(1 + \xi_0) + k\xi_0 \right\} = \angle \left\{ Z_m[k_P] \cdot Z_{m-1}^*[k_P] \right\} \quad (9.34)$$

The problem here is that this is only one expression and there are two unknowns, v_0 and ξ_0 . A single equation with two unknowns cannot have a unique solution.

To overcome this limitation, we divide the available set of pilot subcarriers in two subsets, one in the positive half of the spectrum and the other in the negative half and denote them by k_P^+ and k_P^- , respectively (where the subscript P stands for pilot). For an ease of understanding, we again consider the IEEE 802.11a standard from Example 9.1 in which there are 4 pilot subcarriers at locations -21 , -7 , $+7$ and $+21$.

The positive and negative pilots subcarrier sets are

$$k_p^+ = \{+7, +21\}$$

$$k_p^- = \{-7, -21\}$$

We draw these subcarriers in Figure 9.44 which is an *IQ* version of Figure 9.43 that depicted a 2D plot. Observe that they are symmetrically and uniformly distributed in both halves of the spectrum around the DC subcarrier ($k = 0$).

The reason we have redrawn this figure here is to clarify one point. It seems that a CFO should shift the spectrum to the right or left as this is what usually happens. However, notice that we are not drawing the spectrum of the Rx OFDM signal $z[n]$. Instead, we are drawing the spectrum of the correlation between pilot subcarriers k_p of m^{th} OFDM symbol $Z_m[k_p]$ and $(m-1)^{th}$ OFDM symbol $Z_{m-1}[k_p]$ at the FFT output. In view of Eq (9.34), the CFO then just imparts a constant phase offset proportional to λ . If SCO $\xi_0 = 0$, then this portion of the phase offset contributed by the CFO is shown by red dotted lines in Figure 9.44 which is an equal *phase rotation* (but not a spectral shift) for all subcarriers! It is only after the addition of the SCO induced term (proportional to subcarrier index k) that the subcarriers in the subset k_p^+ rotate in the anticlockwise direction while those in the subset k_p^- rotate in the clockwise direction in proportion to λ .

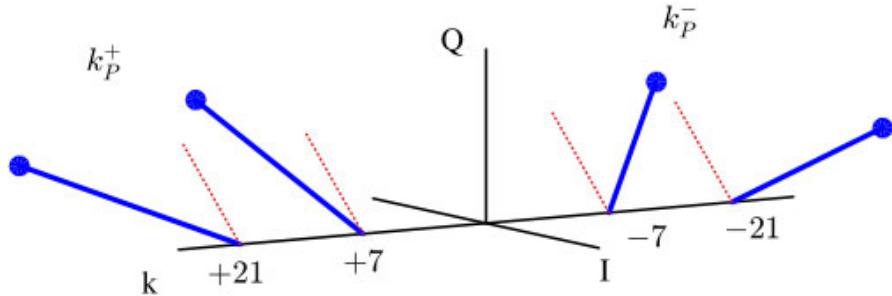


Figure 9.44: Division of pilot subcarriers k_p into two sets, one in the positive half and the other in the negative half of the spectrum. The phase of the correlation between one OFDM symbol to the next is also drawn depicting a residual CFO and a sampling frequency offset

Next, we examine the phase term of *one such subcarrier* in each half.

$$\varphi_\Delta^+[m] = 2\pi\lambda\nu_0(1 + \xi_0) + 2\pi\lambda k_p^+ \xi_0$$

$$\varphi_\Delta^-[m] = 2\pi\lambda\nu_0(1 + \xi_0) + 2\pi\lambda k_p^- \xi_0$$

Here, each positive subcarrier k_p^+ has a corresponding negative subcarrier k_p^- (for example, $+7$ and -7). Therefore, we can add the above two equations to cancel their second terms.

$$\varphi_\Delta^+[m] + \varphi_\Delta^-[m] = 2\pi\lambda 2\nu_0(1 + \xi_0) \quad (9.35)$$

Similarly, we can subtract the above two equations to cancel the first terms containing ν_0 .

$$\varphi_\Delta^+[m] - \varphi_\Delta^-[m] = 2\pi\lambda(k_p^+ + k_p^-)\xi_0 \quad (9.36)$$

With this understanding in place, we have two choices to average out the noise.

- We can measure the *angles* of all the positive pilot subcarriers and add them together. Similarly, we can measure the *angles* of all the negative pilot subcarriers separately and add them together.
- As an alternative, we can add the positive pilot subcarriers first (instead of their angles) and take the angle of the result later. Similarly, we can add the negative pilot subcarriers first (instead of their angles) and take the angle of the result later.

The latter strategy is clearly superior to the former because measuring the angle is a non-linear operation and hence it is better to sum the subcarriers first to average out the actual noise contribution instead of the noise angles.

So we redefine $\varphi_{\Delta}^+[m]$ and $\varphi_{\Delta}^-[m]$ to construct the following two different correlation phases from one OFDM symbol to the next.

$$\begin{aligned}\varphi_{\Delta}^+[m] &= \angle \left\{ \sum_{k_p^+} Z_m[k_p^+] \cdot Z_{m-1}^*[k_p^+] \right\} \\ \varphi_{\Delta}^-[m] &= \angle \left\{ \sum_{k_p^-} Z_m[k_p^-] \cdot Z_{m-1}^*[k_p^-] \right\}\end{aligned}$$

The final estimates of the ξ_0 and v_0 are given as follows, see Eq (9.35) and Eq (9.36).

$$\begin{aligned}\hat{v}_0 &= \frac{1}{2\pi\lambda} \cdot \frac{1}{2(1+\hat{\xi}_0)} \cdot \left\{ \varphi_{\Delta}^+[m] + \varphi_{\Delta}^-[m] \right\} \\ \hat{\xi}_0 &= \frac{1}{2\pi\lambda} \cdot \frac{1}{K/2+1} \cdot \left\{ \varphi_{\Delta}^+[m] - \varphi_{\Delta}^-[m] \right\}\end{aligned}\tag{9.37}$$

Here, K denotes the total number of used subcarriers which is less than the total number of subcarriers N . For example, as evident from Table 9.1, $K = 48 + 4 = 52$ in IEEE 802.11a WiFi standard for OFDM systems. It appears in the estimate $\hat{\xi}_0$ because of the summation over terms involving the subcarrier index k , either in the form of k_p^+ or k_p^- . When pilots are symmetrically and uniformly distributed around DC, their average on each side yields the desired term.

The final question is that whether a feedback or a feedforward strategy should be adopted for this purpose. With high sample rates of the modern OFDM systems, it is possible to estimate and compensate for the SCO in a feedforward manner. In fact, many receiver systems designed for OFDM estimate and compensate for the SCO on a block by block basis by taking advantage of its frame structure.

Otherwise, a feedback decision-directed based approach can also be employed in a timing locked loop where the above expressions are fed into their respective Proportional plus Integrator loop filters. Remember that until now for the case of pilot subcarriers $a_{m-1}[k] = a_m[k]$ and furthermore, we assumed $H_{m-1}[k] \approx H_m[k]$. In the decision-directed case, the first assumption does not hold true and the data decisions can be used to nullify their impact. Using $Z_m[k] = a_m[k] \cdot H_m[k]$, we get

$$\begin{aligned}Z_m[k] \cdot Z_{m-1}^*[k] \cdot \hat{a}_m^*[k] \cdot \hat{a}_{m-1}[k] &= \\ a_m[k] H_m[k] \cdot a_{m-1}^*[k] H_{m-1}^*[k] \cdot \hat{a}_m^*[k] \cdot \hat{a}_{m-1}[k] &\approx |a_m[k]|^2 \cdot |a_{m-1}[k]|^2 \cdot |H_m[k]|^2\end{aligned}$$

The only contribution towards the phase of this expression comes from the residual CFO and SCO, as in the scenario discussed before. Notice the use of index k instead of k_p which implies that all subcarriers are employed for tracking purpose thus improving the synchronization performance.

9.7 Channel Estimation

We covered some techniques to estimate a wireless channel in the context of single-carrier systems in Section 8.2. In OFDM systems, each subcarrier acts as an independent channel as long as there is no Inter-Carrier Interference (ICI) left in the synchronized signal. The options of both a training sequence and individual pilots are available for channel estimation and the choice between the two depends on time variation rate of the channel as well as the computational complexity. Many systems acquire the channel through the preamble while employ the pilots for channel tracking. The discussion in this section is mostly based on Ref. [47].

From Eq (9.13), we can write the signal after the FFT at the Rx as a product between the data symbols $a[k]$ and channel frequency response $H[k]$.

$$Z[k] = a[k] \cdot H[k] \\ \text{for each } k = -N/2, \dots, N/2 - 1$$

This is the frequency domain representation of the Rx samples that will now be exploited for the estimation purpose.

Training Based Estimation

In some systems, training OFDM symbols are periodically sent in the Rx signal so that the channel estimate can be updated at the same rate. In such a scenario, all subcarriers of an OFDM symbol are occupied by the training as shown in Figure 9.45 where OFDM symbols for $m = 0$ and $m = 4$ are the training symbols. After estimating the channel for the initial training sequence, it is common to assume a time-invariant channel (and hence valid estimates for subsequent OFDM data symbols) until the next training sequence arrives. Then, the channel estimate is updated and used again for subsequent OFDM symbols and this process is known as piecewise constant interpolation. For example, training based channel estimation is adopted in IEEE 802.11a/b/g and fixed WiMAX systems.

Since the symbols $a[k]$ are known in this context, we can write

$$\hat{H}[k] = \frac{Z[k]}{a[k]} \\ \text{for each } k = -N/2, \dots, N/2 - 1 \quad (9.38)$$

This is commonly known as **Least Squares (LS)** channel estimation which in case of OFDM subcarriers is also Maximum Likelihood (ML) solution. Depending on the channel characteristics and the system requirements, this can suffice alone or can be computed as a first estimate before refining it through a more advanced technique.

From Figure 9.45, it is evident that frequency selectivity of the channel is not a problem due to its division into underlying flat fading subcarriers. Even if the channel

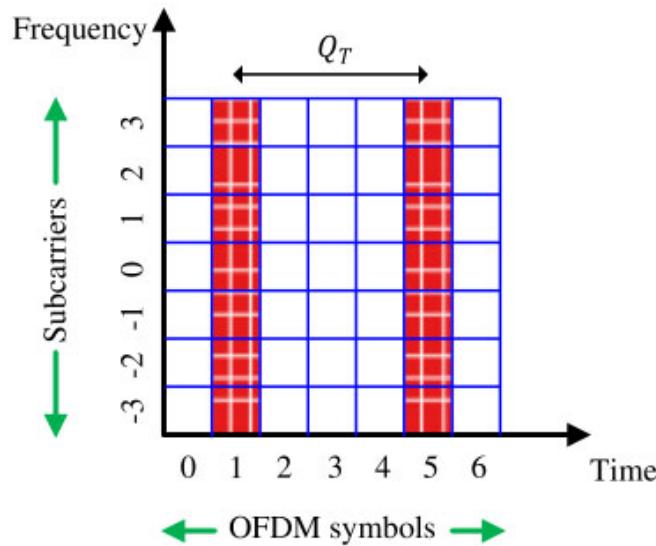


Figure 9.45: Training sequence occupies all the subcarriers in frequency domain for one OFDM symbol

is drastically different from one subcarrier to the next, it can satisfactorily be estimated by probing on all frequencies.

On the other hand, it is the time domain where the gaps are left and hence susceptible to time selectivity of the channel. Ideally, the training sequences need to be as far apart as possible to minimize the overhead and maximize the data rate. To find out the maximum required spacing between them, we refer to the sampling theorem. Let us denote by Q_T the period between two training sequences which needs to be at least as frequent as the coherence time of the channel T_C which is the inverse of the Doppler spread B_{Dop} (normalized by subcarrier spacing Δ_F), see Eq (8.30). Therefore, we have

$$Q_T \leq \frac{1}{B_{\text{Dop}}/\Delta_F} = \frac{1}{B_{\text{Dop}} T_u}$$

where the scaling factors have been ignored. Since the OFDM symbols between the training are used to deliver the data symbols, a decision-directed approach can also be employed to smooth the channel tracking between the training sequences.

Pilot Based Estimation

The other option is to insert regularly spaced pilots in some subcarriers that are kept occupied for each OFDM symbol. In this way, the pilots are multiplexed with the data and this arrangement is shown in Figure 9.46. For example, the subcarriers $k = -2$ and $k = 1$ are taken by the pilots in Figure 9.46. Then, Eq (9.38) can also be used for pilot-aided channel estimation for values of k corresponding to the pilot symbols.

In this case, time variations of the channel are not a problem since the pilots are present for each OFDM symbol. On the other hand, there is a gap between them in the frequency domain and hence they are susceptible to frequency selectivity of the

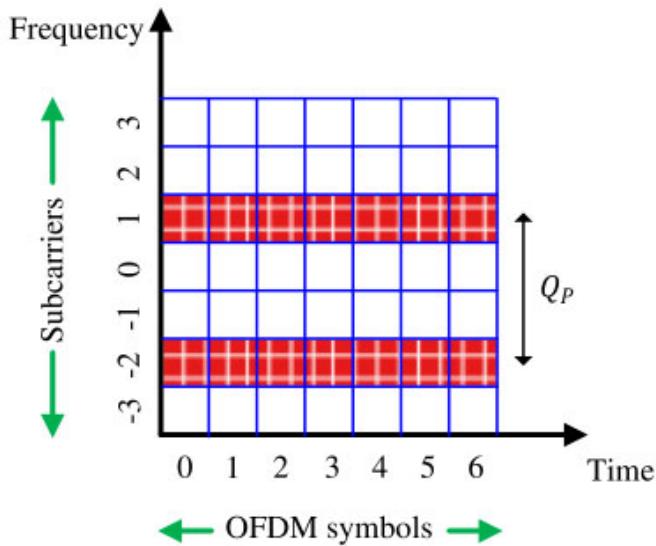


Figure 9.46: Pilot symbols occupy only fixed subcarriers in frequency domain for all OFDM symbols

channel. For this purpose, the maximum gap between two pilots, Q_P is governed by the coherence bandwidth B_C . We know that B_C is inversely proportional to the channel delay spread T_{Del} (which is T_{Del}/T_u samples) leading to the following conclusion.

$$Q_P \leq \frac{1}{T_{\text{Del}}/T_u} = \frac{1}{T_{\text{Del}}\Delta_F}$$

As before, the data symbols between the pilots can be exploited in a decision-directed manner to improve on the channel estimates. Some comments are now in order.

Optimal pilot locations: This is determined by the channel fading characteristics in time and frequency domains. Our objective is to insert a minimum number of pilots in the OFDM time-frequency grid. A larger number is a waste of power while a smaller number is a performance loss due to insufficient channel sampling. The tradeoffs involved are accuracy (the more, the better) and spectral efficiency (the lesser, the better). Once a certain number of pilots is determined, it has been reported that the minimum mean square error is obtained when the pilots are equispaced with maximum distance from each other.

Power and modulation: Many systems transmit more power at the pilot subcarriers than the data subcarriers. Also, a lower-order modulation scheme, such as BPSK, is well suited for pilot assignment since a higher data rate is not an objective and maximum power can be allocated to BPSK symbols instead of dividing it into M PSK symbols for normalization, thus implying a longer range.

Interpolation: Interpolation between the pilot subcarriers can be carried out in both time and frequency domains. A piecewise constant interpolation technique that implies a constant channel between two training sequences can be employed.

In the case of pilot subcarriers, linear interpolation offers better performance without adding much complexity to the block. Linear interpolation was studied in detail in the context of timing synchronization in Section 7.8. Assume that two pilot measurements are available at subcarriers $k = k_1$ and $k = k_2$ where

$$k_2 - k_1 = Q_P$$

Then, we can find the channel estimate for any other k by linear interpolation.

$$\hat{H}(k) = \mu_k \hat{H}[k_1] + (1 - \mu_k) \hat{H}[k_2], \quad k_1 < k < k_2$$

where μ_k is given by

$$\mu_k = \frac{k}{Q_P}$$

It can be viewed as a 2 tap filter applied to the channel estimate sequence after upsampling the channel estimates at pilot locations by Q_P . This is shown in Figure 9.47. Better interpolation techniques such as Wiener, Gaussian, cubic or spline are also frequently applied for this purpose. Finally, taking an iFFT/FFT pair also accomplishes pilot interpolation if the FFT size is divisible by the pilot spacing.

Another option is to arrange the pilots on adjacent subcarriers in adjacent OFDM symbols such that they cover the time-frequency grid in a diagonal fashion. This helps in sampling the channel at all times and all subcarriers in a periodic manner.

Transform Domain Techniques

There are scenarios in which the number of taps in the channel impulse response is much less than the Cyclic Prefix (CP) length. This extra information helps in refining an initial channel estimate (such as the Least Squares (LS) estimate) by concentrating on the channel taps within the delay spread T_{Del} and nullifying the effect of the noise outside the farthest channel tap.

Consider an OFDM system with the CP length N_{CP} and the number of subcarriers N operating in a channel with length $N_{\text{Tap}} + 1$. Let $\hat{H}[k]$ for each subcarrier k denote the Least Squares estimate for the corresponding channel gain. Next, we take the iFFT of $\hat{H}[k]$ and denote it by $\hat{h}[n]$ that represents an estimate of the channel impulse response $h[n]$. This sequence in time domain is given by

$$\hat{h}[n] = h[n] + \text{noise}, \quad n = 0, 1, \dots, N - 1$$

Since the actual channel length is $N_{\text{Tap}} + 1$, all the taps after N_{Tap} consist of noise only. Here, we form a new channel estimate $\hat{h}_{\text{FFT}}[n]$ by removing all the taps from $\hat{h}[n]$ beyond N_{Tap} .

$$\hat{h}_{\text{FFT}}[n] = \begin{cases} h[n] + \text{noise} & n = 0, 1, \dots, N_{\text{Tap}} \\ 0 & n = N_{\text{Tap}} + 1, \dots, N - 1 \end{cases}$$

Now when we perform an FFT of the above sequence to go back to the frequency domain, the resultant channel estimates are free of noise beyond $n = N_{\text{Tap}}$ and consequently yield better performance.

$$\hat{H}_{\text{FFT}}[k] = \text{FFT}\{\hat{h}_{\text{FFT}}[n]\}$$

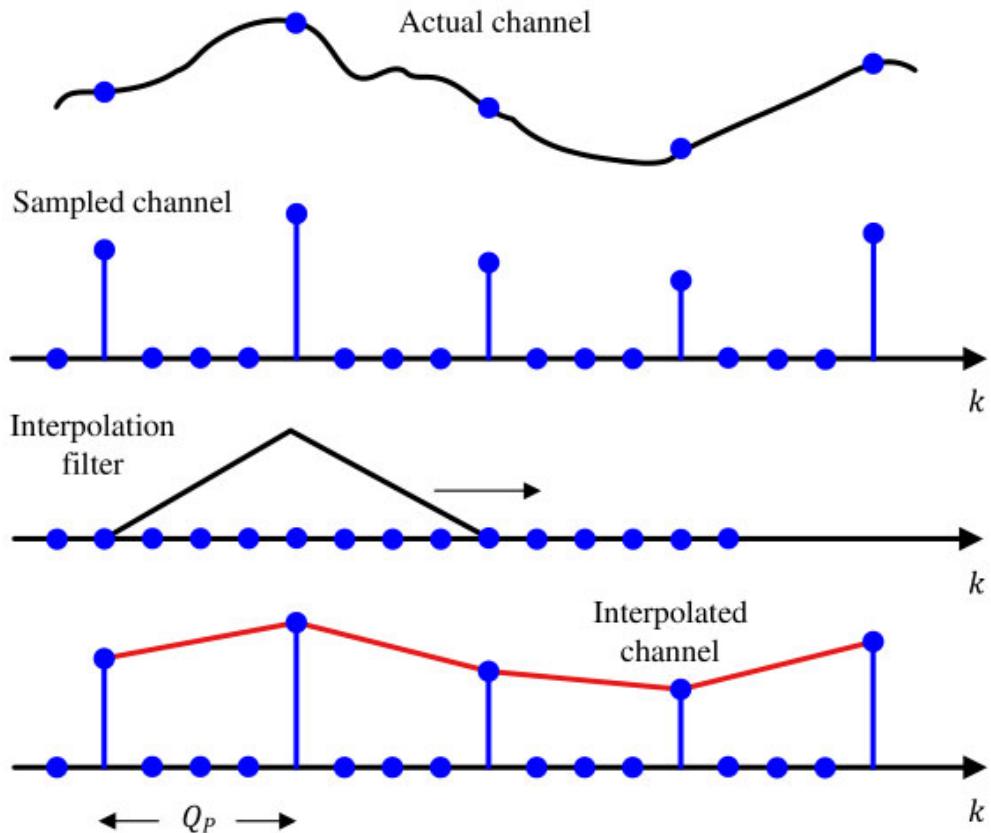


Figure 9.47: Linear interpolation between pilot subcarriers

A block diagram for the implementation of such an approach is drawn in Figure 9.48. Although a similar concept can be applied to some other transforms, the FFT is particularly attractive due to its efficient implementation.

Decision-Directed Channel Estimation

Due to its widespread use in single-carrier systems, decision-directed channel estimation was one of the earliest methods explored for OFDM systems. It is particularly useful because after the initial estimates, the channel coefficients can be updated with the help of decisions from data symbols. The basic idea is to employ the channel estimates from the previous OFDM symbol to detect the data in the current OFDM symbol.

- Assume that the channel is slowly varying from one OFDM symbol to the next. Let $Z_m[k]$ be the m^{th} Rx signal after the FFT and $\hat{H}_{m-1}[k]$ be the channel estimate for $(m-1)^{\text{th}}$ OFDM symbol. Then, it can be employed to detect the current data symbols as

$$\hat{a}_m[k] = \frac{Z_m[k]}{\hat{H}_{m-1}[k]}$$

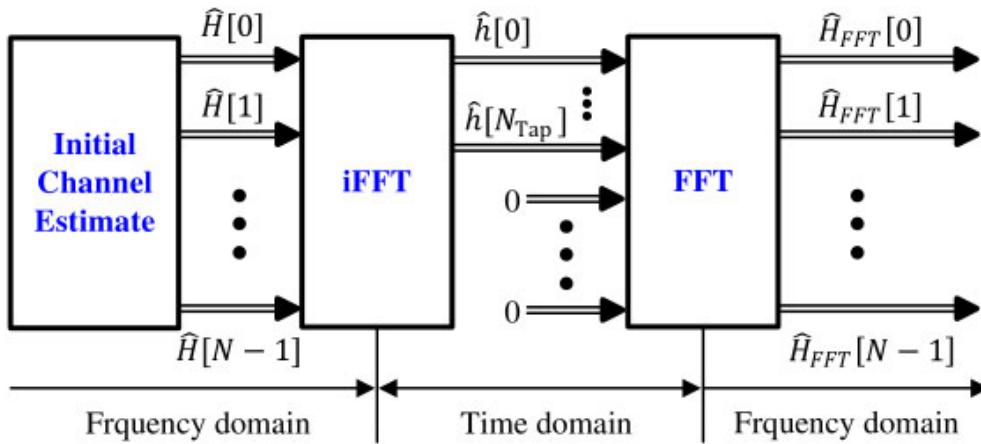


Figure 9.48: Channel estimation through the iFFT-FFT operations

- Next, the symbol candidate $\hat{a}_m[k]$ can be used for estimating the channel $\hat{H}_m[k]$ at time m as

$$\hat{H}_m[k] = \frac{Z_m[k]}{\hat{a}_m[k]}$$

As another option, the data symbol decision sign $\{\hat{a}_m[k]\}$ can also be used for this purpose to reduce the effect of noise.

$$\hat{H}_m[k] = \frac{Z_m[k]}{\text{sign}\{\hat{a}_m[k]\}}$$

There are two main problems with this kind of decision-directed approach.

Channel estimate expiry: For a quickly varying channel on a time scale of an OFDM symbol, the channel estimates at time $m - 1$ are no longer valid at time m . Consequently, the detected symbols $\hat{a}_m[k]$ increasingly suffer from errors.

Error propagation: Wrong data symbols fed back to estimate the current channel gives rise to error propagation which renders the system dysfunctional. This is a problem associated with all decision-directed techniques whether in the context of equalization or synchronization.

To resolve such issues, significant improvement over the decision-directed channel estimation as well as training and pilot based methods described above can be gained through involving a powerful error correcting code for the channel estimation process. Such codes operate in a recursive manner and improve the reliability of symbol decisions in each iteration. Inserting the channel estimation process within such a feedback loop improves the quality of the channel estimates which in turn refines the reliability of the decisions.

Now when we have discussed several synchronization and channel estimation strategies, it is interesting to see the direction where the high rate wireless systems are heading. For this purpose, we compare IEEE 802.11a standard specifications studied in the examples of this chapter to later and modern IEEE 802.11 standards

such as n, ac and ax. Such a comparison is given in Table 9.2. Notice that better frequency synchronization enables a closer spacing of the subcarriers, 4 times closer in ax for example. Moreover, the fundamental rate increase comes from either increasing the channel bandwidth (160 MHz), the number of antennas (8) or (to some extent) a higher-order modulation scheme (1024-QAM).

Table 9.2: A comparison of IEEE 802.11 systems

	802.11a	802.11n	802.11ac	802.11ax
Frequency (GHz)	5	2.4, 5	5	2.4, 5
Bandwidth (MHz)	20	40	160	160
Δ_f (kHz)	312.5	312.5	312.5	78.125
T_u (μ s)	3.2	3.2	3.2	0.8, 1.6, 3.2
MIMO	1	4	8	8
Multi-User MIMO	No	No	Downlink	Uplink, Downlink
Modulation	64-QAM	64-QAM	256-QAM	1024-QAM
Data rate (Mbps)	54	600	1000-3000	10000

9.8 The Small Picture

In high rate wireless systems, multipath effect becomes the major bottleneck due to resource intensive time domain equalization techniques. This problem is overcome in time domain by transmitting the information in several low rate parallel streams which segments in frequency domain the available bandwidth into many parallel frequency flat channels. Since convolution in time is multiplication in frequency, equalization for each narrow slice requires just a single division operation, significantly reducing the computational load of the equalizer.

Viewed from another angle, OFDM employs a set of complex sinusoids in time domain that *samples the channel in frequency domain* at discrete intervals in frequency. See Note 3.5 for the dual phenomenon in time domain. I find this perspective very fascinating and beautiful.

