AN5664

Application note

RSSI and SNR for LoRa® modulation on STM32WL5x/Ex microcontrollers

Introduction

This application note provides an overview of LoRa® modulation and demodulation concepts on STM32WL5x/Ex microcontrollers. This document describes how this modulation technique operates below the system noise floor through the CSS (chirp spread spectrum).

General pseudorandom spread-spectrum techniques are used to introduce the working principles of spread-spectrum as related to CSS.

1 General information

The STM32WL5x/Ex microcontrollers are based on the Arm®Cortex®-M processor.

Note: Arm is a registered trademark of Arm Limited (or its subsidiaries) in the US and/or elsewhere.

arm

Table 1. Acronyms

Acronym	Definition
BER	Bit error rate
CSS	Chirp spread spectrum
CW	Continuous wave
DSSS	Direct sequence spread spectrum
FFT	Fast Fourier transform
FHSS	Frequency hop spread spectrum
FSK	Frequency-shift keying
IF	Intermediate frequency
LoRa	Long-range radio technology
NF	Noise figure
PA	Power amplifier
PER	Packet error rate
PG	Processing gain
PN	Pseudo random
RSSI	Received signal strength indicator
SNR	Signal-to-noise ratio

2 Spread-spectrum concepts

The LoRa modulation is a spread-spectrum technique. The spreading function is performed through the implementation of two methods:

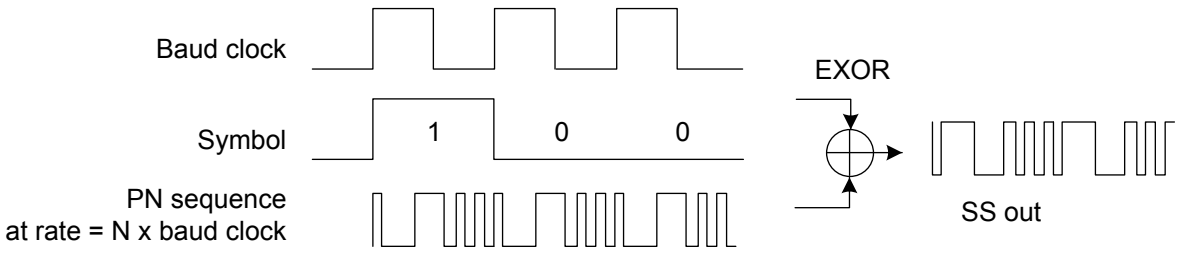
- Sweeping a linearly increasing frequency, between frequency-start and frequency-stop boundaries. This allows the low SNR operation to be used.
- Frequency hop method that enables multiple access from many end-node configured devices

The combination of these two schemes provides a robust communication reliability.

This document focuses on the low SNR operation principles. From an analysis point of view, the behavior is close to DSSS (direct sequence spread spectrum). Although the implementation method is completely different for both modulation types, systems can be described using similar mathematical properties.

In a DSSS system, the message signal, at R_b rate, is multiplied (EXOR digital equivalent) by a high-rate pseudo-random (PN) sequence at R_{chip} rate. The time domain process is shown in the figure below.

Figure 1. Time-domain process of a DSSS baseband generation



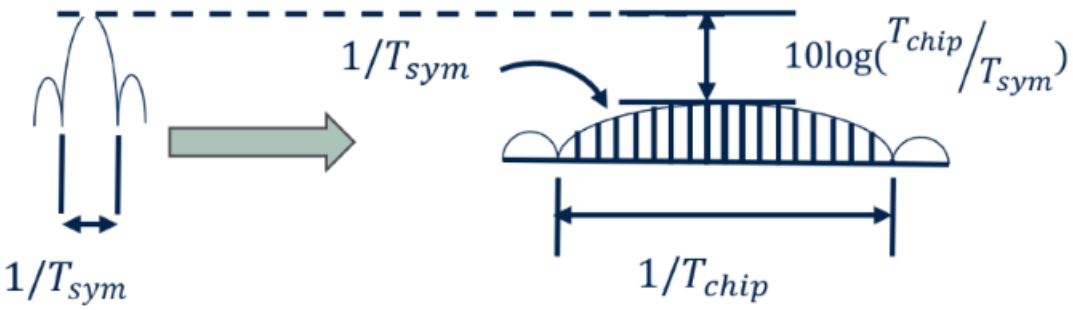
The resulting bandwidth is increased by the 1/ R_{chip} rate, and the resultant power (normalized to R_b) is reduced by the following ratio, called spreading bandwidth or PG (processing gain):

$$PG = 10 \times Log_{10}\left(\frac{R_{chip}}{R_b}\right) \quad (1)$$

The relationship between the baseband signal and the resulting spread signal forms an important parameter when CSS properties are described in next sections.

The figure below summarizes the spectral effect of this spreading process. The power is maintained in this process: power in the resulting spreading process is equal to the original power in the message bandwidth.

Figure 2. Basic spread spectrum modulation

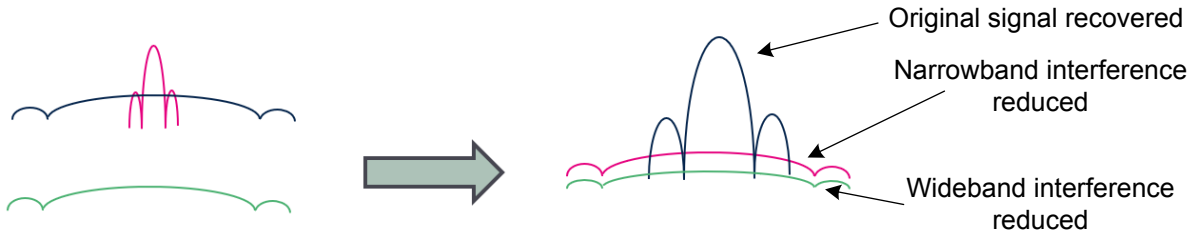


At the receiver, the demodulator applies the opposite mathematical equivalent process to recover the original message. The demodulator multiplies or correlates the incoming signal with the “exact same PN sequence” to remove the effects of the spreading process and recover the original transmitted signal.

During the demodulation process, any narrowband or wideband interference that happens to coexist with the chip bandwidth, is reduced by the PG factor. This is a key advantage of DSSS.

The figure below provides a very high-level summary of the demodulator process demonstrating how the various interfering signals are transformed at the demodulator input.

Figure 3. Demodulation process: effects on original signal and interference



Note: Several DSSS signals may coexist at the same RF center frequency. Provided that the PN sequences used for spreading are orthogonal, the demodulator can select the signal with a specific PN signal of interest. There is an SNR limitation regarding how many similar signals or the external interference level can coexist at the same RF center frequency. This subject is beyond the context of this application note.

3 Narrowband sensitivity

The below example demonstrates the spreading process for a LoRa CSS system, and how the negative SNR is modelled.

Some baseline prerequisites must be established. The first one is to define and determine the noise floor reference for the receiver front end (RF portion of the receiver, different from the final demodulator noise floor).

The receiver noise floor, including receiver bandwidth BW_{Rx} and receiver noise figure NF, is defined as:

$$Noisefloor = 10Log_{10}(kTB_{1Hz}) + 10Log_{10}(BW_{Rx}) + NF \quad (2)$$

where:

- k: Boltzmann constant
- T: temperature in Kelvin degrees (290 ºK room temperature)
- B_{1Hz}: bandwidth normalized to 1 Hz
- BW_{Rx}: bandwidth of the front-end receiver or system bandwidth
- NF: receiver noise figure (in dB)

With T = 290 °K, $–10Log_{10}(kTB_{1Hz})$ = –174 dBm/Hz. As BW_{Rx} = 125 kHz for this analysis, NF ranges between 6 and 7 dB, including all matching and filtering prior to the receiver input pins.

The minimum input noise floor is then –174 +51 + 7 = –116 dBm.

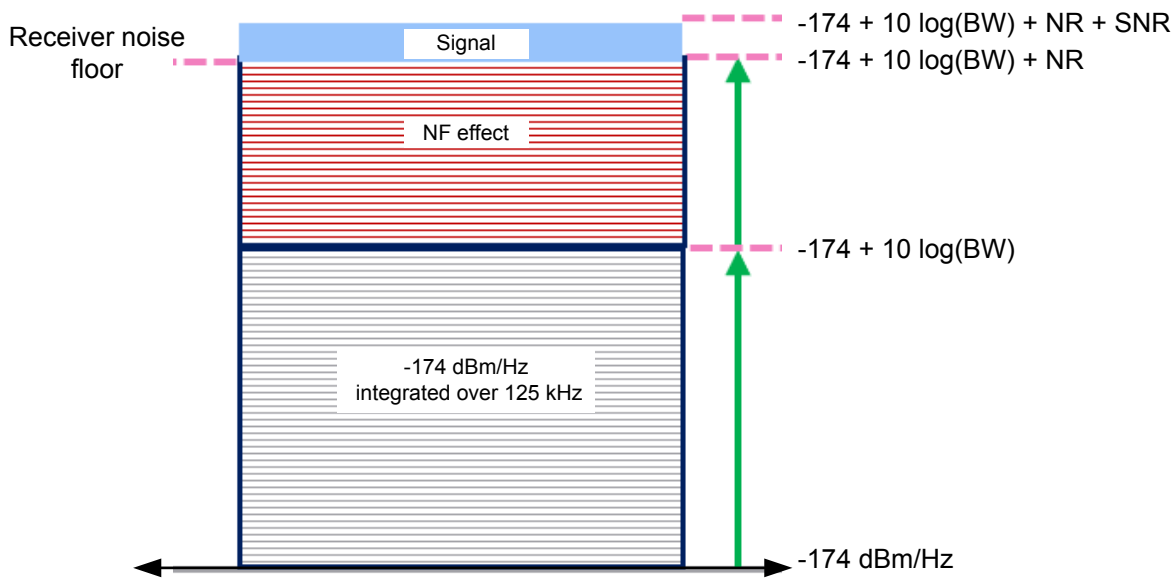
When modulation SNR requirements are introduced, the noise floor equation can be modified to reflect the minimum input at the receiver input, to guarantee a specific SNR within a specific bandwidth:

$$Sensitivity_{dBm} = -10Log_{10}(kTB_{1Hz}) + 10Log_{10}(BW_{Rx}) + NF + SNR \quad (3)$$

Where Sensitivity represents the minimum input at the receiver input to guarantee a specific BER (bit error rate) or PER (packet error rate).

The figure below summarizes the process as presented in the above equation. The receiver noise floor is established and the resultant signal power, and therefore, SNR is positive relative to this noise floor.

Figure 4. Link budget representation of $Sensitivity_{min}$ (BW = 125 kHz)



3.1 Lora: Applying processing gain and spreading factor effects

As described in previous sections, the spreading concept operates on a baseband signal, and spreads its energy over a wider bandwidth. The power of the signal itself is maintained even if the SNR changes based upon the measurement reference.

For LoRa, the processing gain, is defined as:

$$PG = 10 \times Log_{10}\left(2^{SF}\right) \quad in \quad dB \tag{4}$$

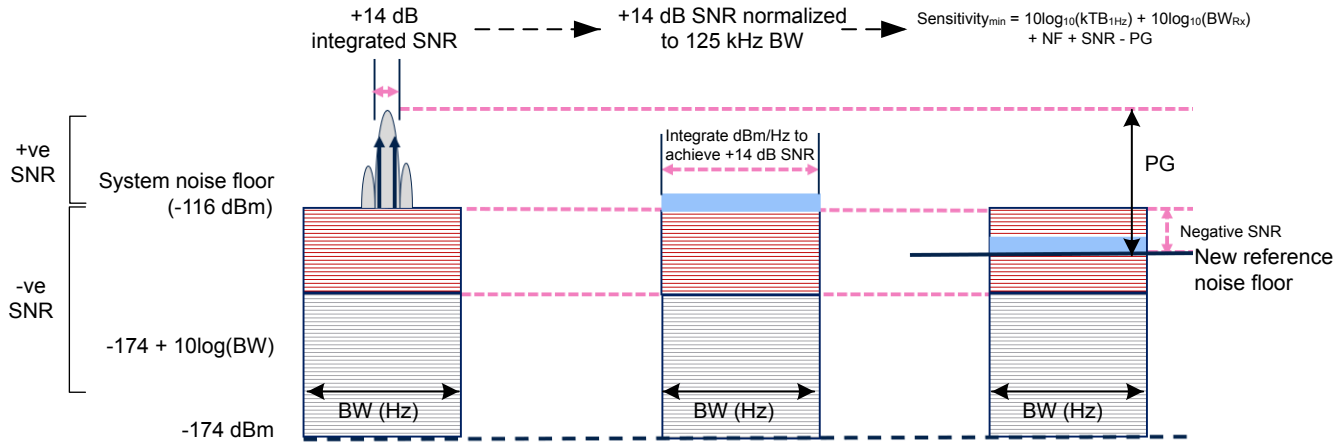
For consistency, SF = 7 for this example. The PG is then 21 dB, regardless of the system bandwidth.

The processing gain can be incorporated into the link budget equation as follows:

$$Sensitivity_{\min} = 10Log_{10}(kTB_{1Hz}) + 10Log_{10}(BW_{Rx}) + NF + SNR \quad - PG \tag{5}$$

The effect of this equation is demonstrated in the figure below that shows a narrowband signal with positive SNR. For analysis purposes, the narrowband power has been redrawn and normalized to fit within the receiver bandwidth. The integrated power remains the same in both cases .

Figure 5. SNR processing model



Depending upon the narrowband SNR and PG, Sensitivity$_{min}$ can reach a value which is less than the computed noise floor figure established in Figure 4 and equation (5). This effect is shown in the final drawing of the above figure.

The effect can be demonstrated by inserting numerical values for the key parameters. Assume the narrowband signal exhibits a SNR of 14 dB relative to the 125 kHz bandwidth noise floor. The receive RSSI measures also +14 dBm power over this bandwidth. As PG = 21 dB for SF = 7, the effect as computed in equation (5) is to provide a signal with a SNR reading of -7 or, a power reading, Sensitivity$_{min}$ of -123 dBm. Relative to the receiver bandwidth of 125 kHz, this represents a negative SNR situation. However, this is from the wideband reference perspective of the receiver bandwidth only and does not address the full narrowband system properties required to accurately describe the modulation and demodulation process. The concept of negative SNR is detailed in subsequent sections when the demodulation process is described.

3.2 RSSI and SNR

Sensitivity$_{min}$ computation must be considered when the input signal level falls below the input receiver sensitivity value computed by equation 5. In this case, with the 125 kHz bandwidth, this sensitivity level is -116 dBm. The receiver front end receiver power can be represented by the RSSI reading value (when greater than -116 dBm).

When the actual signal power falls below the -116 dBm value, Sensitivity$_{min}$ becomes a combination of the RF front end RSSI reading, and the SNR reading obtained by the baseband processing circuitry, The input power is therefore adjusted to reflect this important observation.

$$Sensitivity_{\min} = RSSI + SNR,\ SNR < 0 \tag{6}$$

The remaining question when considering this scenario is how to detect and recover a signal that appears to reside below the RF system noise floor. This is addressed in subsequent sections.

4 LoRa modulation

LoRa modulation is based upon CSS. For each symbol, a frequency sweep is generated with defined start and stop sweep limits. This is known as a chirp. The start and stop frequency are determined by the properties of the symbol mapping and SF. The process includes two steps: taking a narrowband symbol and spreading its energy over a finite bandwidth. It is described through the use of CSS general equations with some specific examples.

The LoRa modulation can be mathematically parametrized by the following exponential complex:

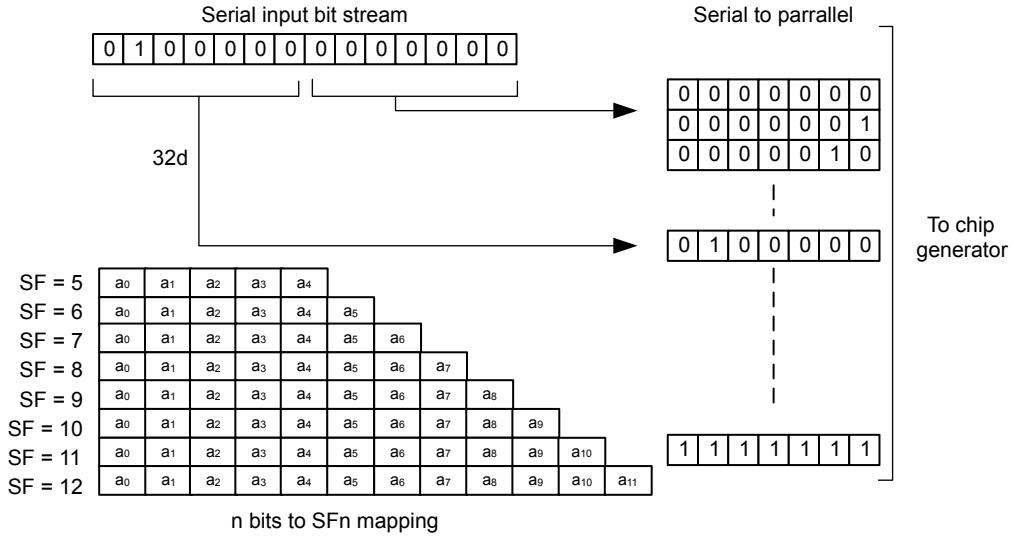
$$x(t) = \exp\left(\theta_0 + 2\pi \times \left(\frac{c}{2} \times t^2 + f_0 \times t\right)\right) \quad (7)$$

where:

- c = (f_1 - f_0) / T: frequency rate-of-change
- f_0: initial chirp starting frequency (set to 0 for this discussion)
- f_1: final sweep ending frequency, defined by the bandwidth (BW)
- BW = 125 kHz, 250 kHz, or 500 kHz (LoRaWAN specific)
- T = 2^{SF} / BW: basic chirp period and number of SF cycles required to fit within the specified bandwidth
- SF = 7, 8, 9, 10, 11, or 12 for many systems operating worldwide

In order to understand how these elements are integrated to form the modulation, a symbol representation must be defined. The figure below shows how bits are grouped into symbols. These symbols are used to set the properties of the chirp generator.

Figure 6. Binary n bits to SFn mapping

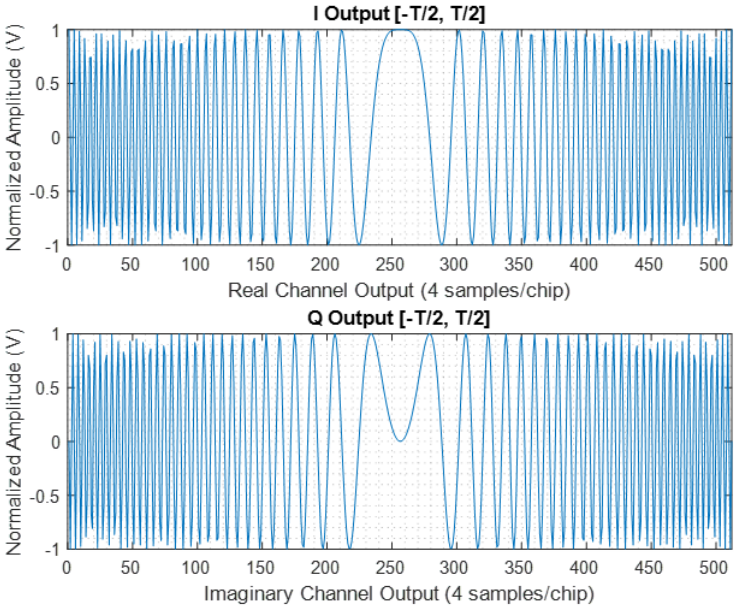


For SF = 7, bits are grouped into one 8-bit symbol that is used as frequency start and frequency stop mechanism for the chirp generator. Other SFs groups are also shown in this figure.

For SF = 7, 2^{SF} discrete chirp waveforms can be generated. The all zero symbol represents the up chirp symbol that is shown in the figure below.

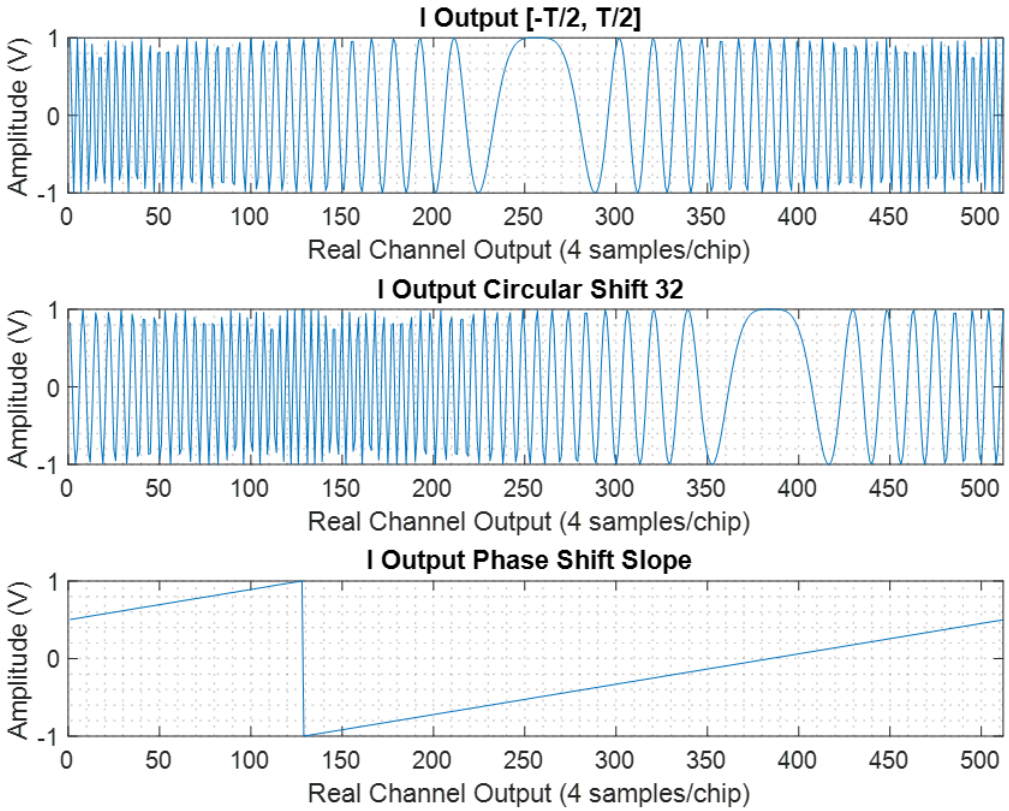
Note: This is in complex notation and symbol period spans from [-T/2, T/2]. The sampling rate has been increased to 4 samples/chip to display full waveform details.

Figure 7. UpChirp real and imaginary components



All other symbols can be generated from this base waveform. The process involves the modifying the pointer location within a circular buffer. The figure below shows symbol which represents 32 decimal. Only the I channel is shown in this example. The phase of the frequency slope is also shown. With 4 samples, chip, the location is shown to be 128 sample or a chip location of 128/4 = 32 relative to the zero reference.

Figure 8. Phase shifted output representing 32d (only I channel)



All other symbols can be generated from this base waveform. The pointer location is modified within a circular buffer. The figure below shows symbol which represents 32 decimal (only I channel shown), and the frequency slope phase. With four samples per chip, the location is 128 sample or a chip location of 128/4 = 32 relative to the zero reference.

Each symbol is therefore, chirped, or spread, within the 125 kHz bandwidth. Within this bandwidth, a complete chirp waveform must be transmitted. The rate at which this occurs per symbol is defined as:

$$\frac{1}{T} = \frac{BW}{SF} \quad \left(Hz\right) \tag{8}$$

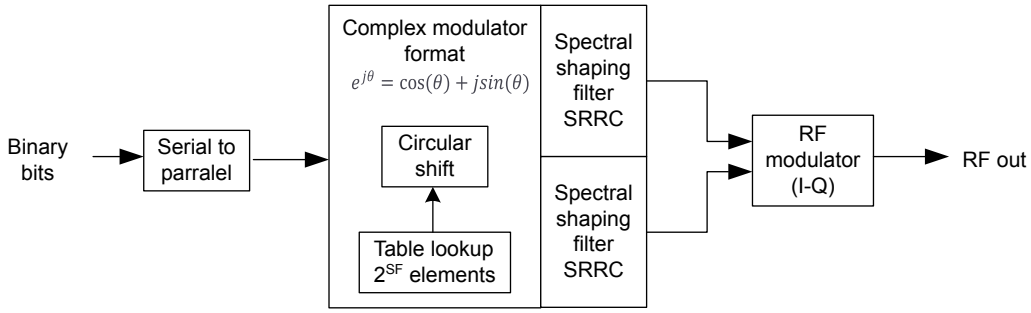
The chip rate is defined as $1/2^{SF}$, in chips/second.

The same process is implemented for all SFs.

Note: When SF increases, the processing gain increases. This provides a longer time on air and lower overall throughput in terms of symbols or bits/sec.

The figure below presents a high-level block diagram of the modulator. Blocks such as the whitening process and error correction encoder along with the CRC process have been omitted in order to show only the key signal processing elements. This figure represents the combined contribution of all the elements previously described within this document. The incoming binary bits are converted to symbols and modulated via the complex modulator. After spectral shaping, the output is modulated to a RF carrier frequency.

Figure 9. Modulator high-level representation

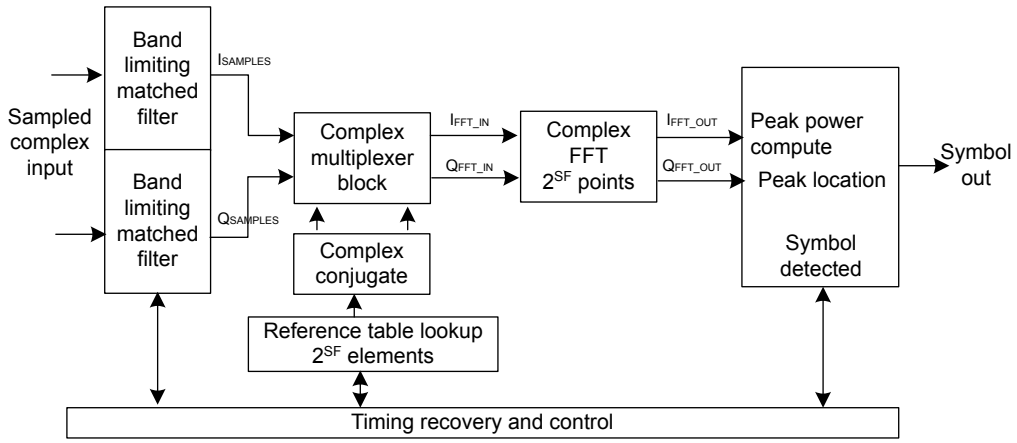


Note: Lora modulation is a constant envelop modulation: the signal amplitude is constant. A PA (power amplifier) can operate in non-linear region improving transmitter efficiency.

5 Demodulator functional description

The demodulator extracts and recovers the initial information sent from the modulator. It also removes any effects linked to channel impairments and interference due to similar signal structures. The signal processing model representation of the demodulator is presented in the figure below.

Figure 10. Demodulator high-level representation



The input is a complex digitized signal presented to the channel matched filters. It can be derived from using analog I and Q direct demodulation techniques, or from digitizing an IF (intermediate frequency) signal and deriving the I and Q elements through the appropriate sample manipulation of the digital IF.

The channel filter outputs are multiplied with a reference signal. This signal represents the complex conjugate of the base up chirp waveform:

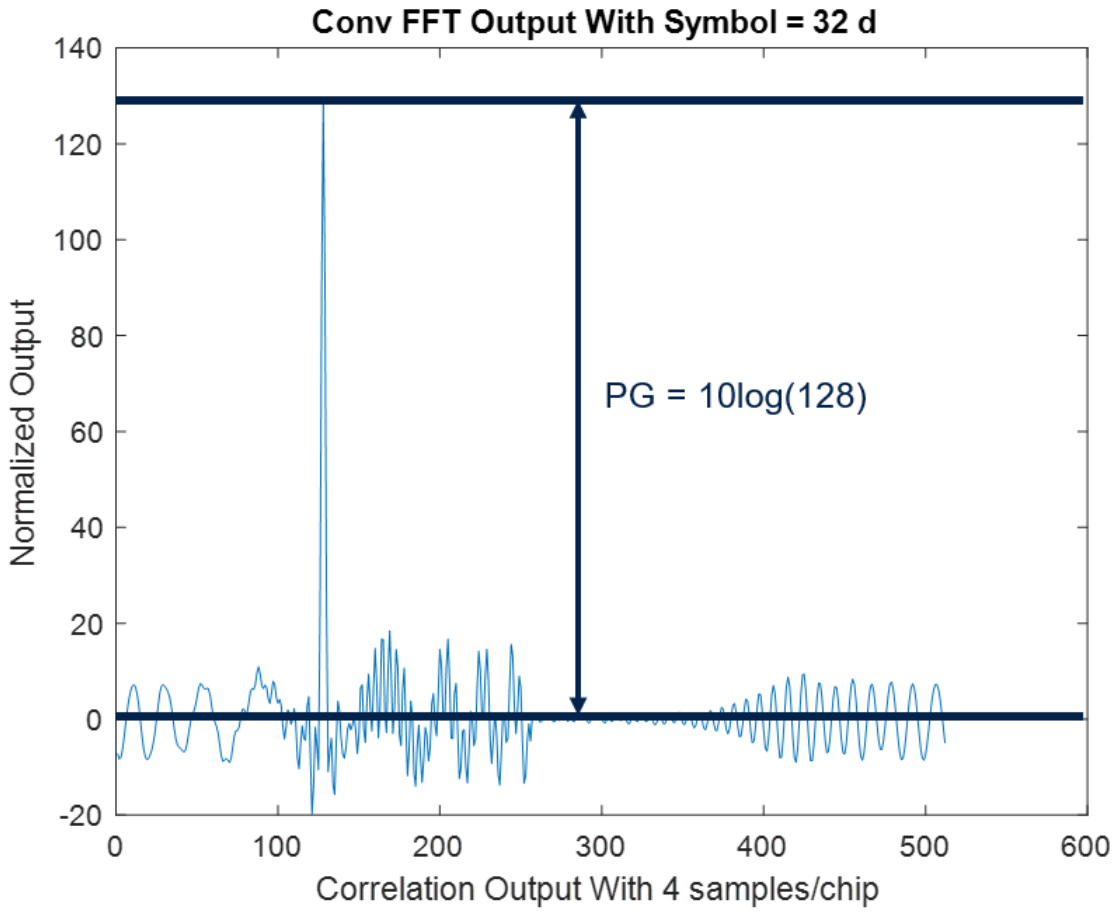
- This step removes the chirp modulation and provides a baseband waveform ready for FFT (fast Fourier transform) processing.
- This process can be configured as a correlator. By inserting an integrate and dump additional processing block, the timing circuitry can quickly adjust the position of the base sampling reference for the FFT block and also be used to detect the coarse initial position of the sync word.

The FFT block performs the key function of the entire process. The FFT output represents a waveform with a bin size given by the equation 8 and provides the information necessary to detect the peak power, the sync word location, start of received packet or burst. With this information, the entire received packet is demodulated.

The figure below shows an output example with the 32d symbol input symbol. This output represents the post processed output from the demodulator described in Figure 10. The post processed peak location is 128/4 = 32. The SNR and received power can also be computed through this process.

Note: In an ideal process, the output represents the processing gain, and can be related to peak power.

Figure 11. Peak detector and demodulator output



6 Conclusion

This application note presents the spread-spectrum concept and its relationship to chirp spectrum, an overview of basic spread spectrum techniques and their operation, the method for which spread spectrum is used within a CSS system, and the modulation/demodulation techniques.

As with many spread-spectrum techniques, the multiple access (many users share the same bandwidth) is a key advantage of such systems. The CSS provides multiple access operation with frequency hopping that allows many devices to share the same frequency spectrum.

Revision history

Table 2. **Document revision history**

Date	Version	Changes
6-Dec-2021	1	Initial release.
21-Jan-2025	2	Changed the scope of this document from STM32WL Series to STM32WL5x/Ex devices.

Contents

List of tables

List of figures