# GreenOFDM a new method for OFDM PAPR reduction : application to the Internet of Things energy saving

Jorge Gulfo Monsalve

UNIVERSITÉ GRENOBLE ALPES

# THÈSE

pour obtenir le grade de

**DOCTEUR DE LA COMMUNAUTÉ UNIVERSITÉ GRENOBLE ALPES**

École Doctorale **EEATS**
Électronique, Électrotechnique, Automatique et Traitement du Signal
Spécialité : **Signal, Image, Parole, Télécoms**

Arrêté ministériel : 25 mai 2016

Présentée par
**Jorge Luis GULFO MONSALVE**

Thèse co-dirigée par
**Jean-Marc BROSSIER** Professeur à Grenoble INP. GIPSA-lab. Directeur.
**Denis J.G. MESTDAGH**, PhD. GreenWaves Technologies SAS. Co-encadrant.

préparée au sein du laboratoire
**GIPSA-lab** (Grenoble Image Parole Signal Automatique)
et de l'entreprise
**GreenWaves Technologies SAS**

## GreenOFDM a new method for OFDM PAPR reduction
**Application to the Internet of Things energy saving**

## GreenOFDM une nouvelle méthode de réduction de PAPR en OFDM
**Application à l'économie d'énergie de l'IoT**

Thèse soutenue publiquement le **18 octobre 2019**,
devant le jury composé de :

**M. Benoît GELLER**
Professeur à ENSTA ParisTech. U2IS. Rapporteur
**M. Jean-François HELARD**
Professeur à l'INSA de Rennes. IETR. Rapporteur

**M. Christophe JEGO**
Professeur à Bordeaux INP. IMS. Examinateur (Président)
**Mme. Ghislaine MAURY**
Maître de Conférences à Grenoble INP. IMEP. Examinatrice
**M. Laurent ROS**
Maître de Conférences (HDR) à Grenoble INP. GIPSA-lab.
Examinateur

# Acknowledgements

In this section I want to thank all those people who directly or indirectly have helped me to progress in my thesis and have influenced me in a positive way.

Among those people the first to come to my mind are my thesis supervisors. This is why, first of all, I am very grateful to Jean-Marc and Denis. For all their patience and dedication. I have learned many things in this time by their side and they have helped me to grow. For this I will always be grateful.

I would like to thank too the members of the jury for accepting to evaluate this work and for all the enriching remarks.

In the places where this work was carried out, I have also come across people who have affected me positively. To all of them, both in GreenWaves Technologies and in the CICS team in GIPSA-lab, I thank you.

A mis amigos, a mi combo latino en Grenoble y a mis llaves del fútbol, gracias a todos por su apoyo y por darme ánimo todo el tiempo. También a los que me apoyan a distancia, con los que me hice amigos desde 2013, ustedes saben quiénes son. En Colombia también hay una parte de mi corazón, mis hermanos los HsH. Como dice la canción : gracias totales.

Por último, pero no menos importante, mi familia entera en Colombia, Barranquilla, Cartagena, Sincelejo, Copey, Valledupar, los llevo en mi corazón. A mami, a papi, gracias por todos los valores que me han inculcado, por la educación que es la mejor herencia que me pueden dejar y por ser mi apoyo incondicional. A mi amada esposa, a mi JE y a mi Sarita que está a punto de llegar ; gracias por tanto amor y por aguantar todo este tiempo, los amo. A ustedes dedico este logro. A ti Diosito, aunque poco te entiendo, sé que me quieres.

To the whole family of The Refuge, special thanks for having me always present in your thoughts and in your prayers.

Finalmente, de manera indirecta, a los que poco se mencionan, los que han sido parte de mi proceso de formación, pasando por la escuela hasta la universidad, a mis compañeros en cada una de estas etapas, un pedacito de este logro también es suyo.

# Contents

# List of Figures

# List of Tables

# List of Acronyms

| | |
|---|---|
| **3GPP** | *3rd Generation Partnership Project* |
| **ACF** | *AutoCorrelation Function* |
| **ACS** | *AutoCorrelation Sequence* |
| **ADC** | *Analog-to-Digital Converter* |
| **AWGN** | *Additive White Gaussian Noise* |
| **BER** | *Bit Error Rate* |
| **BPSK** | *Binary Phase Shift Keying* |
| **BS** | *Base Station* |
| **CCDF** | *Complementary Cumulative Distribution Function* |
| **CDF** | *Cumulative Distribution Function* |
| **CIR** | *Channel Impulse Response* |
| **CP** | *Cyclic Prefix* |
| **CSS** | *Chirp Spread Spectrum* |
| **DAC** | *Digital-to-Analog Converter* |
| **DFT** | *Discrete Fourier Transform* |
| **ECR** | *Energy Consumption Ratio* |
| **FDM** | *Frequency Division Multiplexing* |
| **FFT** | *Fast Fourier Transform* |
| **FSK** | *Frequency Shift Keying* |
| **HPA** | *High Power Amplifier* |
| **IDFT** | *Inverse Discrete Fourier Transform* |
| **IFFT** | *Inverse Fast Fourier Transform* |
| **i.i.d.** | *independent and identically distributed* |
| **IoT** | *Internet of Things* |
| **ISI** | *Inter Symbol Interference* |
| **LFDMA** | *Localized Frequency Division Multiple Access* |
| **LPWAN** | *Low Power Wide Area Network* |
| **MCS** | *Modulation and Coding Scheme* |
| **NB-IoT** | *NarrowBand Internet of Things* |
| **OFDM** | *Orthogonal Frequency Division Multiplexing* |
| **PA** | *Power Amplifier* |
| **PAPR** | *Peak-to-Average-Power Ratio* |

| | |
|---|---|
| **pdf** | ***p**robability **d**ensity **f**unction* |
| **PPM** | ***P**ulse **P**osition **M**odulation* |
| **PSD** | ***P**ower **S**pectral **D**ensity* |
| **PSK** | ***P**hase **S**hift **K**eying* |
| **QAM** | ***Q**uadrature **A**amplitude **M**odulation* |
| **QPSK** | ***Q**uadrature **P**hase **S**hift **K**eying* |
| **RF** | ***R**adio **F**requency* |
| **SC-FDMA** | ***S**ingle **C**arrier **F**requency **D**ivision **M**ultiple **A**ccess* |
| **SI** | ***S**ide **I**nformation* |
| **SLM** | ***S**e**L**ected **M**apping* |
| **SNR** | ***S**ignal-to-**N**oise **R**atio* |
| **UE** | ***U**ser **E**quipment* |

# Notations

In this section the definition of the variables and functions used in this thesis is presented in alphabetical order. It is divided by chapter and grouped with their first definition in the order of the chapters.

## Chapter 1

| | |
|---|---|
| $(.)^*$ | Complex conjugate. |
| $\star$ | The convolution $y(t) = x(t) \star h(t)$ defined as $y(t) = \int_{-\infty}^{+\infty} x(\tau)h(t-\tau)d\tau$. |
| $\mathbb{1}_{[a,b]}(f)$ | The indicator function equals to 1 in the interval $[a, b]$ and 0 outside. |
| $A_k$ | Constellation point (PSK, QAM, ...) at the subcarrier index $k$. |
| $B$ | The signals bandwidth. |
| $c(t)$ | Analog linear chirp. From $c(t)$ the up-chirp $c_{\mathrm{u}}(t)$ and the down-chirp $c_{\mathrm{d}}(t)$ are defined. |
| $c[n]$ | Digital linear chirp. From $c[n]$ the up-chirp $c_{\mathrm{u}}[n]$ and the down-chirp $c_{\mathrm{d}}[n]$ are defined. |
| $c_{k_0}(t)$ | $2^{\mathrm{SF}} - CSS$ analog symbol. The equivalent in digital is denoted by $c_{k_0}[n]$. |
| $f_c$ | Carrier frequency. |
| $\Gamma_w[n, m]$ | Autocorrelation sequence of a given discrete random process $w[n]$.<br>By definition, $\Gamma_w[n, m] = \mathbb{E}\{w[n]w^*[n-m]\}$. If $w[n]$ is stationary then its ACS reads $\Gamma_w[m]$. |
| $\Gamma_w(t, \tau)$ | Autocorrelation function of a given random process $w(t)$.<br>By definition, $\Gamma_w(t, \tau) = \mathbb{E}\{w(t)w^*(t-\tau)\}$. If $w(t)$ is stationary then its ACF reads $\Gamma_w(\tau)$. |
| $\gamma_w(f)$ | Power spectral density of a given random process $w(t)$.<br>If $w(t)$ is stationary, then $\gamma_w(f) = \mathrm{TF}\{\Gamma(\tau)\}$ (Wiener-Khintchine theorem). |
| $M$ | Size of the alphabet of the digital modulations (M-PSK, M-QAM, M-FSK, ...). |
| $N$ | Number of subcarriers in OFDM. |
| $t$ | The symbols time duration. |
| $w(t)$ | Additive White Gaussian Noise. |
| $w_{\mathrm{RF}}(t)$ | Pass-band filtered noise. |
| $w_{\mathrm{BB}}(t)$ | Down-converted low-pass filtered noise. |
| $w[n]$ | Discrete white Gaussian noise sampled at Nyquist rate. |
| $x(t)$ | The baseband complex signal occupying a given bandwidth $B$. Also denoted as<br>$x_{\mathrm{BB}}(t) = x_I(t) + ix_Q(t)$, with $x_I(t)$ and $x_Q(t)$ the I and Q components.<br>The equivalent of $x(t)$ in digital is denoted by $x[n]$. |
| $x_{\mathrm{RF}}(t)$ | The real carrier frequency modulated signal to be transmitted through the channel. |
| $y[n]$ | Digital symbols after ADC on the receiver side. |
| $y(t)$ | Received symbols. |
| $y_{\mathrm{I}}(t)$ | In-phase component of the down-converted received symbols. |
| $y_{\mathrm{Q}}(t)$ | Quadrature-phase component of the down-converted received symbols. |
| $Y_k$ | Demodulated OFDM symbols at the subcarrier index $k$. |

# Chapter 2

| | |
|---|---|
| $C$ | Total number of candidate symbols in SLM-OFDM and GreenOFDM. |
| CCDF($\gamma$) | Complementary Cumulative Distribution Function of the PAPR. CCDF($\gamma$) = Pr{PAPR > $\gamma$} . |
| PAPR | Peak-to-Average Power Ratio calculated on the complex digital baseband symbols, *i.e.*, PAPR = $\frac{\max_n\{\lvert x[n]\rvert^2\}}{\mathbb{E}\{\lvert x[n]\rvert^2\}}$ . |
| PAPR$^{I/Q}$ | The PAPR calculated on the I or Q component. |
| PAPR$_{RF}$ | The PAPR calculated on the real carrier frequency modulated signal $x_{\mathrm{RF}}(t)$. |
| PAPR$_u$ | The PAPR of the SLM-OFDM candidate $x_u[n]$. |
| PAPR$_{g_1,g_2}$ | The PAPR of the GreenOFDM candidate $x_{g_1,g_2}[n]$. |
| $\phi_{k,u}$ | The $u^{\text{th}}$ random sequence at index $k$ used to generate the $u^{\text{th}}$ SLM-OFDM candidate. |
| $\phi_{k,g_1}$ | The $g_1^{\text{th}}$ random sequence at index $k$ used to generate the symbols from the first group in GreenOFDM. |
| $\phi_{k,g_2}$ | The $g_2^{\text{th}}$ random sequence at index $k$ used to generate the symbols from the second group in GreenOFDM. |
| $U$ | Total number of computed IFFTs in SLM-OFDM and in GreenOFDM. |
| $x_u[n]$ | The $u^{\text{th}}$ SLM-OFDM symbol candidate. |
| $x_{\tilde{u}}[n]$ | The SLM-OFDM symbol candidate with the lowest PAPR. |
| $x_{g_1}[n]$ | The $g_1^{\text{th}}$ symbol from the first group in GreenOFDM. |
| $x_{g_2}[n]$ | The $g_2^{\text{th}}$ symbol from the second group in GreenOFDM. |
| $x_{g_1,g_2}[n]$ | The GreenOFDM symbol candidate obtained from the sum of the $g_1^{\text{th}}$ and the $g_2^{\text{th}}$ symbol. |
| $x_{\tilde{g_1},g_2}[n]$ | The GreenOFDM symbol candidate with the lowest PAPR. |

# Chapter 3

| | |
|---|---|
| CCDF($\mathcal{M}$) | The probability to compute more than $\mathcal{M}$ operations. |
| CCDF($\varrho$) | The probability to process more than $\varrho$ samples in "Hierarchical sampling". |
| CCDF($v$) | The probability to compute more than $v$ IFFTs in "IFFTs-on-demand". |
| $\gamma_{p_0}$ | A PAPR value related to the probability $p_0 = CCDF(\gamma_{p_0})$. |
| $\gamma_{p_i}$ | A PAPR value related to the probability $p_i = CCDF(\gamma_{p_i})$, with $\gamma_{p_i} > \gamma_{p_0}$. |
| $\mathcal{M}$ | The random variable representing the total number of operations. |
| $\varrho$ | The random variable representing the total amount of processed samples in "Hierarchical sampling". |
| $v$ | The random variable representing the total number of computed IFFTs in "IFFTs-on-demand". |

# Résumé étendu

Ce manuscrit présente les études réalisées lors de mon doctorat en collaboration entre GIPSA-lab et la start-up grenobloise GreenWaves Technologies SAS.

Mon doctorat est centré sur les communications utilisant la modulation « OFDM » (*i.e.*, Orthogonal Frequency Division Multiplexing). Plus précisément, il se focalise sur la diminution de la consommation d'énergie de l'émetteur du signal OFDM.

Les symboles OFDM présentent parfois un ou plusieurs pics d'amplitude quadratique beaucoup plus grande que la valeur quadratique moyenne des symboles. Ces pics imposent une caractéristique linéaire sur une large dynamique, ce qui implique un très mauvais rendement de l'amplificateur analogique, avec une consommation qui peut représenter l'essentiel de l'énergie consommée par le transpondeur.

En écrêtant l'amplitude du signal OFDM à transmettre à une valeur maximale $|A_p|$, les pics d'amplitude d'un symbole OFDM dépassant $|A_p|$ sont éliminés. Ceci introduit une distorsion sur chaque porteuse composant le symbole, distorsion qui ne peut être supprimée ni même parfaitement compensée au niveau du récepteur. En plus d'une possible remontée spectrale hors bande à cause des non linéarités, il en résulte une augmentation du taux d'erreur du système de communication qui peut ne pas être acceptable pour les systèmes de transmission de données dans de nombreux domaines d'applications et évidemment dans le contexte émergeant de « l'Internet des Objets » (*i.e.*, « l'IoT »).

Depuis plus de quatre décennies, de nombreuses techniques de réduction des pics de puissance instantanée par rapport à la puissance moyenne (rapport désigné par le sigle « PAPR » qui signifie « Peak to Average Power Ratio ») ont été proposées et étudiées en détails pour limiter le taux de distorsion à l'émission et ainsi obtenir un système de communication à taux d'erreur suffisamment faible et donc suffisamment fiable pour le domaine d'applications envisagé.

Il est à noter que le PAPR du signal OFDM est une variable aléatoire puisque les pics d'un symbole OFDM sont créés uniquement lorsqu'un nombre suffisamment important de sous-porteuses, chacune étant modulée par des données aléatoires, sont en phase.

Pour représenter ce phénomène probabiliste, une fonction CCDF (Complementary Cumulative Distribution Function) est définie :

$$\mathrm{CCDF}(\gamma) = \Pr\{\mathrm{PAPR} > \gamma\} \approx 1 - (1 - e^{-\gamma})^{2.8N} \tag{1}$$

Par exemple, la Figure 1 ci-dessous représente la CCDF($\gamma$) pour $N = 16, 64, 256$ et 1024, pour une modulation des sous-porteuses en QPSK et avec facteur de suréchantillonnage égal à 4.

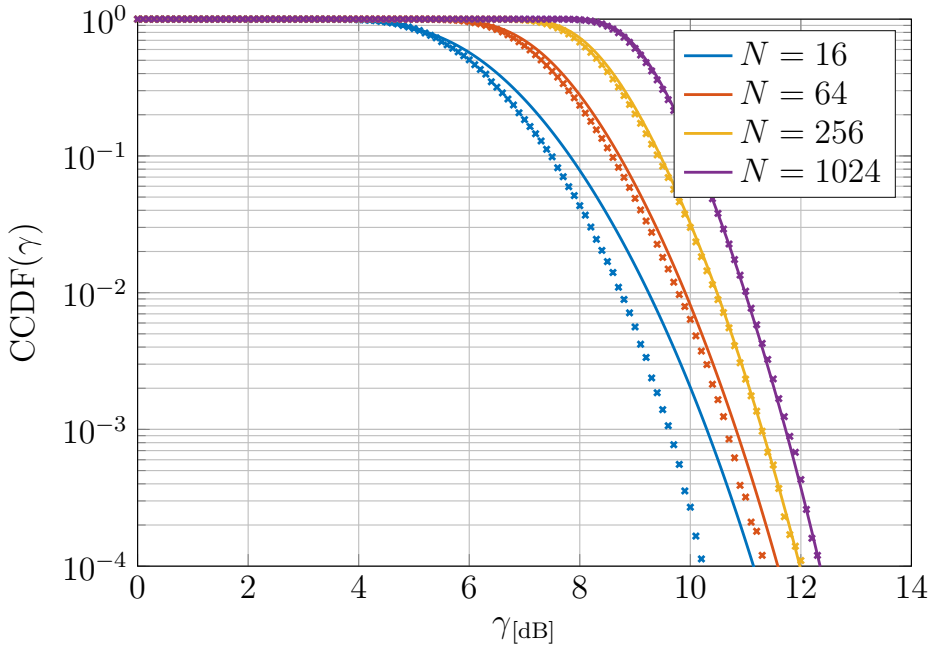



Figure 1 – CCDF($\gamma$) pour $N = 16, 64, 256$ et 1024, modulations QPSK sur chaque sous-porteuse et facteur de suréchantillonnage égal à 4. En ligne pointillée, les résultats de simulation et en ligne continue le résultat donné par l'équation (1).

**Chapitre 1.** Avant de se focaliser sur les apports de mon doctorat, le premier chapitre fournit un aperçu général des fonctions de base des différents blocs qui composent les modulateurs numériques les plus courants actuellement : (i) les modulations linéaires (*e.g.*, PSK, QAM et OFDM) et (ii) les modulations non linéaires (*e.g.*, PPM, FSK, Chirped Spread Spectrum et Chirped-FSK). Il décrit aussi la partie analogique du transmetteur avec une focalisation sur l'amplificateur de puissance radio fréquence. Finalement, le canal à bruit blanc Gaussien additif (en anglais connu sous l'acronyme AWGN, Additive White Gaussian Noise) est présenté.

Comme cela a été mentionné ci-dessus, le caractère probabiliste du PAPR doit être pris en compte pour se rapprocher au plus près de la réalité et aussi parce que certaines expressions analytiques d'un phénomène physiquement probabiliste ne peuvent être obtenues, mais peuvent seulement être approximées en donnant des résultats ne pouvant être considérés comme fiables vis-à-vis de la réalité.

**Chapitre 2.** Le deuxième chapitre aborde la notion de PAPR, c'est-à-dire le rapport de la puissance crête sur la puissance moyenne des symboles OFDM :

(i) la répartition CCDF($\gamma$) du PAPR est analysée et formulée analytiquement et

(ii) estimée par simulations numériques.

Ensuite, par souci de concision, seules trois techniques de réduction de la CCDF($\gamma$) sont

présentées. La technique SLM (Selected Mapping) est la plus détaillée car la méthode que nous proposons, appelée GreenOFDM, s'appuie sur elle et en est une variante plus efficace (*i.e.*, donnant une plus grande réduction de la CCDF($\gamma$) avec les mêmes paramètres). Pour bien comprendre les subtilités du GreenOFDM, il est important de connaître l'essence du principe physique de base de la méthode SLM-OFDM qui est illustré sur la Figure 2.

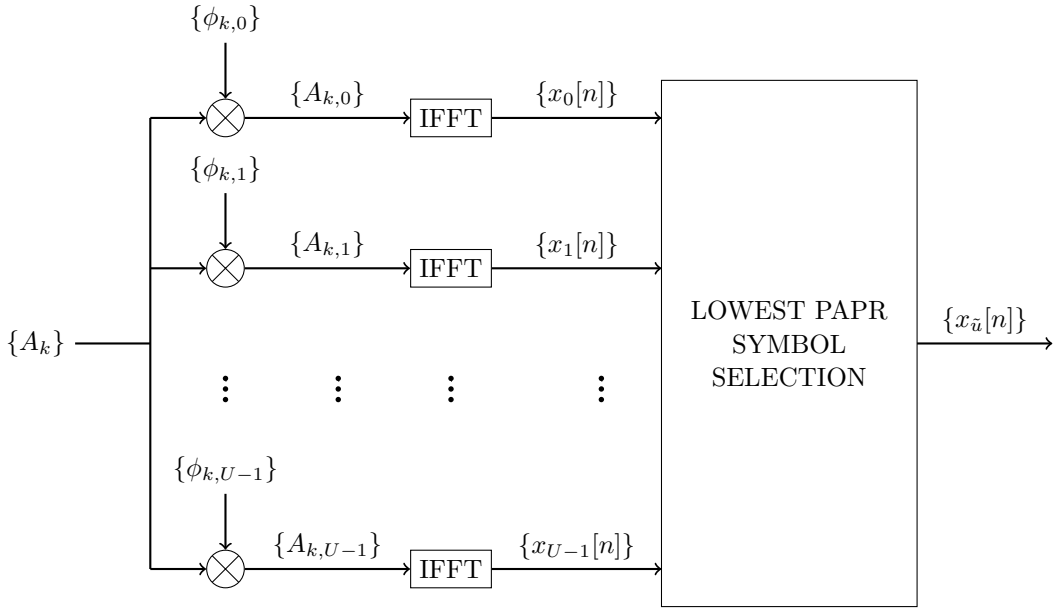



Figure 2 – Illustration schématique du principe de base de la méthode SLM-OFDM.

Le principe physique de base du SLM-OFDM consiste à multiplier les données d'entrée (encodées en PSK, QPSK, 16-QAM ou autre) qui modulent chaque sous-porteuse par un ensemble de séquences aléatoires et à générer, avec une IFFT, un symbole OFDM pour chacune de ces séquences. Le PAPR de chaque symbole OFDM ainsi généré est calculé et on sélectionne le symbole ayant le plus petit PAPR pour la transmission. Pour le SLM-OFDM, CCDF($\gamma$) est bien approchée par :

$$\mathrm{CCDF}(\gamma) \approx (1 - (1 - e^{-\gamma})^{2.8N})^{C} \tag{2}$$

où $N$ est le nombre de sous-porteuses et $C$ est le nombre de symboles OFDM « candidats » devant être générés pour finalement obtenir un symbole OFDM que l'on peut sélectionner pour transmission puisqu'il possède un PAPR compatible avec les caractéristiques de l'amplificateur de puissance. L'essentiel à extraire de l'Eq.(2) est que le CCDF($\gamma$) du SLM-OFDM de base est d'autant réduit que le nombre de candidats $C$ est grand. Cependant, pour augmenter $C$ à des valeurs suffisamment grandes il faut calculer autant de IFFTs, ce qui représente une consommation d'énergie proportionnellement plus importante alors qu'il faudrait idéalement pouvoir générer un grand $C$ avec un minimum de IFFTs.

Le GreenOFDM proposé ici apporte une solution et permet de générer $C_{\mathrm{GreenOFDM}} = U^2/4$ candidats en utilisant seulement $U$ IFFTs alors que le SLM-OFDM original de la Figure 2 nécessite $C_{\mathrm{SLM}} = U^2/4$ IFFTs pour générer autant de candidats. Le rapport du nombre de IFFTs à calculer entre le GreenOFDM et le SLM-OFDM original pour générer le même nombre

de candidats est donc donné par $4/U$, la Figure 3 illustre son principe physique de base.

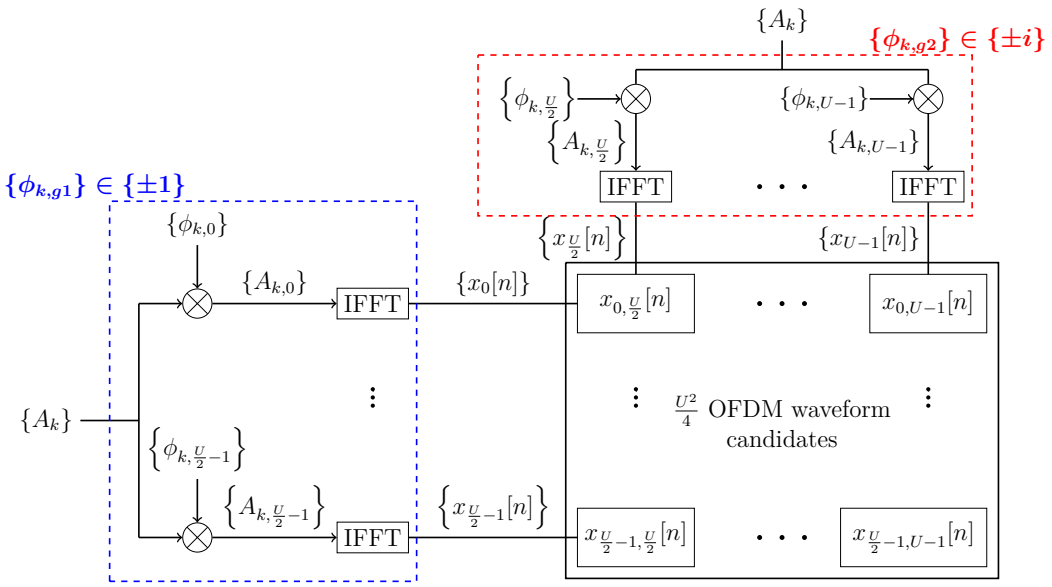
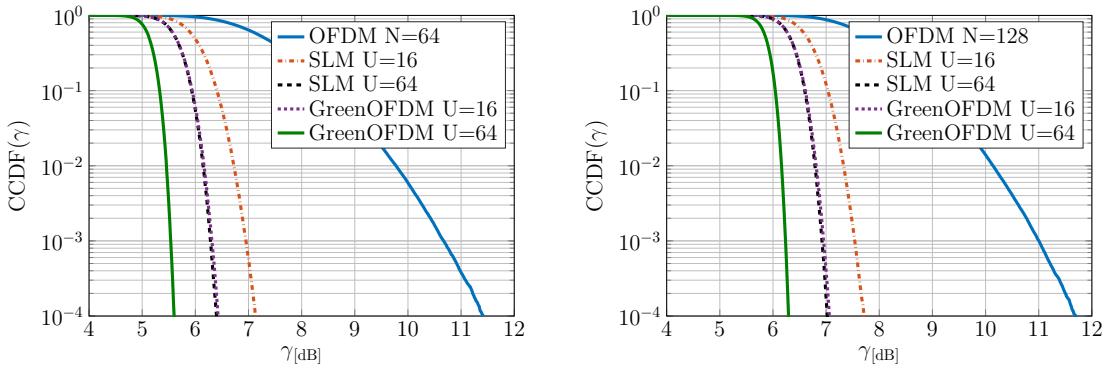



Figure 3 – Illustration schématique du principe de base de la méthode GreenOFDM.

La Figure 3 montre que la génération de symboles GreenOFDM, tout comme le SLM-OFDM d'origine, multiplie les données modulées sur les sous-porteuses par une séquence aléatoire, mais la subtilité essentielle du GreenOFDM est que la séquence aléatoire est scindée en deux parties « orthogonales » ; c'est-à-dire que l'on scinde la séquence aléatoire $\{\pm 1 \pm i\}$ du SLM-OFDM original présenté auparavant dans ce résumé en deux séries indépendantes dont la première est à valeurs dans $\{\pm 1\}$ et la deuxième dans $\{\pm i\}$. La séquence aléatoire $\{\pm 1\}$ est utilisée pour $U/2$ IFFTs et la série $\{\pm i\}$ est utilisée pour $U/2$ IFFTs additionnelles. Les symboles GreenOFDM sont obtenus en additionnant les symboles obtenus avec la série aléatoire $\{\pm 1\}$ et ceux obtenus avec la série aléatoire $\{\pm i\}$. Petit détail mais néanmoins important, l'amplitude des symboles obtenus par cette addition est divisée par le facteur $\sqrt{2}$ pour maintenir la distance euclidienne entre les données modulées et randomisées et donc maintenir la puissance moyenne des symboles GreenOFDM ainsi obtenus égale à celle des symboles OFDM que l'on aurait obtenus sans réduction de PAPR.

Il est maintenant clair que cette nouvelle technique de réduction de PAPR n'utilise au total que $U$ IFFTs pour générer $U^2/4$ symboles GreenOFDM qui ont strictement les mêmes caractéristiques que les symboles OFDM obtenus par la méthode classique d'origine inventée aux Bell-Labs il y a environ 50 ans.

La Figure 4 compare la CCDF du SLM-OFDM et celle du GreenOFDM avec tous les paramètres identiques *i.e.*, $N = 64$ et 128, QPSK/sous-porteuse et $U = 16$ et 64. Ces résultats montrent clairement qu'à nombre de IFFTs fixé, la technique GreenOFDM apporte une amélioration très forte des performances.




(a) (b)

Figure 4 – CCDF($\gamma$) pour l'OFDM, le SLM-OFDM et GreenOFDM : $U = 16$ et $U = 64$, facteur de suréchantillonnage égal à 4 et QPSK/sous-porteuse (a) $N = 64$ et (b) $N = 128$.

**Chapitre 3.** Le chapitre 3 aborde la question du coût de mise en œuvre du GreenOFDM, indépendamment du processeur utilisé ou de l'ASIC dédié et optimisé technologiquement (*e.g.*, Low Power CMOS 55nm ou CMOS 22nm etc.) au GreenOFDM. Des questions se sont posées concernant ; (i) la charge de calcul du transmetteur, et (ii) la consommation énergétique globale comprenant la somme de la partie digitale et de la partie analogique. Dans ce chapitre, nous répondons à ces deux questions essentielles en introduisant deux techniques qui réduisent la charge de calcul.

Premièrement, la charge de calcul est réduite par

(i) le fait que, pour GreenOFDM ou le SLM-OFDM et toutes ses autres variantes la CCDF($\gamma$) diminue rapidement en fonction du nombre de candidats $C$,

(ii) la structure particulière de la modulation GreenOFDM (représentée à la Figure 3) qui implique que, pour chaque IFFT supplémentaire calculée, davantage de candidats sont générés que pour le SLM-OFDM.

Deuxièmement, la sélection du candidat adéquat est effectuée en calculant le PAPR des symboles GreenOFDM sur une partition en trois sous-ensembles des échantillons du signal suréchantillonné d'un facteur 4. Ce mode de calcul du PAPR permet d'éliminer le plus efficacement, et donc le plus rapidement, les symboles GreenOFDM non adéquats du fait qu'ils présentent au moins un pic qui dépasse le $|A_{\max}|$ acceptable par l'amplificateur.

Le chapitre se termine par une comparaison entre la consommation d'énergie globale du GreenOFDM et celle de LoRa dans les applications devant utiliser des LPWAN.

**Chapitre 4.** Finalement, le chapitre 4 est dédié à l'implémentation physique du GreenOFDM, et des deux normes (i) l'IEEE 802.15.4g MR-OFDM et (ii) le 3GPP NB-IoT. Avant l'implémentation physique sur FPGA (Field Programmable Gate Array), la modélisation logicielle de ces trois

systèmes s'est faite en premier lieu sur un émulateur logiciel d'un circuit intégré de GreenWaves Technologies SAS nommé « GAP8 » qui était en cours de fabrication durant la fin de mon doctorat.

Par esprit de concision, la norme 3GPP NB-IoT ne peut pas être soutenue sur le GAP8 parce que la taille maximum des IFFTs de cette norme est de 8K.

Par contre, l'émulateur et ensuite le FPGA du IEEE 802.15.4g MR-OFDM et du GreenOFDM, avec $N = 64$ et les sous-porteuses modulées en 16-QAM, sont tous les deux d'une complexité compatible avec la capacité de calcul du processeur GAP8. Le IEEE 802.15.4g nécessite une fréquence d'horloge du GAP8 de 50 MHz, et le GreenOFDM a besoin d'une fréquence d'horloge de 97.65 MHz.

**Conclusion.** L'ensemble de ce travail apporte plusieurs techniques concrètes et complémentaires permettant une réduction du PAPR des signaux OFDM pour une complexité de calcul donnée.

La technique principale est le GreenOFDM, une variante de SLM qui permet de calculer beaucoup plus de symboles OFDM candidats pour un même nombre de IFFTs ; elle peut être associée à d'autres idées également présentées dans ce document qui permettent de réduire encore plus la charge de calcul.

# Introduction

This manuscript presents the studies carried during my PhD in collaboration between the GIPSA-lab and the Grenoble based start-up GreenWaves Technologies SAS. This research is focused on low power communications using OFDM modulation. The methods presented in this thesis can be implemented in several multi-carrier systems with power consumption constraints. One of the possibilities is in the context of the emerging Internet of Things (**IoT**).

In this introduction we would try to clarify what the IoT is, why it is found in almost everything that is technology-related nowadays, and also introduce what GreenWaves Technologies SAS proposes in order to improve the IoT. Then we introduce an overview of the content of the thesis.

## A Brief History of the IoT

The term Internet of Things or **IoT** was evoked for the first time back in 1999 when Kevin Ashton presented to P&G [1], in order to promote the RFID (Radio-Frequency IDentifier) technology, the fact that computers had only the ability to do what we told them to. However, the computers were just like brains without senses that couldn't collect the data by themselves. This data had to be provided by the human being. The idea exposed back in that time was a type of network in which we can connect several 'things', and those 'things' could be able to measure by themselves the physical world and share what they measured with other 'things' and/or people.

Today, the concept of the IoT has materialized and defines an interconnection of different kind of sensors/devices that are linked through wired and wireless networks. The number of connected devices has exploded within the years and estimations (IDC [2]) say that in 2017 the number of connected devices was around 17 billions. Projections to 2020 say that there will be up to 30-50 billion connected devices. Having such numbers imposes different challenges.

The tendency is that the connected 'things' are more and more battery powered rather than directly connected to power sources (otherwise we would be surrounded by wires everywhere and it would be difficult for the networks to scale). Because of that, the devices must be power efficient. Such high density of devices also impose another challenge related to crowded networks. It would be optimum if the connected 'things' were powered with the sufficient 'intelligence' required to send the data only when it is needed rather than sending unnecessary data and generating bottlenecks in the network. To make this possible, we have to pack the battery powered devices with a processor required to analyze the collected data and hence send it only when needed.

1. Procter and Gamble
2. www.blog-idcfrance.com/les-predictions-idc-2017-linternet-des-objets/

To preserve battery life, the processor must exhibit low power consumption which generally implies computing power limitations. This rises another issue, the security solutions are not the most efficient because of the reduced computing power. Also, the communication techniques must be low power. Several progresses have been done, but improvements are always ongoing. However, it is important to mention that the materialization of the IoT became true thanks to different contributions: i) The idea of the IoT is to interconnect billions of devices, so the adoption of IPv6 became mandatory in order to identify worldwide wise the objects. ii) To have battery powered devices, the arrival of low power processors required to run IoT sensing and applications was a key factor. iii) Also, battery powered devices require low power communication solutions so the today existing techniques were required in order to make possible the IoT.

The research around the IoT encompasses several domains like previously mentioned, *e.g.* the conception and design of power efficient processors, the security issues, the design of smart sensors and devices, the development of innovative applications involving artificial intelligence for data analysis, the low power communication technologies, etc.

As we can see, the IoT covers a large number of research topics with a common aspect: they are technology related. Because of that, it is not a surprise to find the IoT expression appear in almost everything related with technology, going from the most banal marketing announces to research articles and conferences.

Different organizations and companies continue to contribute to the development of the IoT and GreenWaves Technologies SAS is one of these companies.

## GreenWaves Technologies SAS

GreenWaves Technologies SAS is a fabless semiconductor start-up co-founded by 5 former ST experts who reunited with the idea to design an ultra low power processor that enables artificial intelligence algorithms to run on the IoT devices. The product, the first IoT application processor named GAP8, is an 8+1 RISC-V cores SoC (System-on-Chip) with an unprecedented low power consumption at a computing power up to the order of several GOPS (Giga Operations Per Second) at a power consumption at the order of tens of $mW$. Its parallel and fully programmable structure makes it ideal to deal with analysis and classification algorithms in the context of IoT devices. The kind of targeted applications encompasses the classification and processing of several signals like low to medium quality images and videos, sound, vibrations, etc.

The idea with this CIFRE PhD thesis is to show the capability of the GAP8 IoT processor to run different digital communication schemes. The core of these communications being OFDM [3]-based, in order to cope with low energy consumption constraints, a solution to deal with the high Peak-to-Average Power Ratio (PAPR) issue of the OFDM is proposed. This solution, called GreenOFDM, is not limited to be implemented in the IoT only. However, the

3. ***O**rthogonal **F**requency **D**ivision **M**ultiplexing*

IoT being within the business of the company and of this CIFRE thesis, details about some existing IoT solutions is then outlined.

## Manuscript organization

The manuscript is organized as follows:

— In Chapter 1 the general overview of the digital communications chain is presented. The general digital modulations are presented divided in two classes: the linear modulations (*e.g.*, PSK [4], QAM [5]) and the non-linear modulations (*e.g.*, FSK [6], PPM [7]). Then the analog Front-end is presented and the AWGN channel is introduced. Finally the OFDM is presented together with a well know modulation scheme for the IoT, the LoRa modulation.

— In Chapter 2, the focus is on the PAPR issue in OFDM symbols. Here the notion of PAPR is described and analyzed for the baseband OFDM symbols and is compared to the obtained PAPR in RF and for the I and Q components. Then some of the existing solutions for PAPR reduction are presented. Among them, the SLM-OFDM is the most detailed one since the GreenOFDM is a variant of it. Finally the GreenOFDM is introduced and its performances in terms of PAPR reduction are presented.

— In Chapter 3 two essential questions are answered regarding (i) the computational charge of the implementation of GreenOFDM and (ii) the overall energy consumption of the transmitter. Here, two methods for computational charge reduction are presented, called IFFTs-on-demand and hierarchical sampling. Then regarding the energy consumption, the digital implementation of GreenOFDM is compared to the HPA [8] energy consumption. Also the overall energy consumption of the transmitter is compared to other modulation schemes.

— Finally, in Chapter 4, the IEEE 802.15.4g and the 3GPP NB-IoT are overviewed and details about their implementation on GAP8 IoT processor together with the GreenOFDM are drawn.

4. ***P**hase **S**hift **K**eying*
5. ***Q**uadrature **A**amplitude **M**odulation*
6. ***F**requency **S**hift **K**eying*
7. ***P**ulse **P**osition **M**odulation*
8. ***H**igh **P**ower **A**mplifier*

Chapter 1

# IoT Communications

## Contents



In this chapter, the general digital communication chain is presented. In the first section, the two families of digital modulations are presented (i) the linear digital modulations and (ii) the digital non-linear modulations. In this same section, the digital signals that are to be converted to the analog domain pass through the analog Front-end stage. In the end of this section the channel is introduced.

In the second section we present two modulation schemes: (i) the OFDM that is the focus of this thesis and that we can find implemented in the Internet of Things and (ii) the LoRa modulation (a dedicated IoT communication solution). For this last modulation we present a contribution about the description of the modulation.

Finally, conclusions of the chapter appear in the last section.

## 1.1 Digital Communications

In telecommunications, the way in which data is exchanged is defined by protocols. The protocols cover a set of rules defining how the communication happens between the different actors. To ensure successful communications, protocols are divided into different and generally independent layers where each layer treats its specific problems and errors (the design of the protocol becomes simpler in that way).

The most common layer models are the Open Systems Interconnection (OSI) model, composed of 7 abstraction layers and the 5 layered internet model. For both models, the lower layer corresponds to the Physical layer. In this layer, a binary stream is transmitted/received to/from a physical medium (an electrical cable, an optical fibre or a wireless radio frequency link). To do so, in the Physical layer the binary stream coming from the upper layers is converted into physical signals adapted to the transmission medium. On the receiver side the reverse operations are performed to retrieve the binary stream from the received signal.

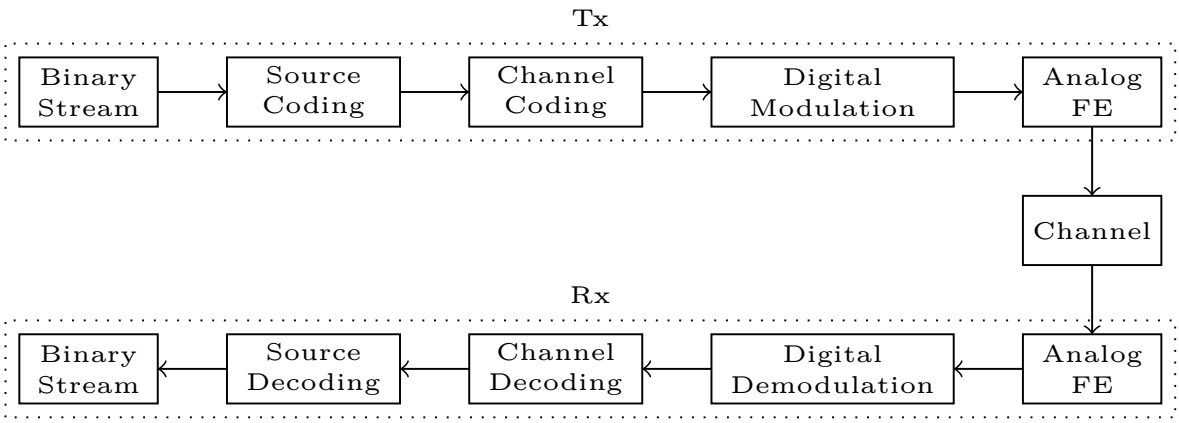



Figure 1.1 – General Physical Layer Block Diagram.

The communication process between two entities can be summarized by the sequential steps depicted in Figure 1.1. These blocks are:

— **Binary Stream**: This is the data to be sent. It might come from upper layers of a protocol stack or directly from the Analog to Digital Conversion of the raw data. This stream is commonly composed of the binary stream itself plus a set of headers and security checks. On the receiver side, once the binary stream is retrieved, it is dispatched to the upper layers in the case of a protocol stack or it is used by the receiver for the purpose it was originally sent.

— **Source Coding/Decoding**: The source coding is related to the compression of the binary stream. Generally data tend to be inherently redundant. The goal of this block is to reduce the amount of binary data to be transmitted by cleverly suppressing the redundancy. On the receiver side the decompression is performed at the source decoding block.

— **Channel Coding/Decoding**: In the channel coding block, controlled redundancy is introduced to the binary data to increase the robustness of the system against errors

produced after channel transmission and reception (interference, fading, noise, ...). The goal here is to detect and at best correct the errors on the receiver side in the channel decoding block.

— **Digital Modulation/Demodulation**: The digital modulator transforms the data to be transmitted into signals that are adapted to the channel medium. On the receiver side, the demodulation not only transforms the received signals into a binary stream but also deals with the perturbations and distortions induced by the channel (the so called channel equalization).

— **Analog Front-end**: Here the digital signals are converted to the analog domain via a Digital-to-Analog-Converter (DAC) and adapted to the transmission medium (carrier frequency up-conversion, filtering, power amplification, ...). On the receiver side the dual operations are performed (amplification, frequency down-conversion, filtering, Analog-to-Digital-Converter (ADC), ...).

— **Channel**: The physical channel represents the medium through which the signals are transmitted. In the channel block gathers all the different interferences and attenuation that affects the transmission. Also the noise is generally included in the channel block despite being a perturbation produced at the receiver side. The transmission channel has a model regarding the medium properties.

The focus of this thesis is on the Digital Modulation aspect of the IoT. This is why in this chapter, before addressing some of the existing IoT Communication schemes, a global overview of the common digital modulation schemes is presented.

### 1.1.1 Digital Modulation schemes

The goal of the digital modulator is, as mentioned above, to map the previously encoded binary data onto signals that have to be transmitted through the channel medium. To do so, the digital modulator has to map groups of $\log_2(M)$ bits onto one among $M$ possible symbols that belong to a predefined alphabet. The symbols having a time support of $T$ seconds, the binary data rate is hence given by $R_b = \frac{\log_2(M)}{T}$ [bits/s].

The digital modulators can be divided into two classes, the linear modulations and the non-linear modulations. In this section, some of the most commonly used digital modulation schemes are presented (for further details, refer to [1]).

#### 1.1.1.1 Linear modulations

In the category of digital linear modulation schemes, the most representative are the Phase Shift Keying (PSK) and the Quadrature Amplitude Modulation (QAM).

**Shift Keying** In the PSK modulation, the alphabet of $M$ symbols is composed of elements that belong to the unitary circle of the complex plane. These elements are given by the expression $A_m = \exp\left\{i\frac{2\pi}{M}(m)\right\}$ with $m = 0, 1, \cdots, M-1$ and $i = \sqrt{-1}$. The elements of

the alphabet are generally represented by their constellation, that is the plot of the imaginary part of $A_m$ ($\Im\{A_m\}$) versus its real part ($\Re\{A_m\}$). The left part of Figure 1.2 depicts the constellations of some M-PSK schemes: (i) the 2-PSK or BPSK for Binary PSK and (ii) the 4-PSK or QPSK for Quadrature PSK [1].

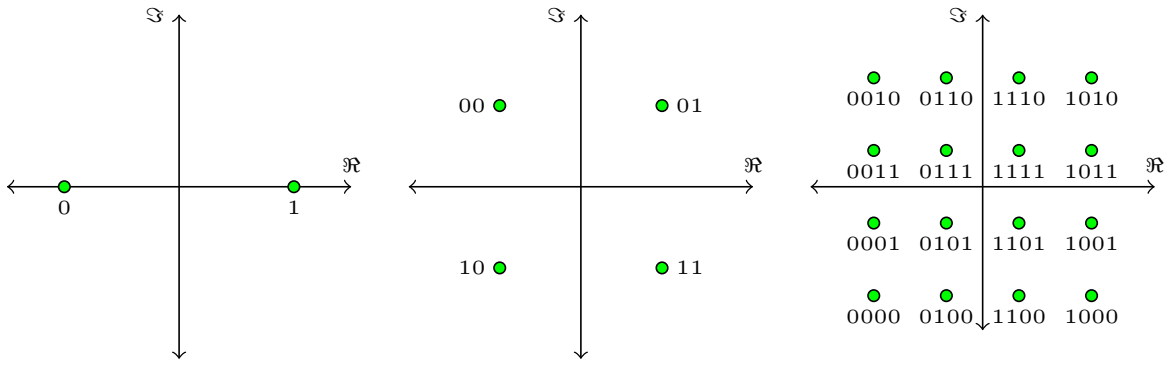



Figure 1.2 – Mapping from binary to complex amplitudes for BPSK, QPSK and $16-$QAM.

Higher values of $M$ lead to increase the data rate. However, on the receiver side, there is a higher probability of error since the minimum distance between the elements of the alphabet decreases as $M$ increases. This is mainly because, in order to maintain a predetermined energy per symbol $\mathrm{E}_s$, the corresponding energy per bit $\mathrm{E}_b = \frac{\mathrm{E}_s}{\log_2(M)}$ decreases as $M$ increases making it error prone.

**Quadrature Amplitude Modulation** In the previously described M-PSK scheme, the constellation points depend only on the phase variations given by $\frac{2\pi}{M}m$. In the M-QAM scheme, the amplitude is also considered to construct the symbols alphabet. The M-QAM symbols are given by $A_m = A_{m_r} + iA_{m_i}$ with $A_{m_r} = \Re\{A_m\}$, $A_{m_i} = \Im\{A_m\}$ and $A_{m_r}, A_{m_i} \in \{\pm 1, \pm 3, \cdots, \pm\sqrt{M}-1\}$ where $M$ is often even. Figure 1.2 depicts the constellations of two M-QAM schemes, the 4-QAM and the 16-QAM. It is noteworthy that here the 4-QAM is also represented by the QPSK modulation meaning that both schemes are equivalent.

For both M-PSK and M-QAM schemes, higher values of $M$ lead to increase the data rate at a cost of a higher error probability on the receiver side. An equilibrium between data rate and error probability is necessary for such linear modulation schemes, specially in energy constrained scenarios.

#### 1.1.1.2 Non-linear modulations

Non-linear modulations are modulations that depend on the past (*e.g.* the phase of a continuous phase modulation). The non-linear modulations are often composed of orthogonal signaling schemes (also known as Orthogonal Signaling modulations). The orthogonal signaling alphabet is composed of $M$ elements $s_m$, with $M = 0, 1, \cdots, M-1$, that satisfy the

1. A $\frac{\pi}{4}$ rotation is represented without affecting the understanding and/or performance of the system

orthogonality condition:

$$< s_m, s_n >= \begin{cases} 0, & m \neq n \\ E_s, & m = n \end{cases} \tag{1.1}$$

where $E_s$ corresponds to the energy per symbol and $< s_m, s_n >$ is the inner product between $s_m$ and $s_n$.

In this family we can find schemes like M-ary Frequency Shift Keying (M-FSK), Pulse Position Modulation (PPM) and the LoRa modulation as well (this scheme is studied in section 1.3). The orthogonality, for the first two examples (*i.e.*, FSK and PPM) is achieved either in the frequency domain, by dividing the available bandwidth into a set of $M$ disjoint frequencies (the FSK case), or in the time domain, by dividing the available time duration into a set of $M$ time slots (the PPM case).

After the digital modulation block we find the analog Front-end which is introduced below.

### 1.1.2 The Analog Front-end

Wireless transmission is the predominant way of communication in the IoT. This requires that, prior to transmission through the wireless antenna, some analog processing is performed. A bridge between the result of the Digital Modulation (denoted from now on as $x(t_n)$) and the analog Front-end is the Digital to Analog Converter (DAC). This bridge produces a continuous time signal denoted as $x(t)$ or $x_{\mathrm{BB}}(t)$ which is a baseband signal occupying a bandwidth of $B$ and with time support $0 \leq t < T$.

Then, this signal is modulated by the carrier frequency $f_c$ through the operation:

$$\begin{aligned} x_{\mathrm{RF}}(t) &= \Re\{x(t)\exp\{i2\pi f_c t\}\} \\ &= \Re\{x(t)\}\cos(2\pi f_c t) - \Im\{x(t)\}\sin(2\pi f_c t) \\ &= x_{\mathrm{I}}(t)\cos(2\pi f_c t) - x_{\mathrm{Q}}(t)\sin(2\pi f_c t) \end{aligned} \tag{1.2}$$

with $\Re\{x(t)\} = x_{\mathrm{I}}(t)$ the In-phase component and $\Im\{x(t)\} = x_{\mathrm{Q}}(t)$ the Quadrature component. This signal is finally amplified through the High Power Amplifier.

On the receiver side, we do have also an Analog Front-end that begins, by duality, with the receiver antenna. Amplification (through the Power Amplifier or PA) and filtering (through the PassBand filter or PB)in the corresponding bandwidth of $x_{\mathrm{RF}}(t)$ is performed prior to carrier frequency down-conversion. At this stage (assuming synchronization has already been performed) the 'bridge' between the analog Front-end and the digital demodulator is obtained via the Analog to Digital Converter (ADC).

Figure 1.3 depicts the block diagram of the mentioned receiver analog Front-end. Assuming a back-to-back scheme, *i.e.* $y(t) = x_{\mathrm{RF}}(t)$, without considering the effects of the channel and

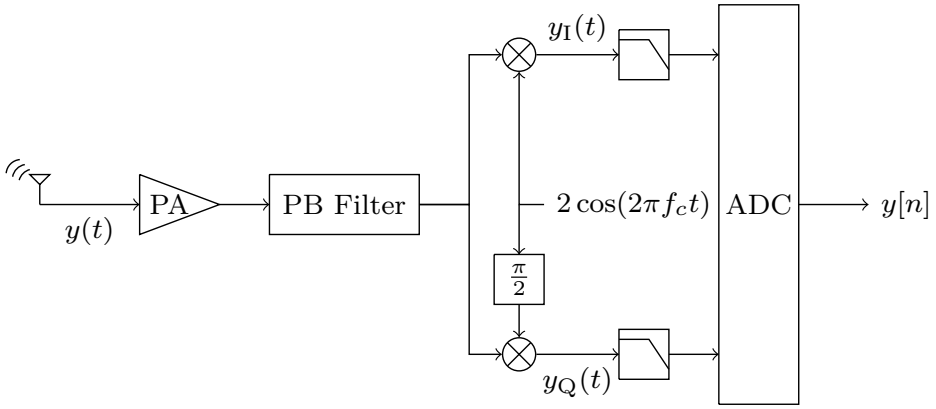



Figure 1.3 – Receiver analog Front-end block diagram.

of the PA (making it a constant equal to 1), we have that for $y_{\mathrm{I}}(t)$:

$$\begin{aligned} y_{\mathrm{I}}(t) &= 2x_{\mathrm{RF}}(t)\cos(2\pi f_c t) \\ &= 2x_{\mathrm{I}}(t)\cos(2\pi f_c t)\cos(2\pi f_c t) - 2x_{\mathrm{Q}}(t)\sin(2\pi f_c t)\cos(2\pi f_c t) \\ &= x_{\mathrm{I}}(t) + (x_{\mathrm{I}}(t)\cos(4\pi f_c t) - x_{\mathrm{Q}}(t)\sin(4\pi f_c t)) \end{aligned} \tag{1.3}$$

and for $y_{\mathrm{Q}}(t)$:

$$\begin{aligned} y_{\mathrm{Q}}(t) &= 2x_{\mathrm{RF}}(t)\cos\left(2\pi f_c t + \frac{\pi}{2}\right) \\ &= -2x_{\mathrm{I}}(t)\cos(2\pi f_c t)\sin(2\pi f_c t) + 2x_{\mathrm{Q}}(t)\sin(2\pi f_c t)\sin(2\pi f_c t) \\ &= x_{\mathrm{Q}}(t) - (x_{\mathrm{I}}(t)\sin(4\pi f_c t) + x_{\mathrm{Q}}(t)\cos(4\pi f_c t)) \end{aligned} \tag{1.4}$$

A low pass filter is required in order to obtain the baseband components only and hence retrieve $x(t)$ prior to Analog-to-Digital Conversion (ADC):

$$\begin{aligned} x_{\mathrm{I}}(t) &= \mathrm{LPF}\left\{y_{\mathrm{I}}(t)\right\} \\ x_{\mathrm{Q}}(t) &= \mathrm{LPF}\left\{y_{\mathrm{Q}}(t)\right\} \end{aligned} \tag{1.5}$$

Another common analog Front-end scheme is depicted in Figure 1.4 where the complex form eases up the calculus. The same way, assuming $y(t) = x_{\mathrm{RF}}(t)$ we have that $y_{\mathrm{BB}}(t)$ reads:

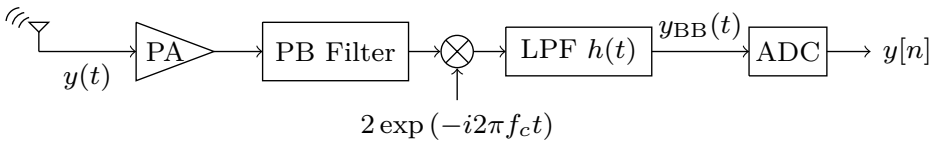



Figure 1.4 – Another scheme for the receiver analog Front-end block diagram.

$$
\begin{aligned}
y_{\mathrm{BB}}(t) &= \mathrm{LPF}\left[2x_{\mathrm{RF}}(t)\exp\left\{-i2\pi f_c t\right\}\right] \\
&= \mathrm{LPF}\left[2\left(x_{\mathrm{I}}(t)\cos(2\pi f_c t) - x_{\mathrm{Q}}(t)\sin(2\pi f_c t)\right)\ \left(\cos(2\pi f_c t) - i\sin(2\pi f_c t)\right)\right] \\
&= \mathrm{LPF}\left[x_{\mathrm{I}}(t)\left(1+\cos(4\pi f_c t)\right) - x_{\mathrm{Q}}(t)\sin(4\pi f_c t) - ix_{\mathrm{I}}(t)\cos(4\pi f_c t) + ix_{\mathrm{Q}}(t)\left(1+\cos(4\pi f_c t)\right)\right] \\
&= x_{\mathrm{I}}(t) + ix_{\mathrm{Q}}(t)
\end{aligned}
\tag{1.6}
$$

Until now we have not talked about a very important element in the communication process: the communication channel.

### 1.1.3 The channel

A communication channel can suffer from several physical phenomena: multipaths, attenuation, shadowing, interferences, fading, noise, etc.

The model of all these phenomena depends on several conditions like the area (for example the channel conditions drastically differ in a rural environment as compared with an urban environment), if there is mobility or not, the occupied spectrum of the transmitted signal (and by analogy, the time support of the transmitted signal), if there exists a Line Of Sight (LOS) between the transmitter and the receiver, just to cite a few.

First, we only consider the channel with additive noise.

#### 1.1.3.1 The AWGN channel

An Additive White Gaussian Noise (AWGN), denoted as $w(t)$, typically models the electronic noise produced on the receiver side. This noise is the result of *the large sum of currents* generated by the random movement of the electrons, this fact allows to approximate the distribution of this noise as a Gaussian (Central Limit Theorem) [2, 3]. Also this noise is considered to be white because, even if its Power Spectral Density (PSD) does not occupy the whole spectrum, it spans over a band larger than the received signal bandwidth.

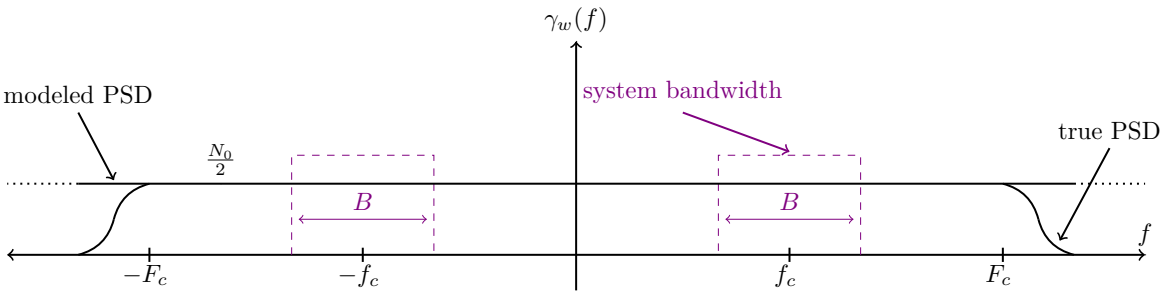



Figure 1.5 – True and modeled Power Spectral Density of white Gaussian noise.

Figure 1.5 depicts the true and the modeled PSD of such a "white" noise. Here it can be seen that the information is carried in the portions of the spectrum $|f \pm f_c| \leq \frac{B}{2}$ around $\pm f_c$. The modeled and the true PSD coincide until a certain frequency $F_c$ where the true PSD begins to decrease, thus both PSD are equal in the signal bandwidth $|f \pm f_c| \leq \frac{B}{2}$. That is why it is possible to use the modeled PSD instead of the true PSD. The modeled PSD of $w(t)$ is given by:

$$\gamma_w(f) = \frac{N_0}{2} \quad -\infty < f < +\infty \tag{1.7}$$

And hence its autocorrelation function (ACF) is given by [2]:

$$\Gamma_w(\tau) = \frac{N_0}{2}\delta(\tau) \tag{1.8}$$

where $\delta(\tau)$ denotes a "Dirac impulse" at $\tau = 0$.

In the band of the system ($[-f_c - \frac{B}{2}, -f_c + \frac{B}{2}] \cup [f_c - \frac{B}{2}, f_c + \frac{B}{2}]$), the resulting noise, called $w_{\mathrm{RF}}(t)$, has other properties. Its PSD is now given by:

$$\gamma_{w_{\mathrm{RF}}}(f) = \frac{N_0}{2}\left(\mathbb{1}_{[-f_c-\frac{B}{2},-f_c+\frac{B}{2}]}(f) + \mathbb{1}_{[f_c-\frac{B}{2},f_c+\frac{B}{2}]}(f)\right) \tag{1.9}$$

where $\mathbb{1}_{[a,b]}(x)$ is the indicator function:

$$\mathbb{1}_{[a,b]}(x) = \begin{cases} 1 \text{ for } a \leq x \leq b \\ 0 \text{ otherwise,} \end{cases}$$

and its ACF is given by:

$$\begin{aligned} \Gamma_{w_{\mathrm{RF}}}(\tau) &= \int_{-f_c-\frac{B}{2}}^{-f_c+\frac{B}{2}} \frac{N_0}{2}\exp\{i2\pi f\tau\}\,df + \int_{f_c-\frac{B}{2}}^{f_c+\frac{B}{2}} \frac{N_0}{2}\exp\{i2\pi f\tau\}\,df \\ &= N_0 B\mathrm{sinc}(\pi B\tau)\cos(2\pi f_c\tau), \quad \text{with } \mathrm{sinc}(x) = \frac{\sin(x)}{x} \end{aligned} \tag{1.10}$$

This result shows that the noise $w_{\mathrm{RF}}(t)$ at the output of the bandpass filter is correlated.

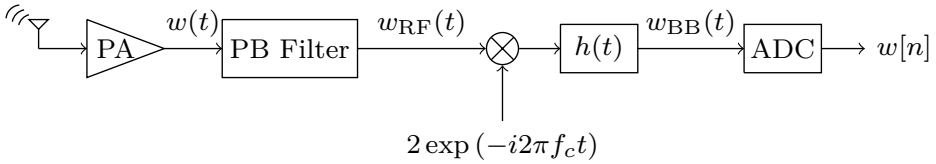



Figure 1.6 – Receiver noise analog Front-end block diagram.

After that, the frequency down-conversion takes place. From Figure 1.6 we have that the component $w_{\mathrm{BB}}(t)$ is given by:

$$w_{\mathrm{BB}}(t) = 2w_{\mathrm{RF}}(t)\exp\{-i2\pi f_c t\} \star h(t) \tag{1.11}$$

2. Inverse Fourier transform of the PSD (Wiener-Khintchine theorem)

where $\star$ represents the convolution; $h(t)$ is the impulse response of the low pass filter in the band of the system $B$ whose transfer function $H(f)$ is given by:

$$H(f) = \mathbb{1}_{[-\frac{B}{2},\frac{B}{2}]}(f) \tag{1.12}$$

The ACF at the output of the filter is by definition given by:

$$\begin{aligned}
\Gamma_{w_{\mathrm{BB}}}(t,\tau) &= \mathbb{E}\left\{w_{\mathrm{BB}}(t)w^*_{\mathrm{BB}}(t-\tau)\right\} \\
\Gamma_{w_{\mathrm{BB}}}(t,\tau) &= \mathbb{E}\left\{\left(2w_{\mathrm{RF}}(t)\exp\left\{-i2\pi f_c t\right\} \star h(t)\right)\left(2w^*_{\mathrm{RF}}(t-\tau)\exp\left\{i2\pi f_c(t-\tau)\right\} \star h^*(t-\tau)\right)\right\} \\
\Gamma_{w_{\mathrm{BB}}}(t,\tau) &= 4\int\int_{-\infty}^{+\infty} h(\alpha)\exp\left\{-i2\pi f_c(t-\alpha)\right\}\mathbb{E}\left\{w_{\mathrm{RF}}(t-\alpha)w^*_{\mathrm{RF}}(t-\tau-\beta)\right\} \\
&\qquad\qquad h^*(\beta)\exp\left\{i2\pi f_c(t-\tau-\beta)\right\}d\alpha d\beta \\
\Gamma_{w_{\mathrm{BB}}}(\tau) &= 4\int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} h(\alpha)h^*(\beta)\Gamma_{w_{\mathrm{RF}}}(\tau-\alpha+\beta)\exp\left\{-i2\pi f_c(\tau-\alpha+\beta)\right\}d\alpha d\beta.
\end{aligned} \tag{1.13}$$

Since the noise at the output of the filter is stationary, its PSD can be obtained through its ACF Fourier Transform. With the substitution $u = \tau - \alpha + \beta$ we have:

$$\begin{aligned}
\gamma_{w_{\mathrm{BB}}}(f) &= \int_{-\infty}^{+\infty}\Gamma_{w_{\mathrm{BB}}}(\tau)\exp\left\{-i2\pi f\tau\right\}d\tau \\
&= 4\int\int\int h(\alpha)h^*(\beta)\Gamma_{w_{\mathrm{RF}}}(u)\exp\left\{-i2\pi f_c(u)\right\}\exp\left\{-i2\pi f(u+\alpha-\beta)\right\}d\alpha d\beta du \\
&= 4\underbrace{\int h(\alpha)\exp\left\{-i2\pi f\alpha\right\}d\alpha}_{H(f)}\underbrace{\int h^*(\beta)\left\{i2\pi f\beta\right\}d\beta}_{H^*(f)}\underbrace{\int\Gamma_{w_{\mathrm{RF}}}(u)\exp\left\{-i2\pi(f+f_c)u\right\}du}_{\gamma_{w_{\mathrm{RF}}}(f+f_c)} \\
&= 4|H(f)|^2\gamma_{w_{\mathrm{RF}}}(f+f_c) \\
&= 4\frac{N_0}{2}\mathbb{1}_{[-\frac{B}{2},\frac{B}{2}]}(f)\left(\mathbb{1}_{[-2f_c-\frac{B}{2},-2f_c+\frac{B}{2}]}(f)+\mathbb{1}_{[-\frac{B}{2},\frac{B}{2}]}(f)\right) \\
&= 2N_0\mathbb{1}_{[-\frac{B}{2},\frac{B}{2}]}(f)
\end{aligned} \tag{1.14}$$

Hence the ACF obtained from the inverse Fourier Transform of $\gamma_{w_{\mathrm{BB}}}(f)$ reads:

$$\Gamma_{w_{\mathrm{BB}}}(\tau) = 2N_0B\mathrm{sinc}(\pi B\tau) \tag{1.15}$$

It is observed that the obtained noise baseband representation is correlated whereas it is white in the system bandwidth.

Finally, the receiver converts the analog signals to the digital domain before the received signals are digitally demodulated. Say the receiver samples at the rate $LB$ samples/sec where the integer $L$ is equal or greater than 1 ($L = 1$ corresponding to the Nyquist rate). The digital version of the noise $w_{\mathrm{BB}}(t)$ sampled at the instants $t_n = \frac{n}{LB}$ is denoted as $w[n] = w_{\mathrm{BB}}(t_n)$. Its autocorrelation sequence (ACS) $\Gamma_w[m]$ is obtained by sampling the ACF $\Gamma_{w_{\mathrm{BB}}}(\tau)$ at the delays $\tau_m = \frac{m}{LB}$. The ACS of the sampled noise $w[n]$ is thus given by:

$$\begin{aligned}\Gamma_w[m] &= 2N_0B\text{sinc}(\pi B\tau_m) \\ &= 2N_0B\text{sinc}(\pi\frac{m}{L})\end{aligned} \tag{1.16}$$

This result says that the digital noise remains correlated for all oversampling required in the digital demodulation ($L > 1$). However, in the particular case of the Nyquist rate ($L = 1$), the noise ACS becomes:

$$\Gamma_w[m] = \begin{cases} 2N_0B & \text{for } m = 0 \\ 0 & \text{otherwise since } \text{sinc}(\pi m) = 0,\ m \in \mathbb{Z}^* \end{cases}$$

We conclude that the filtered noise sampled exactly at the Nyquist rate becomes uncorrelated.

## 1.2 OFDM for the IoT

The first type of IoT that emerged were the sensors networks. These networks were, and continue to be, a combination of cellular and/or local area solutions.

The main constraint of this kind of networks is that the devices must have continuous access to a cabled power source, making it difficult for them to scale. Today's IoT solutions need low power communications and use long battery-life devices. Motivated by the low power communication idea, several techniques have been proposed. Among today's IoT physical communication solutions one finds: LoRa, IEEE 802.15.4g, NB-IoT, $\cdots$.

The above mentioned OFDM-based IoT PHYs (IEEE 802.15.4g with MR-OFDM option — where MR stands for Multi-Rate — and the 3GPP NB-IoT) will be detailed later, in chapter 4 in which their implementation on GreenWaves Technologies SAS GAP8 processor will be carried out.

In this chapter we focus on the OFDM modulation scheme used for example in 3GPP NB-IoT and IEEE 802.15.4g MR-OFDM. In the sequel, we introduce the OFDM modulation and we provide a contribution to the analysis of the LoRa modulation (one of the most spread communication scheme in the so-called Low Power Wide Area Networks).

In order to explain how an OFDM modulator works, we first explain how a single symbol is generated, then we explain how to produce a sequence of OFDM symbols.

### 1.2.1 Generation of a single OFDM symbol

The Orthogonal Frequency Division Multiplexing (OFDM [4]) is a linear modulation in which the signal is multi-carrier rather than single carrier. The idea of OFDM is to separate the available bandwidth into a set of orthogonal sub-channels, each channel carries a fraction of the whole signal. This allows to have either:

(i) a multi-user scheme [3] where each user has a specific channel allocated only to him,

(ii) a longer time support signal obtained by the division of a high rate signal into many lower rate signals in order to increase the robustness of the communication scheme against Inter Symbol Interference (ISI).

The idea in OFDM is to allow the sub-channels to overlap in the frequency domain while maintaining their orthogonality. Therefore, the available bandwidth $B$ is split into $N$ overlapping sub-channels — or subcarriers — with main lobes occupying a frequency band of:

$$\Delta f = \frac{B}{N} \tag{1.17}$$

The OFDM signal $x_{\mathrm{BB}}(t)$ is the result of the sum of $N$ orthogonal time-limited sub-signals using subcarriers $\exp\{i2\pi f_k t\}$ over a finite time duration $T$, with $f_k = k\Delta f$ and modulated by the data $A_k$ (these $A_k$ symbols belong to an alphabet like M-PSK or M-QAM):

$$x_{\mathrm{BB}}(t) = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} A_k \exp\{i2\pi f_k t\}, \quad 0 \leq t < T \tag{1.18}$$

The analog implementation of a bank of overlapping subcarriers has a high cost. However, within the years it was found that OFDM complexity could be drastically reduced by means of digital implementation. The OFDM symbol in Eq. (1.18) sampled at the Nyquist rate $t_n = \frac{n}{B}$, $n = 0, \cdots, N-1$ is given by:

$$\begin{aligned} x[n] &= \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} A_k \exp\left\{i2\pi f_k \frac{n}{B}\right\} \\ &= \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} A_k \exp\left\{i2\pi \left(k\Delta f\right) \frac{n}{B}\right\} \text{ with } \Delta f = \frac{B}{N} \end{aligned}$$

Therefore, the OFDM symbol is obtained by Inverse Discrete Fourier Transform (IDFT) of the sequence $\{A_k\}$:

$$x[n] = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} A_k \exp\left\{i2\pi \frac{kn}{N}\right\}, \qquad \begin{cases} t_n &= \frac{n}{B} \\ n &= 0, 1, \cdots, N-1 \end{cases} \tag{1.19}$$

Figure 1.7 depicts the digital block diagram of the time-domain OFDM symbol generation. The data $A_k$, is the result of the binary data mapping (PSK or QAM) and the "S/P" block corresponds to a serial-to-parallel conversion obtained by regrouping the data in complex sets $\{A_k\}$ of $N$ elements that input the $N - IDFT$ (or IFFT for lower digital implementation complexity).

3. OFDMA

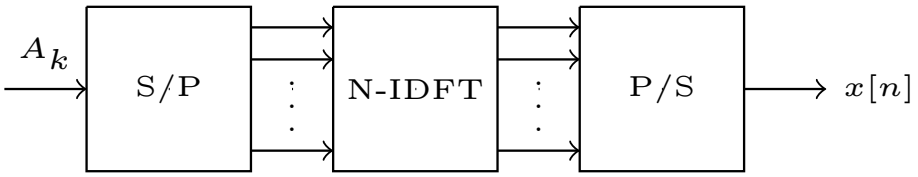



Figure 1.7 – OFDM digital modulator block diagram.

### 1.2.2 Generation of the OFDM signal

The OFDM signal to be transmitted is the result of the concatenation of successive OFDM symbols. In this thesis only the AWGN channel is considered since its aim is to tackle the PAPR[4] reduction problem. We focus on the principle of an OFDM transmission:

**The Channel** In practical systems, the channel is convolutive. A common representation of the channel is through its Channel Impulse Response (CIR). The digital baseband equivalent is given by $h[n]$ for $n = 0, \cdots, N_L - 1$.

**The Guard Interval** In order to prevent the ISI due to the channel response, a guard interval is to be inserted between the successive symbols at a cost of overall data rate degradation. The length $N_{\text{CP}}$ of this guard interval is chosen to be greater than the impulse channel duration $N_L$.

Among possible choices, appending an extension to each generated OFDM symbol $x[n]$ composed of its last $N_{\text{CP}}$ samples (with $N_{\text{CP}} \geq N_L$) is the most common solution. This is the so-called Cyclic Prefix (CP).

After the CP is appended, the successive symbols are concatenated to obtain the OFDM signal. The final steps of the transmitter are depicted in Figure 1.8.

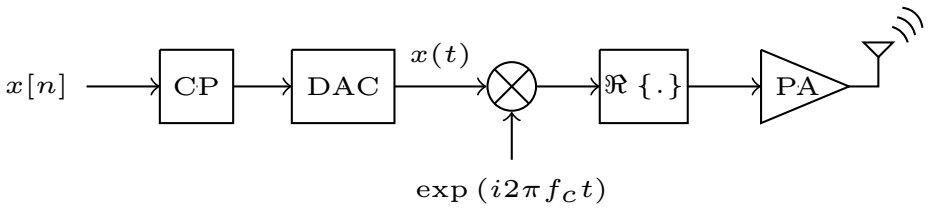



Figure 1.8 – OFDM transmitter Analog Front-end.

**Why a CP?** To explain why the CP is the most famous guard interval, we mention first how to recover the data at the receiver side. After all the analog stages and ADC, each received digital symbol, say $y[n]$, is a version of the transmitted symbol $x[n]$ weighted by the

4. This issue is explained on Chapter 2.

convolutive channel and noise. As previously presented, the OFDM modulator is performed by an IDFT of the data $\{A_k\}$. By duality, the $\{A_k\}$ are recovered by the OFDM demodulator, a $DFT$[5].

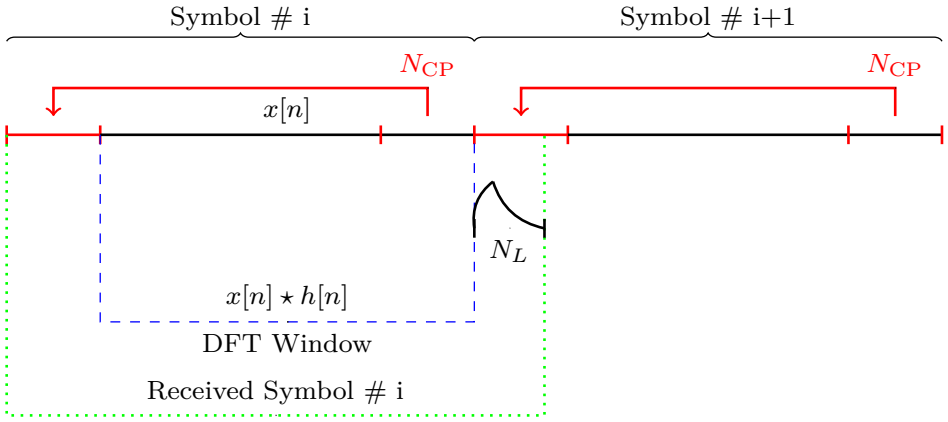



Figure 1.9 – CP example on successive OFDM symbols.

However, due to the convolutive channel, complex equalizers are required to overcome the channel effect. This is where the CP utility is founded. Figure 1.9 depicts an example of the CP for successive OFDM symbols. First, we can observe a trivial CIR with time support $N_L$. It becomes obvious that ISI between the successive OFDM symbols is avoided by choosing $N_{\text{CP}}$ to be greater or equal to $N_L$.

Since the extension of the OFDM symbol after appending the CP makes the new OFDM symbol to be cyclic, hence , neglecting the noise in a first approach, the transmission through the convolutive channel becomes a circular convolution (depicted by $\star$ in the figure):

$$y[n] = x[n] \star h[n]$$

Its Discrete Fourier Transform (DFT) is given by:

$$\begin{aligned} Y_k &= \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} y[n] \exp\left\{-i2\pi \frac{kn}{N}\right\} \\ Y_k &= \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} \sum_{m=0}^{N_L-1} h[m]x[n-m] \exp\left\{-i2\pi \frac{kn}{N}\right\} \\ Y_k &= \frac{1}{\sqrt{N}} \sum_{m=0}^{N_L-1} h[m] \underbrace{\sum_{n=0}^{N-1} x[n-m] \exp\left\{-i2\pi \frac{kn}{N}\right\}}_{\text{A}} \end{aligned} \tag{1.20}$$

With the substitution $n' = n - m$, the expression denoted as A in the previous equation

5. to simplify implementation we use a "FFT" (Fast Fourier Transform).

becomes:

$$\begin{aligned} \mathrm{A} &= \sum_{n'=-m}^{N-1-m} x[n'] \exp\left\{-i2\pi \frac{k(n'+m)}{N}\right\} \\ &= \underbrace{\left(\sum_{n'=-m}^{N-1-m} x[n'] \exp\left\{-i2\pi \frac{kn'}{N}\right\}\right)}_{\mathrm{B}} \exp\left\{-i2\pi \frac{km}{N}\right\} \end{aligned} \tag{1.21}$$

Since the CP made the symbol $x[n]$ cyclic with period $N$, we have for $n' = -m, \cdots, -1$: $x[n'] = x[n'+N]$ and $\exp\left\{-i2\pi \frac{kn'}{N}\right\} = \exp\left\{-i2\pi \frac{k(n'+N)}{N}\right\}$. The expression denoted as B in the previous equation reads:

$$\begin{aligned} \mathrm{B} &= \sum_{n'=-m}^{-1} x[n'] \exp\left\{-i2\pi \frac{kn'}{N}\right\} + \sum_{n'=0}^{N-1-m} x[n'] \exp\left\{-i2\pi \frac{kn'}{N}\right\} \\ &= \sum_{n'=N-m}^{N-1} x[n'] \exp\left\{-i2\pi \frac{kn'}{N}\right\} + \sum_{n'=0}^{N-1-m} x[n'] \exp\left\{-i2\pi \frac{kn'}{N}\right\} \\ &= \sum_{n'=0}^{N-1} x[n'] \exp\left\{-i2\pi \frac{kn'}{N}\right\} \\ &= \sqrt{N} X[k] \end{aligned} \tag{1.22}$$

and this corresponds to the DFT of $x[n]$: $X_k$. Introducing Eq.(1.21) and Eq.(1.22) into Eq.(1.20) reads:

$$\begin{aligned} Y_k &= \frac{1}{\sqrt{N}} \sum_{m=0}^{N_L-1} h[m] \sum_{n=0}^{N-1} x[n-m] \exp\left\{-i2\pi \frac{kn}{N}\right\} \\ Y_k &= X_k \sum_{m=0}^{N_L-1} h[m] \exp\left\{-i2\pi \frac{km}{N}\right\} \\ Y_k &= \sqrt{N} X_k H_k \end{aligned} \tag{1.23}$$

with $H[k]$ the DFT of $h[n]$.

This means that, for each received symbol, after the OFDM demodulator (performed in the correct DFT window) we obtain the data $A_k$ weighted by a single coefficient $H_k$ so that a low complexity channel equalization can be operated. Neglecting the additive noise, this equalization consists in a point-wise multiplication in the frequency domain to suppress the effects of $H[k]$ [6]: the so-called Zero-Forcing equalization. And now, the additional benefits of the CP are evident: it not only prevents the ISI but also eases the channel equalization.

Figure 1.10 depicts the digital demodulation steps required on the receiver side in order to recover the data $\{A_k\}$.

6. Obtained through the so-called channel estimation (*e.g.* pilot-aided channel estimation).

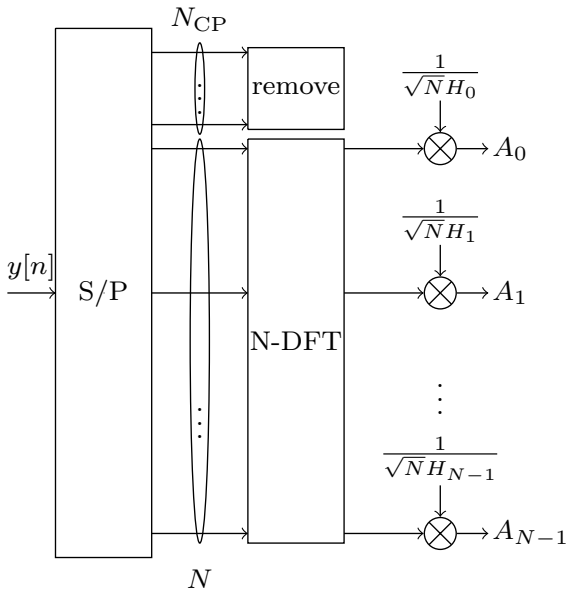



Figure 1.10 – OFDM digital demodulator block diagram.

## 1.3 LoRa in the IoT

The Semtech's LoRa$^{\mathrm{TM}}$(Long Range) modulation technology [5] is a Chirp Spread Spectrum variant technique that supports long-range communications. There is no official LoRa modulation papers explaining its theoretical principles.

The essential basics of the LoRa modulation is slightly explained in [6, 7]. This modulation is based on chirp signals; we first introduce the Linear Chirp technique.

### 1.3.1 Linear Chirp

The chirps are classically used in radar systems. They were fist proposed for communication systems back in 1962 [8]. A linear chirp consists of a linear frequency sweep over a time period $T$. The analog linear chirp is given by:

$$c(t) = \exp\left\{i\phi\left(t\right)\right\}, \quad 0 \leq t < T \tag{1.24}$$

where $i^2 = -1$ and with instantaneous frequency:

$$f(t) = \frac{1}{2\pi}\frac{d\phi(t)}{dt} = \mu\frac{B}{T}\left(t - \frac{T}{2}\right), \quad 0 \leq t < T \tag{1.25}$$

and hence Eq. (1.24) reads:

$$c(t) = \exp\left\{i\pi\mu\frac{B}{T}\left(t^2 - Tt\right)\right\}, \quad 0 \leq t < T \tag{1.26}$$

with $B$ the occupied signal bandwidth and $\mu \in \pm 1$ indicates the slope of the chirp. Usually when $\mu = 1$ the generated chirp is referred as up-chirp ($c_{\mathrm{u}}(t) = c_{\mu=1}(t)$) and when $\mu = -1$ the generated chirp is referred as down-chirp ($c_{\mathrm{d}}(t) = c_{\mu=-1}(t)$):

$$\begin{aligned} c_{\mathrm{u}}(t) &= \exp\left\{ i\pi \frac{B}{T}\left(t^2 - Tt\right)\right\} \\ c_{\mathrm{d}}(t) &= \exp\left\{ -i\pi \frac{B}{T}\left(t^2 - Tt\right)\right\} \end{aligned} \tag{1.27}$$

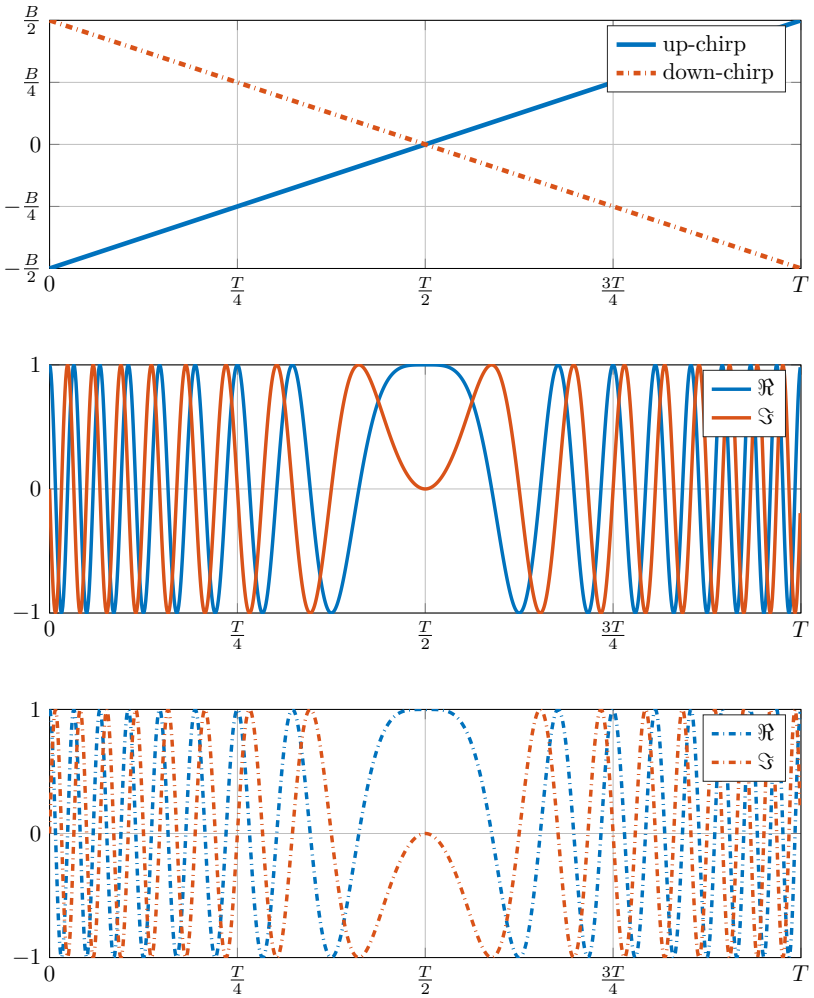



Figure 1.11 – Instantaneous frequency sweep (first figure) and time domain up-chirp and down-chirp (second and third figures respectively) during one chirp duration.

Figure 1.11 depicts the frequency sweep of both up-chirp and down-chirp of Eq.(1.25) as well as the time-domain waveform of Eq.(1.26).

The digital version of the chirp in Eq.(1.26), sampled at the time instants $t_n = \frac{n}{B}$ is given by:

$$c[n] = \exp\left\{ i\pi\mu \left( \frac{n^2}{BT} - n \right) \right\} \tag{1.28}$$

where $n = 0, 1, \cdots, N-1$. Hence, the equivalent up-chirp and down-chirp are given by:

$$\begin{aligned} c_{\mathrm{u}}[n] &= \exp\left\{i\pi\left(\frac{n^2}{BT} - n\right)\right\} \\ c_{\mathrm{d}}[n] &= \exp\left\{-i\pi\left(\frac{n^2}{BT} - n\right)\right\} \end{aligned} \tag{1.29}$$

### 1.3.2 $2^{\mathbf{SF}}-$Chirp Spread Spectrum

The LoRa modulation developed by Semtech [5] is a chirp based technique also denoted as Chirp Spread Spectrum (CSS or $2^{\mathrm{SF}} - CSS$). Here, a set of $N = 2^{\mathrm{SF}} = BT$ chirps is constructed where each chirp encodes $\mathrm{SF} \in \{7, \cdots, 12\}$ bits (SF stands for "Spreading Factor" according to [5]).

#### 1.3.2.1 Analog CSS chirps

In the analog case, the frequency trajectory of each chirp is obtained through a cyclic shift of the up-chirp frequency sweep in Eq.(1.25) ($\mu = 1$). The $k_0-$shifted frequency trajectory ($0 \leq k_0 \leq N-1$) is given by:

$$f_{k_0}(t) = \begin{cases} \left(\frac{B}{T}\left(t - \frac{k_0}{B} + \frac{T}{2}\right)\right), & 0 \leq t < \frac{k_0}{B} \\ \left(\frac{B}{T}\left(t - \frac{k_0}{B} - \frac{T}{2}\right)\right), & \frac{k_0}{B} \leq t \leq T \end{cases} \tag{1.30}$$

The resulting chirp is given by:

$$\begin{aligned} c_{k_0}(t) &= \exp\left\{i\phi_{k_0}(t)\right\} \\ &= \begin{cases} \exp\left\{i2\pi\frac{B}{T}\left(\frac{t^2}{2} - \frac{k_0}{B}t + \frac{T}{2}t\right)\right\}, & 0 \leq t < \frac{k_0}{B} \\ \exp\left\{i2\pi\frac{B}{T}\left(\frac{t^2}{2} - \frac{k_0}{B}t - \frac{T}{2}t\right)\right\}, & \frac{k_0}{B} \leq t \leq T \end{cases} \end{aligned} \tag{1.31}$$

The instantaneous frequency as well as the time domain representation of the obtained chirp are depicted in Figure 1.12 for an arbitrary value of $k_0$. Here we can observe a discontinuity in the time instant $\frac{k_0}{B}$.

We are now interested in the orthogonality between the chirps, *i.e.* the orthogonality of the set $\{c_{k_0}(t)\}_{0 \leq k_0 \leq N-1}$.

**Orthogonality of the analog chirps** $c_{k_0}(t)$ The orthogonality between the chirps from Eq.(1.31) is verified if the inner product $\langle c_{k_1}, c_{k_2} \rangle$ of two different chirps $c_{k_1}(t)$ and $c_{k_2}(t)$ is zero. We have:

$$c_{k_1}(t) = \begin{cases} \exp\left\{i2\pi\frac{B}{T}\left(\frac{t^2}{2} - \frac{k_1}{B}t + \frac{T}{2}t\right)\right\}, & 0 \leq t < \frac{k_1}{B} \\ \exp\left\{i2\pi\frac{B}{T}\left(\frac{t^2}{2} - \frac{k_1}{B}t - \frac{T}{2}t\right)\right\}, & \frac{k_1}{B} \leq t \leq T \end{cases}$$

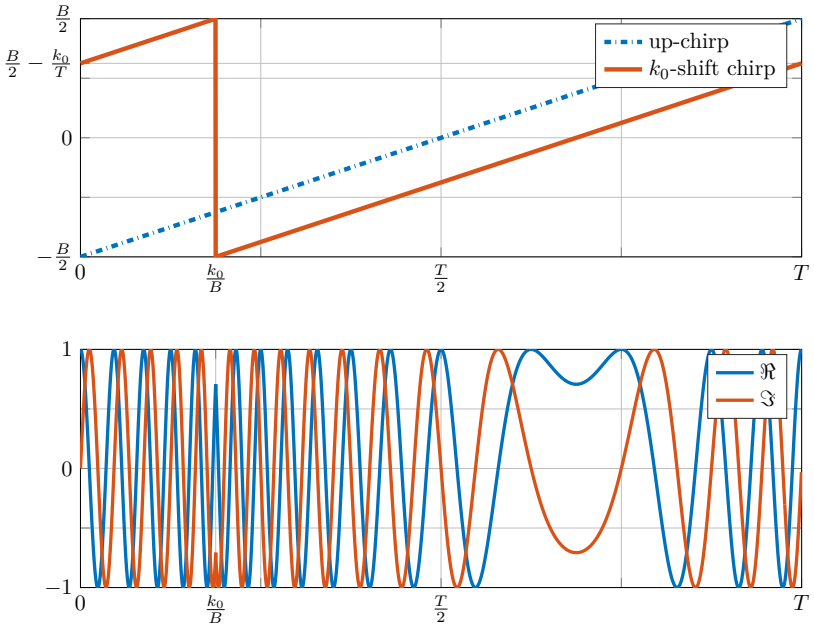



Figure 1.12 – $k_0$-shifted frequency trajectory and corresponding time domain chirp.

$$c_{k_2}(t) = \begin{cases} \exp\left\{i2\pi\frac{B}{T}\left(\frac{t^2}{2} - \frac{k_2}{B}t + \frac{T}{2}t\right)\right\}, & 0 \le t < \frac{k_2}{B} \\ \exp\left\{i2\pi\frac{B}{T}\left(\frac{t^2}{2} - \frac{k_2}{B}t - \frac{T}{2}t\right)\right\}, & \frac{k_2}{B} \le t \le T \end{cases}$$

then for $k_1 < k_2$:

$$c_{k_1}(t)c^*_{k_2}(t) = \begin{cases} \exp\left\{i2\pi\left(k_2 - k_1\right)\frac{t}{T}\right\}, & 0 \le t \le \frac{k_1}{B} \\ \exp\left\{i2\pi\left(k_2 - k_1\right)\frac{t}{T} - Bt\right\}, & \frac{k_1}{B} \le t \le \frac{k_2}{B} \\ \exp\left\{i2\pi\left(k_2 - k_1\right)\frac{t}{T}\right\}, & \frac{k_2}{B} \le t \le T \end{cases}$$

Their inner product, for $k_1 < k_2$, reads:

$$\langle c_{k_1}, c_{k_2}\rangle = \int_0^T c_{k_1}(t)c^*_{k_2}(t)dt \tag{1.32}$$

After some algebra [7]:

$$\langle c_{k_1}, c_{k_2}\rangle = \frac{iNT\left(\exp\left(i2\pi\left(k_2 - k_1\right)\frac{k_1}{N}\right) - \exp\left(i2\pi\left(k_2 - k_1\right)\frac{k_2}{N}\right)\right)}{2\pi\left(\left(k_2 - k_1\right)^2 - N\left(k_2 - k_1\right)\right)} \neq 0 \tag{1.33}$$

7.

$$\langle c_{k_1}, c_{k_2}\rangle = \frac{T\exp\left(i2\pi\left(k_2 - k_1\right)\frac{t}{T}\right)\Big|_0^{\frac{k_1}{B}}}{i2\pi\left(k_2 - k_1\right)} + \frac{T\exp\left(i2\pi\left(\frac{\left(k_2 - k_1\right)}{T} - B\right)t\right)\Big|_{\frac{k_1}{B}}^{\frac{k_2}{B}}}{i2\pi\left(\left(k_2 - k_1\right) - N\right)} + \frac{T\exp\left(i2\pi\left(k_2 - k_1\right)\frac{t}{T}\right)\Big|_{\frac{k_2}{B}}^{T}}{i2\pi\left(k_2 - k_1\right)}$$

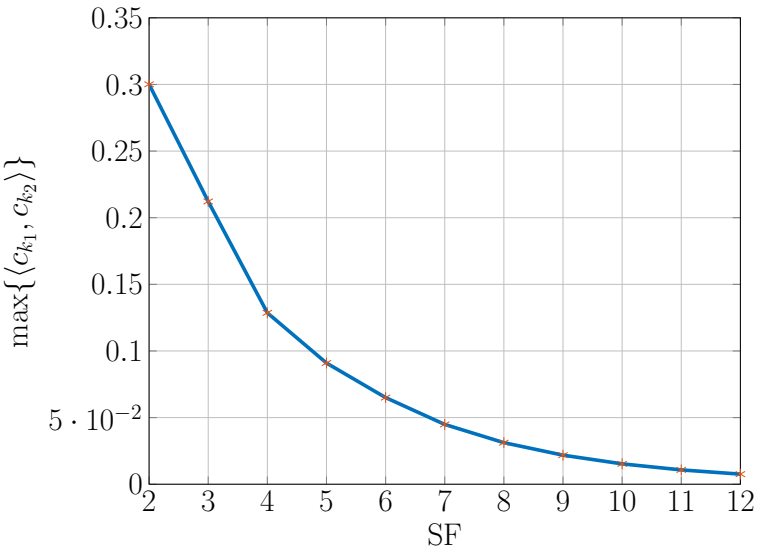



Figure 1.13 – Max inner product $\langle c_{k_1}, c_{k_2}\rangle$ for each SF.

The obtained expression for the inner product of the chirps $c_{k_1}(t)$ and $c_{k_2}(t)$ shows that the $2^{\mathrm{SF}}$-CSS set is not orthogonal.

The inner product was numerically simulated to obtain the maximum cross-correlation as a function of the spreading factor SF and an arbitrary bandwidth of $B = 125$ kHz (coefficients for a different bandwidth can be obtained using the relation $BT = 2^{\mathrm{SF}}$). Results in Figure 1.13 show that the larger the SF, the lower the cross correlation between the chirps.

#### 1.3.2.2 Digital CSS chirps

The digital version of the obtained chirp in Eq.(1.31), sampled at the time instants $t_n = \frac{n}{B}$ reads:

$$\begin{aligned} c_{k_0}[n] &= \begin{cases} \exp\left\{i\pi\left(\frac{n^2}{N} - \frac{2k_0 n}{N} + n\right)\right\}, & 0 \leq n < k_0 \\ \exp\left\{i\pi\left(\frac{n^2}{N} - \frac{2k_0 n}{N} - n\right)\right\}, & k_0 \leq n < N \end{cases} \\ &= \exp\left\{i\pi\left(\frac{n^2}{N} - \frac{2k_0 n}{N} + n\right)\right\} \end{aligned} \tag{1.34}$$

It is interesting to see that this expression corresponds to the up-chirp in Eq.(1.29), with $BT = N$, modulated by a pure frequency at $\frac{k_0}{N}$.

This means that on the receiver side, the information bits carried by $k_0$ can be recovered by multiplying the received chirp with the down-chirp $c_d[n]$ in Eq.1.29 (this operation is usually referred as "de-chirp") obtaining:

$$\begin{aligned} y_{k_0}[n] =& c_{k_0}[n] \times c_{\mathrm{d}}[n] \\ &= \left(\exp\left\{i\pi\left(\frac{n^2}{N} - \frac{2k_0 n}{N} + n\right)\right\}\right) \times \exp\left\{-i\pi\left(\frac{n^2}{N} - n\right)\right\} \\ &= \exp\left\{-i2\pi\frac{k_0 n}{N}\right\} \end{aligned} \tag{1.35}$$

This corresponds to a pure frequency as mentioned before. Indeed, taking the Discrete Fourier Transform (DFT) of $y_{k_0}[n]$ we have:

$$\begin{aligned} Y[k] &= \frac{1}{\sqrt{N}}\sum_{n=0}^{N-1} y_{k_0}[n]\exp\left\{-i2\pi\frac{kn}{N}\right\} \\ &= \frac{1}{\sqrt{N}}\sum_{n=0}^{N-1}\exp\left\{-i2\pi\frac{(k+k_0)n}{N}\right\} \\ &= \begin{cases} \sqrt{N} & \text{for } k+k_0 = mN, \quad m \in \mathbb{Z} \\ \frac{1}{\sqrt{N}}\frac{1-\exp\{-i2\pi(k+k_0)\}}{1-\exp\left\{-i2\pi\frac{(k+k_0)}{N}\right\}} = 0 & \text{otherwise} \end{cases} \\ &= \sqrt{N}\delta[k+k_0-N] \end{aligned} \tag{1.36}$$

with $0 \leq k \leq N-1$ and where $\delta[k]$ is the Kronecker delta function defined as:

$$\delta[k] = \begin{cases} 1, & k = 0 \\ 0, & \text{otherwise} \end{cases}$$

This means that all the energy is concentrated at the frequency bin $N - k_0$.

Although analog chirps were proven not to be orthogonal, digital chirps sampled at the $B$ rate are orthogonal. Another direct proof of this orthogonality is presented below. For $c_{k_1}[n]$ and $c_{k_2}[n]$ with $k_1 \neq k_2$ we have:

$$\begin{aligned} \langle c_{k_1}, c_{k_2}\rangle &= \sum_{n=0}^{N-1} c_{k_1}[n]c_{k_2}^*[n] \\ &= \sum_{n=0}^{N-1}\exp\left(i2\pi\left(k_2-k_1\right)n/N\right) \\ &= \frac{1-\exp\left(i2\pi\left(k_2-k_1\right)\right)}{1-\exp\left(i2\pi\left(k_2-k_1\right)/N\right)} = 0 \end{aligned}$$

### 1.3.3 Chirped-FSK

Since Eq.(1.34) is equivalent to a linear digital up-chirp (Eq.(1.29)) with $BT = N$, modulated by a pure frequency at $\frac{k_0}{N}$, we analyze now what a Chirped-FSK [9, 10] scheme looks like.

The $N-FSK$ modulation can be expressed as:

$$s_{k0}(t) = \exp\left\{-i2\pi\frac{k_0}{T}t\right\} \text{ with } 0 \leq t \leq T \text{ and } 0 \leq k_0 < N = 2^{\text{SF}} \tag{1.37}$$

Hence, the so-called Chirped-FSK waveform over the time duration $T$ is given by:

$$\begin{aligned} s_{c,k0}(t) &= s_{k0}(t) \times c_{\text{u}}(t) \\ &= \exp\left\{-i2\pi\frac{k_0}{T}t\right\} \times \exp\left\{i\pi\frac{B}{T}\left(t^2 - Tt\right)\right\} \\ &= \exp\left\{i2\pi\frac{B}{T}\left(\frac{t^2}{2} - \frac{k_0}{B}t - \frac{T}{2}t\right)\right\} \text{ with } 0 \leq t < T \end{aligned} \tag{1.38}$$

Contrarily to the CSS case (see Eq.(1.31)), there is no discontinuity at $t_{k_0} = \frac{k_0}{B}$.

**Orthogonality of Chirped-FSK.** For $k_1 \neq k_2$, $k_1, k_2 \in \{0, 1, \cdots, N-1\}$ the inner product reads:

$$\begin{aligned} \langle s_{c,k_1}, s_{c,k_2} \rangle &= \int_0^T s_{c,k_1}(t) s^*_{c,k_1}(t) dt \\ &= \int_0^T \exp\left\{i2\pi\frac{(k_2 - k_1)t}{T}\right\} dt \\ &= 0 \end{aligned}$$

which means that the Chirped-FSK set is orthogonal.

**From Chirped-FSK to $2^{\text{SF}}$-CSS.** The Chirped-FSK scheme seems to resemble the $2^{\text{SF}}$-CSS chirp from Eq.(1.31). The instantaneous frequency of this Chirped-FSK waveform is given by:

$$f_{s_c,k_0}(t) = \frac{1}{2\pi}\frac{d}{dt}\phi_{s_c,k_0}(t) \tag{1.39}$$

where

$$\phi_{s_c,k_0}(t) = 2\pi\frac{B}{T}\left(\frac{t^2}{2} - \frac{k_0}{B}t - \frac{T}{2}t\right) \tag{1.40}$$

and hence

$$f_{s_c,k_0}(t) = \frac{B}{T}\left(t - \frac{k_0}{B} - \frac{T}{2}\right) \tag{1.41}$$

The obtained instantaneous frequencies globally span the frequencies in the interval $[-\frac{3B}{2};\ +\frac{B}{2}]$ for $k_0 = 0$ and $k_0 = N-1$ as depicted in Figure 1.14. The Chirped-FSK instantaneous frequency covers a band interval who is $2B$ long, double than that of the $2^{\text{SF}}$-CSS scheme.

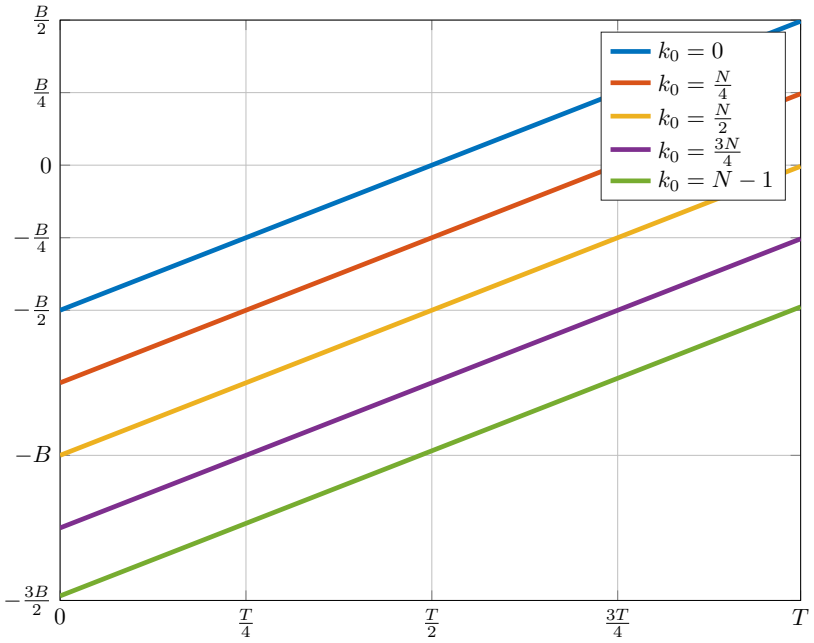



Figure 1.14 – Frequency trajectory of the Chirped-FSK waveform.

One way to reduce the covered band interval to $B$ would be to fold up the exceed of band to fall into the frequency interval $[-\frac{B}{2}; +\frac{B}{2}]$. From Eq.(1.41) we then have that:

$$\begin{aligned} f_{s_c,k_0}(t) &< -\frac{B}{2} \\ \implies \frac{B}{T}\left(t - \frac{k_0}{B} - \frac{T}{2}\right) &< -\frac{B}{2} \\ \implies t &< \frac{k_0}{B} \end{aligned} \tag{1.42}$$

This means that the band is exceeded in the interval $0 \leq t < \frac{k_0}{B}$. The folded version of Eq.(1.41) reads:

$$\begin{aligned} f^{(B)}_{s_c,k_0}(t) &= f_{s_c,k_0}(t) + B\mathbb{1}_{[0,\frac{k_0}{B}]}(t) \\ &= \begin{cases} \frac{B}{T}\left(t - \frac{k_0}{B} + \frac{T}{2}\right), & 0 \leq t < \frac{k_0}{B} \\ \frac{B}{T}\left(t - \frac{k_0}{B} - \frac{T}{2}\right), & \frac{k_0}{B} \leq t < T, \end{cases} \end{aligned} \tag{1.43}$$

which corresponds to the same expression as Eq.(1.30). This means that the $2^{\mathrm{SF}}$-CSS modulation is equivalent to a "Chirped-FSK" scheme with folded up bandwidth. This folding however is what destroys the orthogonality in the analog domain. This non orthogonality is not seen at the receiver side if and only if the receiver is perfectly synchronized, that there are no phase errors and that the received signals are sampled at the time instants $t_n = \frac{n}{B}$. Let us now show what happens if we transmit the $2^{\mathrm{SF}}$-CSS signals over an AWGN channel.

### 1.3.4 Transmission over an AWGN channel

Assuming a transmission over an AWGN (Additive White Gaussian Noise) channel, the received signal, after ADC is given by:

$$r_{k_0}[n] = c_{k_0}[n] + w[n]$$

where (*cf.*, section 1.1.3.1) the noise $w[n]$ is discrete, gaussian and white (when sampled at $B$) and its probability density function (pdf) is $\mathcal{CN}(0, \sigma_n^2)$ (with $\sigma_n^2 = 2N_0B$).

Then, after de-chirp, we obtain:

$$\begin{aligned} y_{k_0}[n] &= r_{k_0}[n] \times c_{\mathrm{d}}[n] \\ &= (c_{k_0}[n] + w[n]) \times c_{\mathrm{d}}[n] \\ &= \exp\left\{-i2\pi\frac{k_0 n}{N}\right\} + w_c[n] \end{aligned} \tag{1.44}$$

where $w_c[n] = w[n]c_d[n]$ corresponds to the "de-chirped" version of the noise. This operation does not change the statistical properties of the noise respect to those of $w[n]$; indeed since $w[n]$ is white, its ACS is given by $\Gamma_w[m] = \sigma_n^2\delta[m]$ and hence the autocorrelation function of $w_c[n]$ is given by:

$$\begin{aligned} \Gamma_{w_c}[n,m] &= \mathbb{E}\left\{w_c[n]w_c^*[n-m]\right\} \\ \Gamma_{w_c}[n,m] &= \mathbb{E}\left\{w[n]c_{\mathrm{d}}[n]w^*[n-m]c_{\mathrm{d}}^*[n-m]\right\} \\ \Gamma_{w_c}[n,m] &= \exp\left\{i\pi\left(\frac{(n-m)^2}{N} - (n-m) - \frac{n^2}{N} + n\right)\right\}\mathbb{E}\left\{w[n]w^*[n-m]\right\} \\ \Gamma_{w_c}[n,m] &= \exp\left\{i\pi\left(\frac{m^2}{N} - \frac{2nm}{N} + m\right)\right\}\Gamma_w[m] \\ \Gamma_{w_c}[n,m] &= \exp\left\{i\pi\left(\frac{m^2}{N} - \frac{2nm}{N} + m\right)\right\}\sigma_n^2\delta[m] \\ \Gamma_{w_c}[n,m] &= \sigma_n^2\delta[m] = \Gamma_{w_c}[m] \end{aligned}$$

It is noteworthy to mention that the whiteness and stationary of $w_c[n]$ is only guaranteed at the sampling rate $B$ and under white noise assumption at the receiver analog Front-end. Otherwise, the properties of $w_c[n]$ would be different (cyclostationary for example).

The N-DFT of the de-chirped signal in Eq.(1.44) reads:

$$\begin{aligned} Y[k] &= \frac{1}{\sqrt{N}}\sum_{n=0}^{N-1} y_{k_0}[n]\exp\left\{-i2\pi\frac{kn}{N}\right\} \\ &= \frac{1}{\sqrt{N}}\sum_{n=0}^{N-1}\left(\exp\left\{-i2\pi\frac{(k+k_0)n}{N}\right\} + w_c[n]\exp\left\{-i2\pi\frac{kn}{N}\right\}\right) \\ &= \frac{1}{\sqrt{N}}\left\{\sum_{n=0}^{N-1}\left(\exp\left\{-i2\pi\frac{(k+k_0)n}{N}\right\}\right) + \sum_{n=0}^{N-1}\left(w_c[n]\exp\left\{-i2\pi\frac{kn}{N}\right\}\right)\right\} \\ &= \begin{cases} \sqrt{N} + b[k] & \text{for } k + k_0 = mN, m \in \mathbb{Z} \\ b[k] & \text{otherwise} \end{cases} \end{aligned} \tag{1.45}$$

where $b[k] \sim \mathcal{CN}(0, \sigma_n^2)$. Assuming that there is a perfect synchronization with no phase error, data is retrieved through coherent demodulation obtaining the estimate of $k_0$ as follows:

$$\widehat{k_0} = N - \operatorname{argmax} \Re\{Y[k]\} \tag{1.46}$$

Errors occurs if there exists an index $k \neq N - k_0$ such that $\Re\{b[k]\} > \sqrt{N} + \Re\{b[N - k_0]\}$. Say $X_k = \Re\{b[k]\}$ with pdf $f_X(x) \sim \mathcal{N}(0, \frac{\sigma_n^2}{2})$ and $Y = \sqrt{N} + \Re\{b[N - k_0]\}$ with pdf $f_Y(y) \sim \mathcal{N}(\sqrt{N}, \frac{\sigma_n^2}{2})$. The error probability reads:

$$\begin{aligned}
P_e &= \Pr\left(\exists\, k \neq N - k_0 \mid X_k > Y\right) \\
P_e &= \int_{-\infty}^{+\infty} \left(1 - \Pr\left(\max\left\{X_k\right\} \leq y\right)\right) f_Y(y) dy \\
P_e &= \int_{-\infty}^{+\infty} \left(1 - \prod_{\substack{k=0 \\ k \neq N - k_0}}^{N-1} \Pr\left(X_k \leq y\right)\right) f_Y(y) dy \\
P_e &= \int_{-\infty}^{+\infty} \left(1 - \left(\Pr\left(X_k \leq y\right)\right)^{N-1}\right) f_Y(y) dy
\end{aligned} \tag{1.47}$$

where $\Pr(X_k \leq y)$ reads:

$$\begin{aligned}
\Pr(X_k \leq y) &= \int_{-\infty}^{y} f_X(x) dx \\
\Pr(X_k \leq y) &= 1 - \int_{y}^{\infty} \frac{1}{\sqrt{\pi}\sigma_n} \exp\left\{-\frac{x^2}{\sigma_n^2}\right\} dx \\
\Pr(X_k \leq y) &= 1 - Q\left(y\sqrt{\frac{2}{\sigma_n^2}}\right),
\end{aligned} \tag{1.48}$$

$Q(x)$ being the tail distribution function of the standard normal distribution:

$$Q(x) = \frac{1}{\sqrt{2\pi}} \int_{x}^{+\infty} \exp\left(-\frac{u^2}{2}\right) du$$

Replacing Eq.(1.48) in Eq.(1.47), the probability of error becomes:

$$\begin{aligned}
P_e &= \int_{-\infty}^{+\infty} \left(1 - \left(1 - Q\left(y\sqrt{\frac{2}{\sigma_n^2}}\right)\right)^{N-1}\right) f_Y(y) dy \\
P_e &= \int_{-\infty}^{+\infty} \left(1 - \left(1 - Q\left(y\sqrt{\frac{2}{\sigma_n^2}}\right)\right)^{N-1}\right) \frac{\exp\left\{-\frac{\left(y-\sqrt{N}\right)^2}{\sigma_n^2}\right\}}{\sqrt{\pi}\sigma_n} dy
\end{aligned} \tag{1.49}$$

The obtained error probability expression cannot be made simpler and hence the integral must be solved numerically. The obtained result in Eq.(1.49) is the same than the error probability of the orthogonal signaling modulations [11] [8][12] *e.g.*, the Frequency Shift Keying (FSK) modulation.

With that said, the studied $2^{\mathrm{SF}}$-CSS modulation can be regrouped with orthogonal signaling techniques under the assumed conditions, *i.e.*, perfect synchronization and no phase error where the digital chirps are guaranteed to be orthogonal (sampled at the rate $B$). Further studies are needed to determine what happens in multipath scenario, with synchronization errors and with other noise characteristics.

## 1.4 Conclusion of the Chapter

In this chapter we have seen a general overview of the different blocks that compose the digital communication schemes. We presented the two families of digital modulation schemes: (i) the linear modulations (*e.g.*, PSK, QAM) and (ii) the non-linear modulations (*e.g.* FSK, PPM). We have introduced the analog front-end and the general model for the AWGN.

Then, we presented the OFDM with details about the symbols generation and the signal generation. We explained also the utility of the Cyclic Prefix and how it transforms the transmission through the channel into a circular convolution.

Finally we analyzed the LoRa modulation. We have started from the definition of the linear chirps that are the basis of the LoRa modulation. Then we introduced the LoRa chirps and we analyzed the orthogonality of both the analog and the digital signals. We obtained that the LoRa chirps are not orthogonal in the analog domain but they are orthogonal in the digital domain only when sampled at the sample rate $B$.

Afterwards, we compared the LoRa modulation with a Chirped-FSK scheme and concluded that the LoRa modulation can be obtained by folding up the instantaneous frequency of such a Chirped-FSK scheme.

8. Precisely in Chapter 4: Optimum Receivers for AWGN Channels.

Chapter 2

# The PAPR issue for OFDM

## Contents



In this chapter we focus on the Peak-to-Average Power Ratio (PAPR) issue for Multi-Carrier Modulations like the previously described Orthogonal Frequency Division Multiplexing (OFDM) technique. The PAPR is here described and analyzed. Then, some PAPR reduction solutions are presented. Finally we propose a novel OFDM PAPR reduction scheme: called "GreenOFDM" [13, 14]. The distribution of the PAPR of the novel scheme is studied and compared with some of the best existing solutions.

## 2.1 The PAPR: definition and properties for OFDM symbols

In the current section, the PAPR is introduced, then the reason why it became an issue for OFDM is explained.

### 2.1.1 PAPR of the complex symbols in baseband

The OFDM is a very popular communication technique that has been deployed in different standards going from wired systems like ADSL and VDSL to wireless systems like Wi-Fi, LTE 4G and the undergoing 5G. OFDM based communication techniques like NB-IoT and IEEE 802.15.4g MR-OFDM are used for the IoT. However, the major drawback of OFDM is its Peak-to-Average Power Ratio (PAPR).

The PAPR is a dimensionless measure of a waveform that indicates how *extreme* the peaks of the signal are. The PAPR can be measured in different ways [15]. In this thesis we focus about the PAPR on the symbol basis.

#### 2.1.1.1 Baseband Nyquist sampling PAPR representation

The PAPR of the baseband OFDM symbol (Eq.(1.19)) introduced in the previous chapter is given by:

$$\text{PAPR} = \frac{\max_n \left\{|x[n]|^2\right\}}{\mathbb{E}\left\{|x[n]|^2\right\}} \quad \text{with } 0 \leq n \leq N-1 \tag{2.1}$$

where $\mathbb{E}\left\{|x[n]|^2\right\} = \mathbb{E}\left\{|A_k|^2\right\}$ [1] for $A_k$ i.i.d. random variables ($\{A_k\} \in 2^q - QAM$ with $q = 2m, m \in \mathbb{N}^*$).

**The OFDM symbols suffer from high PAPR.** This issue results from subcarriers that add in-phase at certain time samples, resulting in peaks that are much larger than the signal average power. Figure 2.1 depicts the squared modulus of an OFDM symbol with $N = 64$ subcarriers and BPSK subcarrier mapping. As it can be seen in this example, the maximum value of squared modulus of the OFDM symbol is around 10 times higher than the symbol average power. However this high peak is only present at one time instant of the OFDM symbol.

**The PAPR is a random variable.** Indeed, since $A_k$ are random variables, $x[n]$ and $\max_n \left\{|x[n]|^2\right\}$ are random variables too. For $N$ sufficiently large, the Central Limit Theorem approximatively holds so that $\Re\{x[n]\}$ and $\Im\{x[n]\}$ tends to Gaussian distributions [2] $\mathcal{N}\left(0, \frac{\sigma^2}{2}\right)$ where $\sigma^2 = \mathbb{E}\left\{|x[n]|^2\right\}$ and hence $|x[n]|^2$ tends to a $\chi^2$ distribution with two degrees of freedom.

For the real and imaginary components of $x[n]$ and its squared modulus $|x[n]|^2$, figure 2.2 depicts both the simulated distributions and the theoretical probability density functions

1. $$\mathbb{E}\left\{|x[n]|^2\right\} = \frac{1}{N}\sum_{k=0}^{N-1}\sum_{l=0}^{N-1}\mathbb{E}\left\{A_k A_l^*\right\}\exp\left\{i2\pi\frac{(k-l)n}{N}\right\} = \frac{1}{N}\sum_{k=0}^{N-1}\mathbb{E}\left\{|A_k|^2\right\} = \mathbb{E}\left\{|A_k|^2\right\}$$

2. Details about their variances are developed in section 2.1.2.

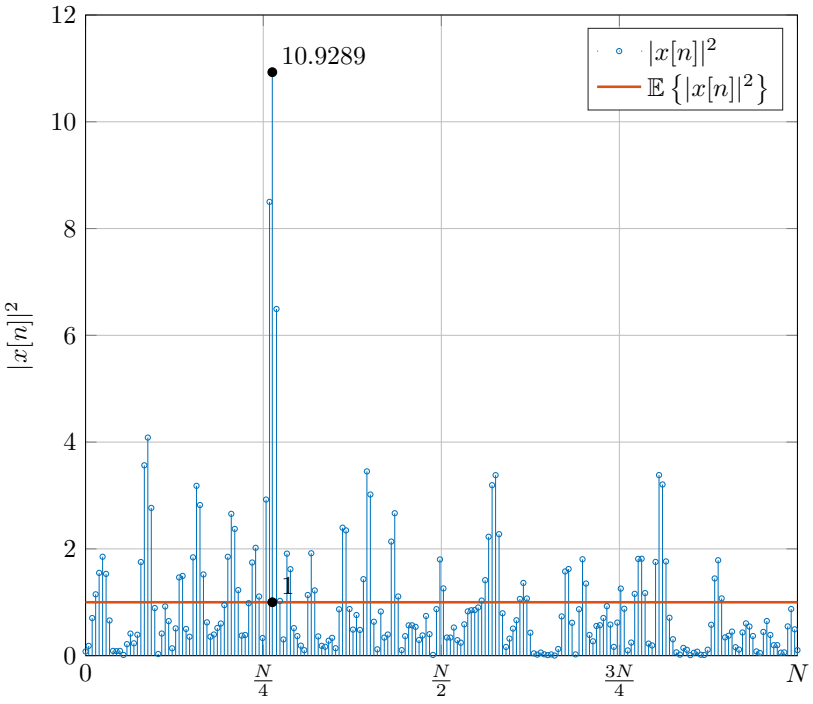
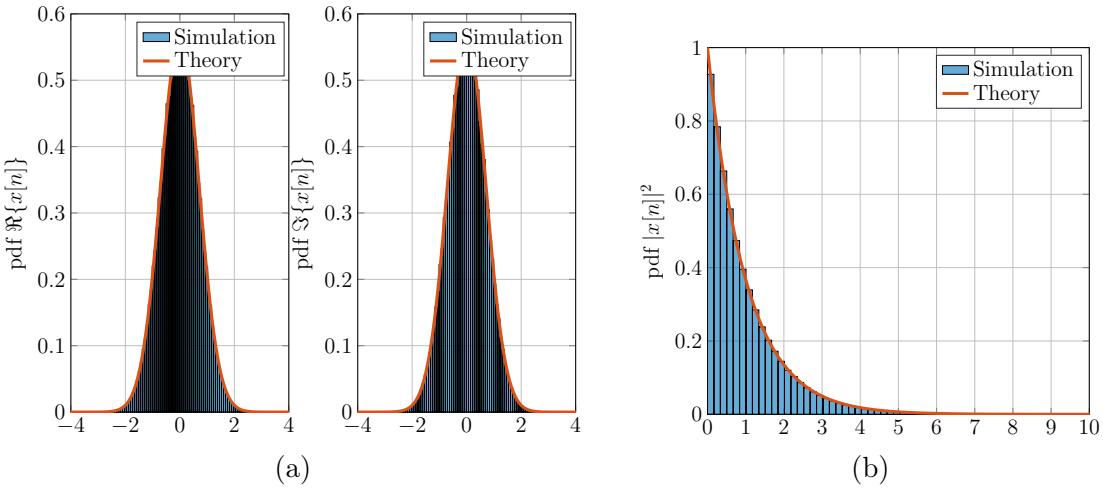



Figure 2.1 – Time domain example of an OFDM Symbol squared modulus $|x[n]|^2$.

(pdf) obtained when the distribution of $x[n]$ is assumed to be Gaussian. This small example tends to show that the gaussian approximation is good even for small values of $N$ ($N = 64$ in this example).

**Statistical properties of the PAPR.** The PAPR is commonly analyzed by its Complementary Cumulative Distribution Function (CCDF), *i.e.* the probability that the PAPR of an OFDM symbol exceeds a predetermined threshold $\gamma$. First note that, since $A_k$ are independent random variables $x[n]$ are independent random variables. Hence:

$$\begin{aligned}
\mathrm{CCDF}(\gamma) &= \Pr\{\mathrm{PAPR} > \gamma\} \\
&= 1 - \Pr\{\mathrm{PAPR} \leq \gamma\} \\
&= 1 - \prod_{n=0}^{N-1} \Pr\left\{\frac{|x[n]|^2}{\mathbb{E}\{|x[n]|^2\}} \leq \gamma\right\} \\
&= 1 - (\Pr\{Z \leq \gamma\})^N
\end{aligned}$$




(a) (b)

Figure 2.2 – Probability Density Functions of (2.2a) the real and imaginary parts of $x[n]$ and (2.2b) its squared modulus $|x[n]|^2$.

with $Z = \frac{|x[n]|^2}{\mathbb{E}\{|x[n]|^2\}}$ and [3] $f_Z(z) = \exp\{-z\}$ for $z \in [0, +\infty)$. Therefore:

$$\begin{aligned} \mathrm{CCDF}(\gamma) &= 1 - (\Pr\{Z \leq \gamma\})^N \\ &= 1 - \left(\int_0^\gamma \exp\{-z\}\, dz\right)^N \\ &= 1 - (1 - \exp\{-\gamma\})^N \end{aligned} \tag{2.2}$$

**Maximum PAPR with a given probability.** In practice, the PAPR is dimensioned with a threshold $\gamma_{\mathrm{th}}$ providing a given probability of clipping (in other words, the probability $\mathrm{CCDF}(\gamma_p) = \Pr\{\mathrm{PAPR} > \gamma_p\}$). Generally a probability of $10^{-3}$ or $10^{-4}$ is currently used (1 OFDM symbol every $1,000$ or 1 OFDM symbol every $10,000$). This threshold $\gamma_p$ is then used to dimension the system circuitry and to determine the complexity of the PAPR reduction digital implementation.

### 2.1.1.2 The need of oversampling to correctly dimension the analog section

Peaks might appear in the analog domain after DAC in symbols $x_{\mathrm{BB}}(t)$. Figure 2.3 depicts an example for which the continuous time symbol $x_{\mathrm{BB}}(t)$ exhibits a higher peak power than the discrete time symbol $x[n]$. This has an impact about the $\mathrm{CCDF}(\gamma)$. The $\mathrm{CCDF}(\gamma)$

3. We have $Z = \frac{|x[n]|^2}{\mathbb{E}\left\{|x[n]|^2\right\}} = \frac{(\Re\{x[n]\})^2 + (\Im\{x[n]\})^2}{2\frac{\mathbb{E}\left\{|x[n]|^2\right\}}{2}} = \frac{Y}{2}$.
with $f_Y(y) = \frac{\exp\left\{-\frac{y}{2}\right\}}{2}$, hence:

$$f_Z(z) = f_Y(y)\left|\frac{dy}{dz}\right| = \frac{\exp\left\{-\frac{y}{2}\right\}}{2} 2 = \exp\{-z\}$$

estimation is required in order to correctly dimension the High Power Amplifier (HPA) and for the calculation of the effective number of bits of the DAC.

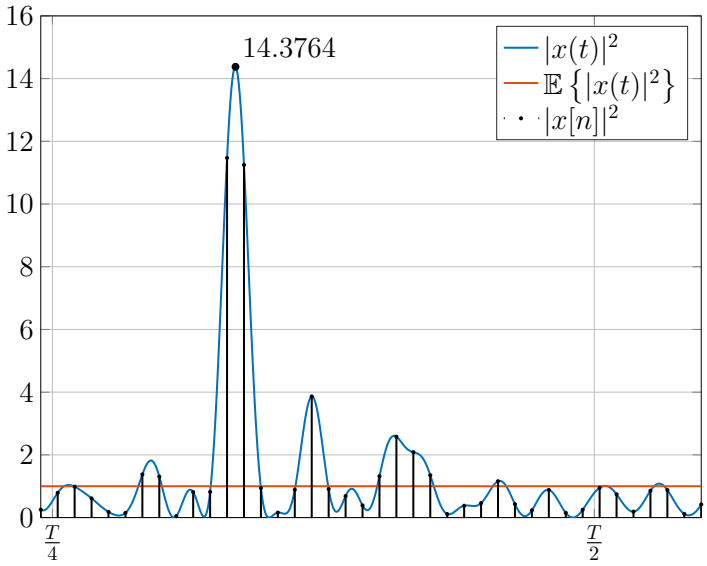



Figure 2.3 – Instantaneous power for $x(t)$ and $x[n]$.

In the sequel we represent the analog symbols with an oversampled version of the discrete symbols.

An oversampled version of $x[n]$ can be obtained by means of zero-padding $LN - IFFT$ with a factor $L$:

$$x[n] = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} A_k \exp\left\{ i2\pi \frac{kn}{LN} \right\}, \quad 0 \leq n \leq LN - 1$$

An oversampling factor $L = 4$ is used to capture peaks of the analog signal [16]. The time domain samples of the oversampled OFDM symbols are now correlated.

In [17, 18] the authors show that compared to Eq.(2.2), the expression of the CCDF for the analog symbols is very well approximated by:

$$\text{CCDF}(\gamma) \approx 1 - (1 - \exp\{-\gamma\})^{2.8N} \tag{2.3}$$

As it can be seen from the previous equation, the CCDF of PAPR of OFDM symbols depends mainly of the number of subcarriers. The higher the number of subcarriers, the higher the probabilities to exceed the PAPR threshold $\gamma$.

Figure 2.4 depicts the simulation result of the CCDF estimation for different values of $N$ and for a total $10^6$ OFDM simulated symbols. It can be seen that for $N > 64$ the result of simulations (dotted lines) matches [4] the curves of equation (2.3) (solid lines). This is because

4. The simulated CCDF is presented until $10^{-4}$ because of computational limitations: $10^{s+2}$ symbols are approximately required for a probability of $10^{-s}$.

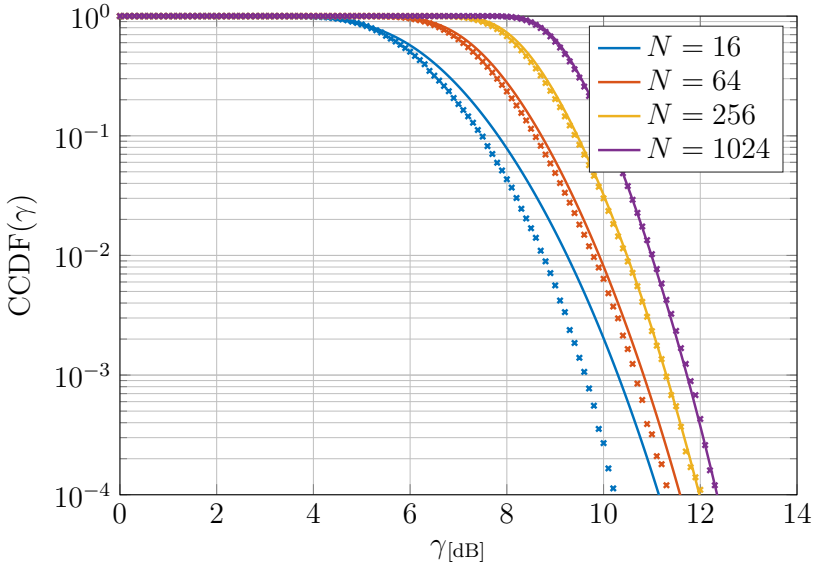



Figure 2.4 – CCDF($\gamma$) for different number of subcarriers $N$. In dotted lines results from computer simulations for $10^6$ OFDM symbols and in solid lines the plot of equation (2.3).

starting from $N > 64$, $\Re\{x[n]\}$ and $\Im\{x[n]\}$ are closely normally distributed (Central Limit Theorem). Also we can see that the PAPR of one every thousand OFDM symbols (or a probability of occurrence of $10^{-3}$) exceeds 10dB.

As it was already mentioned, a high PAPR in OFDM is troublesome because it requires a higher effective number of bits for the DAC [19, 20]; a high PAPR is also a problem for the HPA and is the main contributor of high power consumption and saturation of OFDM transceivers.

However, so far we only considered a PAPR calculated on the complex valued signal in baseband.

At the transmitter circuit level, several signals are involved: the real and imaginary parts of the complex signal and the band translated signal around the carrier frequency. These 3 signals are affected by the different PAPRs, in order to correctly dimension all components, the respective PAPR must be calculated and analyzed. This is the aim of the following sections.

### 2.1.2 PAPR of real valued I/Q signals — DAC dimensioning

The OFDM baseband signal $x[n]$ is decomposed into its In-phase and Quadrature-phase components that correspond to $\Re\{x[n]\}$ and $\Im\{x[n]\}$ respectively. To determine the PAPR by each component we have first that:

$$x_I[n] = \Re\{x[n]\} = \frac{1}{\sqrt{4N}} \sum_{k=0}^{N-1} \left( A_k \exp\left( i2\pi \frac{kn}{LN} \right) + A_k^* \exp\left( -i2\pi \frac{kn}{LN} \right) \right)$$

Hence, for $A_k$ i.i.d., zero-mean and circular random variables at order two (the moments are equal to 0 until the second moment) [5] we have:

$$\mathbb{E}\left\{(x_I[n])^2\right\} = \frac{1}{4N}\sum_{k=0}^{N-1}\sum_{l=0}^{N-1}\mathbb{E}\left\{\left(A_k \exp\left\{\frac{i2\pi kn}{LN}\right\} + A_k^* \exp\left\{-\frac{i2\pi kn}{LN}\right\}\right) \cdot \left(A_l^* \exp\left\{-\frac{i2\pi ln}{LN}\right\} + A_l \exp\left\{\frac{i2\pi ln}{LN}\right\}\right)\right\}$$

where $\mathbb{E}\{A_k A_l\} = \mathbb{E}\{A_k^* A_l\} = \mathbb{E}\{A_k A_l^*\} = \mathbb{E}\{A_k^* A_l^*\} = 0$ for $k \neq l$ (i.i.d and zero-mean) and $\mathbb{E}\left\{(A_k)^2\right\} = \mathbb{E}\left\{(A_k^*)^2\right\} = 0$ (circularity [21]), hence:

$$\mathbb{E}\left\{(x_I[n])^2\right\} = \frac{1}{4N}\sum_{k=0}^{N-1} 2.\mathbb{E}\left\{|A_k|^2\right\} = \frac{1}{2}\mathbb{E}\left\{|A_k|^2\right\}$$

The same analytical derivation applies for the Quadrature-phase component $x_Q[n] = \Im\{x[n]\}$, and thus:

$$\mathbb{E}\{(x_I[n])^2\} = \mathbb{E}\left\{(x_Q[n])^2\right\} = \frac{1}{2}\mathbb{E}\left\{|A_k|^2\right\} \tag{2.4}$$

By definition:

$$\text{PAPR}^{I/Q} = \frac{\max\limits_{0\leq n\leq LN-1}\left((x_{I/Q}[n])^2\right)}{\mathbb{E}\left\{(x_{I/Q}[n])^2\right\}}$$

where $I/Q$ means the $I$ or the $Q$ component. Defining $\Delta_{I/Q}$ as the difference between the $\text{PAPR}^{I/Q}$ and the PAPR in baseband ($\text{PAPR}_{BB}$) in decibels:

$$\begin{aligned}\Delta_{I/Q}[dB] &= \text{PAPR}^{I/Q}[\text{dB}] - \text{PAPR}_{BB}[\text{dB}] \\ &= 10\log_{10}\left(\frac{\max\limits_n\left((x_{I/Q}[n])^2\right)}{0.5\mathbb{E}\{(x[n])^2\}}\right) - 10\log_{10}\left(\frac{\max\limits_n\left(|x[n]|^2\right)}{\mathbb{E}\{(x[n])^2\}}\right) \\ &= 3[\text{dB}] + 10\log_{10}\left(\frac{\max\limits_n\left((x_{I/Q}[n])^2\right)}{\max\limits_n\left(|x[n]|^2\right)}\right)\end{aligned}$$

Since $0 < \frac{\max\limits_n\left((x_{I/Q}[n])^2\right)}{\max\limits_n(|x[n]|^2)} \leq 1$, then $\Delta_{I/Q}$ is maximum when $\max\limits_n\left((x_{I/Q}[n])^2\right) = \max\limits_n\left(|x[n]|^2\right)$. Therefore:

$$\max(\Delta_{I/Q}[\text{dB}]) = 3\text{dB}$$

The conclusion is that in order to avoid any clipping at the inputs of the DACs:

$$\text{PAPR}^{I/Q}[\text{dB}] = \text{PAPR}_{BB}[\text{dB}] + 3\text{dB} \tag{2.5}$$

must be considered before determining the Effective Number of Bits of the DACs.

5. Taking PSK modulation as an example, the following analysis is not true for the BPSK case.

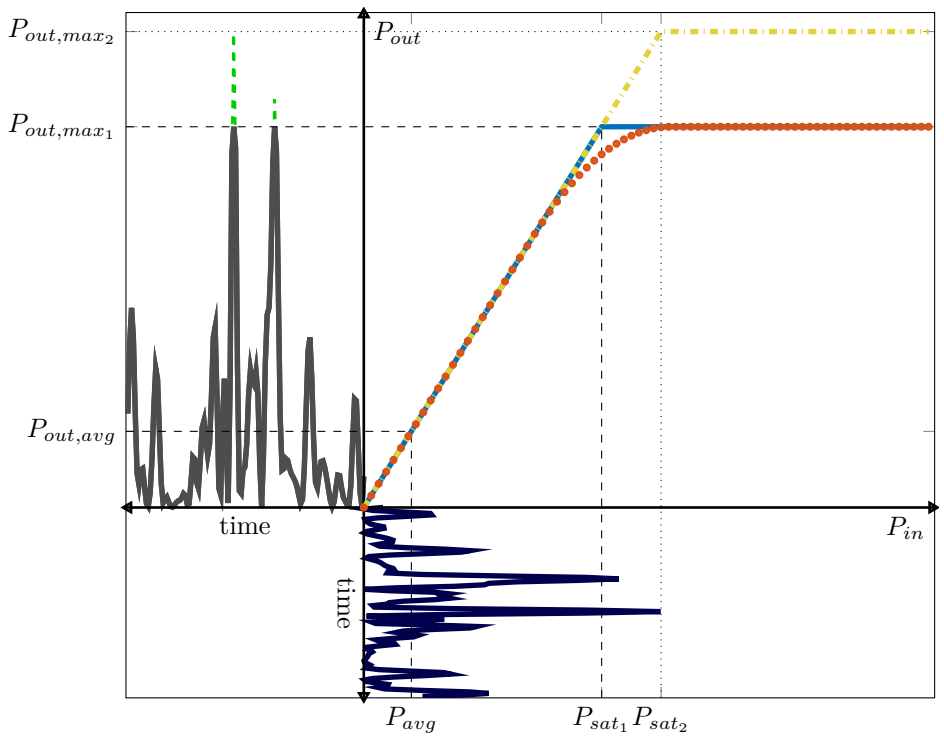



Figure 2.5 – On the upper right side, the ideal and real input-output response of the Power Amplifier. Below (in blue), OFDM symbols with high PAPR at the input of the PA. On the left (in black), the output of the PA.

### 2.1.3 PAPR of the RF signal — PA dimensioning

Let us first explain why non-linearities and distortions appear when the PA is wrongly dimensioned. Figure 2.5 plots the PA's input-output response in solid blue lines for the ideal case and in red dotted line for the real case. Focusing on the ideal case, it is revealed that the input-output response is mainly linear until a saturation point called $P_{sat_1}$. All input signals above this level are clipped to the provided value $P_{out,max_1}$, generating both in-band distortion and out-of-band radiation.

In the same example, there are also OFDM symbols (in solid blue line) that are to be amplified by the ideal PA. The results are amplified versions (depicted in gray) which are clipped at the time instants of power exceeding $P_{sat_1}$. Extending the linear region of the PA towards $P_{sat_2} > P_{sat_1}$ avoids symbol clipping at the expenses of higher power consumption and more powerful transistors. Information of the CCDF($\gamma$) can help at maintaining the input signal into the linear region and in the best cases, to reduce any saturation and hence to reduce the overall power consumption of the system.

The OFDM signal after DAC is modulated by a carrier frequency $f_c$ as presented in section 1.1.2 obtaining a pass-band signal $x_{\mathrm{RF}}(t)$. The PAPR of $x_{\mathrm{RF}}(t)$ is the key factor to specify the analog Front-end PA. A wrong dimensioning of this component leads to nonlinear distortions

that degrade the system performance [22].

The OFDM pass-band equivalent symbol is defined by $x_{RF}(t) = \Re\{x(t).\exp\{i\omega_c t\}\}$ [6] with $t \in [0;T]$, and $\omega_c = 2\pi f_c$ where $f_c$ is the carrier frequency (see Figure 1.8). By definition:

$$\text{PAPR}_{RF} = \frac{\max\limits_{0\leq t<T}(|x_{RF}(t)|^2)}{\mathbb{E}\{|x_{RF}(t)|^2\}}$$

where,

$$\begin{aligned}\mathbb{E}\left\{|x_{RF}(t)|^2\right\} &= \mathbb{E}\left\{|\Re\{x(t).\exp\{i\omega_c t\}\}|^2\right\}\\ &= \mathbb{E}\left\{(\ x_I(t).\cos(\omega_c t) - x_Q(t).\sin(\omega_c t)\ )^2\right\}\\ &= \mathbb{E}\left\{(x_I(t))^2\right\}\cos^2(\omega_c t) - 2\mathbb{E}\left\{x_I(t)x_Q(t)\right\}\cos(\omega_c t)\sin(\omega_c t) + \mathbb{E}\left\{(x_Q(t))^2\right\}\sin^2(\omega_c t)\end{aligned}$$

and:

$$x_I(t) = \Re\{x(t)\} = \frac{1}{2\sqrt{N}}\sum_{k=0}^{N-1}\left(A_k\exp\{i2\pi f_k t\} + A_k^*\exp\{-i2\pi f_k t\}\right)$$

$$x_Q(t) = \Im\{x(t)\} = \frac{1}{2\sqrt{N}i}\sum_{k=0}^{N-1}\left(A_k\exp\{i2\pi f_k t\} - A_k^*\exp\{-i2\pi f_k t\}\right)$$

Since the I and Q components bear independent information, we have:

$$\mathbb{E}\{x_I(t)x_Q(t)\} = 0$$

Back to $\mathbb{E}\left\{|x_{RF}(t)|^2\right\}$, since from Eq.(2.4) $\mathbb{E}\left\{(x_I(t))^2\right\} = \mathbb{E}\left\{(x_Q(t))^2\right\} = \frac{1}{2}\mathbb{E}\left\{|A_k|^2\right\}$, then:

$$\begin{aligned}\mathbb{E}\left\{|x_{RF}(t)|^2\right\} &= \mathbb{E}\left\{(x_I(t))^2\right\}.\left(\cos^2(\omega_c t) + \sin^2(\omega_c t)\right)\\ &= \mathbb{E}\left\{(x_Q(t))^2\right\}.\left(\cos^2(\omega_c t) + \sin^2(\omega_c t)\right)\\ &= \frac{1}{2}\mathbb{E}\left\{|A_k|^2\right\}\end{aligned}$$

In polar coordinates $x(t) = |x(t)|.\exp\{i\varphi(t)\}$, where $\varphi(t) = \arctan\left(-\frac{x_Q(t)}{x_I(t)}\right)$. Therefore, the numerator of $\text{PAPR}_{RF}$ can be written as:

$$\begin{aligned}\max_t(|x_{RF}(t)|^2) &= \max_t\left(\Re\{|x(t)|.\exp\{i(\omega_c t + \varphi(t))\}\}^2\right)\\ &= \max_t\left(|x(t)|^2.\cos^2(\omega_c t + \varphi(t))\right)\end{aligned}$$

Assuming $f_c \gg B$, where $B$ is the base-band OFDM signal bandwidth, smalls time-domain sections of the base-band OFDM symbol can be considered to be a constant over many periods of the carrier frequency and then:

$$\max_t(|x_{RF}(t)|^2) \cong \max_t(|x(t)|^2) \tag{2.6}$$

6. $x(t)$ is defined in Eq.(1.18). It is given by: $x(t) = x_I(t) + ix_Q(t) = \frac{1}{\sqrt{N}}\sum_{k=0}^{N-1}A_k\exp\{i2\pi f_k t\}$

From Eq.(2.6) we conclude that:

$$\begin{aligned}\mathrm{PAPR}_{RF} &= \frac{\max_t \left(|x(t)|^2\right)}{\frac{1}{2}\mathbb{E}\left\{|A_k|^2\right\}} \\ &= 2\frac{\max_t \left(|x(t)|^2\right)}{\mathbb{E}\left\{|A_k|^2\right\}} \\ &= 2\mathrm{PAPR}_{BB}\end{aligned}$$

or equivalently in decibels:

$$\mathrm{PAPR}_{RF}[\mathrm{dB}] = \mathrm{PAPR}_{BB}[\mathrm{dB}] + 3\mathrm{dB}$$

## 2.2 Overview of some existing PAPR reduction techniques

As already mentioned, PAPR reduction techniques allow to:

— reduce the power consumption of the PA. A wrong dimensioning of the PA induces, as previously mentioned, non linearities and thus distortions on the signal to be transmitted and out-of-band interference.
— reduce the effective number of bits of the DAC.

There are different PAPR reduction schemes [23, 24] that can be classified in different categories as follows:

— Signal Distortion techniques: *Clipping and Filtering* [25, 26] and Companding [27].
— Multiple Signaling and probabilistic techniques: *SeLected Mapping (SLM)* [28, 29], Tone Injection [30], Tone Reservation [31, 32], Active Constellation Extension [33], Partial Transmit Sequence [34, 35].
— Pre-coding techniques: *Single Carrier Frequency Division Multiple Access (SC-FDMA)*[36, 37] and Carrier Interferometry OFDM [38, 39].

This list is not exhaustive but shows that during the last two decades there has been many distinct propositions in PAPR reduction techniques. In this section, one technique of each 'category' will be introduced:

(i) The Clipping and Filtering signal distortion technique,

(ii) The SLM-OFDM multiple signaling technique, and

(iii) the SC-FDMA pre-coding technique.

### 2.2.1 Clipping and Filtering

The most intuitive solution for reducing the PAPR is to clip the peaks to a predetermined value. This value is chosen to achieve a given probabilty of clipping, it is obtained using the

CCDF of the PAPR. The clipped signal is given by:

$$x_c[n] = \begin{cases} x[n], & |x[n]| \leq A_{clip} \\ A_{clip} \exp\{i\psi_x[n]\}, & |x[n]| > A_{clip} \end{cases}$$

where $\psi_x[n]$ is the phase of $x[n]$ and $A_{clip}$ corresponds to the chosen clipping amplitude not to exceed the PAPR equal to $\gamma_p = \frac{A_{clip}^2}{\mathbb{E}\{|x[n]|^2\}}$.

The drawback is that brute force clipping, just as mentioned before for the peaks exceeding $P_{sat}$ of the PA, introduces in-band distortions and out-of-band radiation that corrupt the adjacent channels. To overcome this last issue, spectral filtering is employed. However, spectral filtering causes peak re-growth above $A_{clip}$. An iterative solution allows to maintain the resulting peaks below or at worst slightly higher than $A_{clip}$.

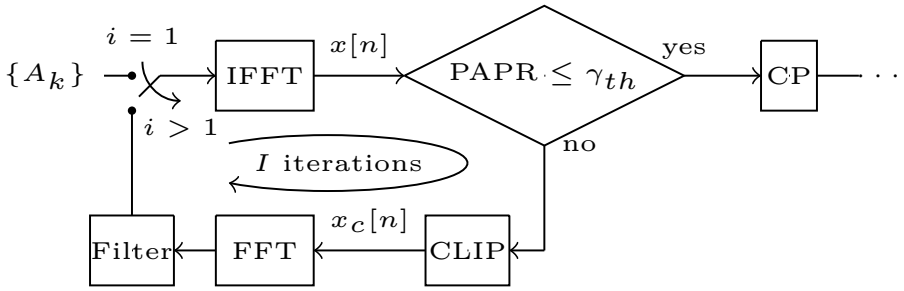



Figure 2.6 – Clipping and Filtering block diagram.

Figure 2.6 depicts the block diagram of the iterative Clipping and Filtering technique. In this method, at worst $2.I$-IFFTs are computed to generate a symbol with an acceptable PAPR. However this technique suffers from BER degradation due to the in-band noise coming from the iterative clipped peaks.

### 2.2.2 SLM-OFDM

In classical SLM-OFDM $U$ different symbols are generated from the same data set $\{A_k\}$. The symbol that exhibits the lowest PAPR is then selected for transmission.

The different waveforms are obtained by changing the phase of $U$ copies of $\{A_k\}$. This is achieved by element-wise multiplication of $\{A_k\}$ with i.i.d. random sequences [7] $\phi_{k,u}$, $0 \leq u \leq U-1$; *i.e.* $A_{k,u} = A_k.\phi_{k,u}$. Then, time-domain symbols $x_u[n]$ are computed through $U$ IFFT with inputs $\{A_{k,u}\}$. As already mentionned, an oversampling by a factor $L$ (*e.g.*, $L = 4$ because we use IFFT) is usually implemented in order not to miss any peak that may appear in the analog domain. Finally, the symbol waveform $x_{\tilde{u}}[n]$ with the lowest PAPR is selected

7. Often, random sequences are implemented with Pseudo Random Noise sequences (PRN-sequences).

for transmission:

$$\tilde{u} = \underset{0 \le u \le U-1}{\operatorname{argmin}} \{\mathrm{PAPR}_u\}$$
$$\text{with } \mathrm{PAPR}_u = \frac{\underset{0 \le n \le L.N-1}{\max} \left\{|x_u[n]|^2\right\}}{\mathbb{E}\left\{|x_u[n]|^2\right\}}$$

To guarantee that $\mathbb{E}\left\{|x_u[n]|^2\right\} = \mathbb{E}\left\{|A_k|^2\right\}$, the PRN-sequences are chosen not only to be i.i.d but also to be drawn from the unit circle of the complex plane so that $|\phi_{k,u}| = 1$ (*e.g.* $\phi_{k,u} \in \{\pm 1\}$ or $\phi_{k,u} \in \{\pm i\}$ or $\phi_{k,u} \in \{(\pm 1 \pm i)/\sqrt{2}\}$). This is done to keep the same amplitude to the original $\{A_k\}$ after randomization.

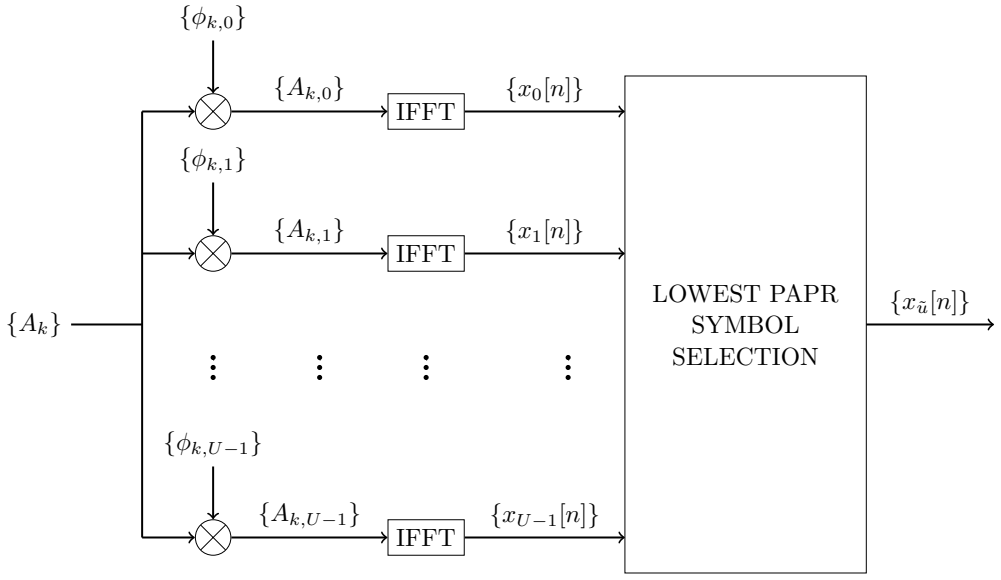



Figure 2.7 – SLM-OFDM block diagram.

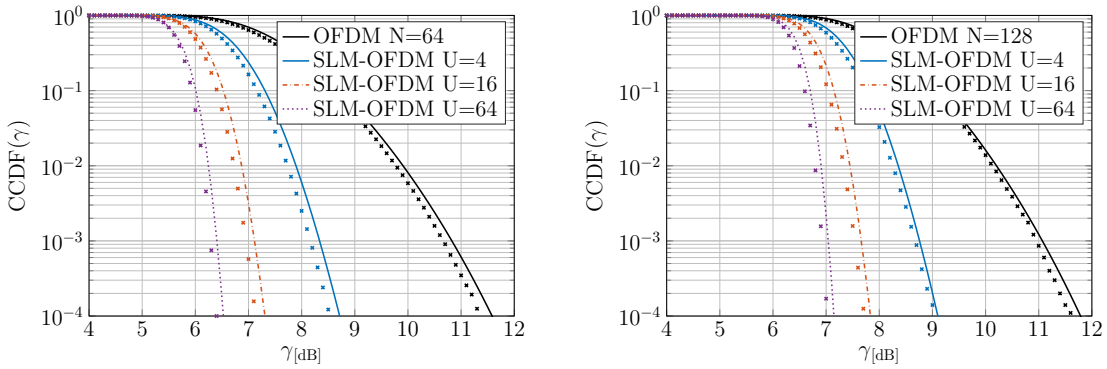
Figure 2.7 depicts a block diagram for generating a SLM-OFDM symbol as explained above ($L = 1$ for the sake of simplicity in the current basic explanation).

The resulting CCDF of SLM-OFDM at Nyquist rate (*i.e.* $L = 1$) is given by [29]:

$$\begin{aligned} \mathrm{CCDF}(\gamma) &= \Pr\{\mathrm{PAPR}_{\tilde{u}} > \gamma\} \\ &= \prod_{u=0}^{U-1} \Pr\{\mathrm{PAPR}_u > \gamma\} \\ &= (\Pr\{\mathrm{PAPR} > \gamma\})^U \\ &= \left(1 - (1 - \exp\{-\gamma\})^N\right)^U \end{aligned} \tag{2.7}$$

which was found to hold under the necessary condition $\mathbb{E}\{\phi_{k,u}\} = 0$ [40] (see independence condition on second line of Eq.(2.7)). However, as already mentioned, oversampling is needed in order not to miss any peaks that might appear in the analog domain. Extensive computer simulations have then shown that Eq. (2.8) fits the empirical approximation (as in Eq. (2.3)):

$$\mathrm{CCDF}(\gamma) \approx \left(1 - (1 - \exp\{-\gamma\})^{2.8N}\right)^U \tag{2.8}$$




(a) (b)

Figure 2.8 – CCDF($\gamma$) for $10^6$ SLM-OFDM & OFDM symbols with different number of subcarriers $N$: (a) $N = 64$ subcarriers and (b) $N = 128$. Oversampling factor $L = 4$ and QPSK subcarrier mapping.

Figure 2.8 depicts the result of computer simulations for $10^6$ SLM-OFDM symbols for each value of $U$ ($U = 4, 16$ and 64 are used). The empirical CCDF (represented by the crosses in Figure 2.8) are very close [8] to Eq.(2.8) (represented by the solid lines in Figure 2.8). It can be seen that increasing values of $U$ improve the CCDF($\gamma$).

To finish with the SLM-OFDM presentation, two points are to be mentioned:

(i) With SLM-OFDM, the PAPR improves with increasing values of $U$ at the cost of an increasing computation.

(ii) A side information of $\log_2(U)$ bits is required to inform the receiver side which random sequence has been used at the transmitter to de-randomize the received data with minimum error. Increasing the value of $U$ increases the data rate loss [9].

### 2.2.3 Single Carrier Frequency Division Multiple Access

There are several pre-coding techniques for PAPR reduction and we give in this section the example of a pre-coding technique used in modern cellular networks.

The Single Carrier Frequency Division Multiple Access or SC-FDMA is the modulation scheme deployed in the communication up-link of the 3GPP LTE mobile networks. SC-FDMA is mainly a multiple access technique in which the users share the same frequency band. Like in OFDMA (OFDM for multiple access), the available $N$ subcarriers are shared between $Z$ users (for example, each user has $M = N/Z$ subcarriers).

8. As mentioned for the CCDF results in OFDM, this is validated until $10^{-4}$. Below this probability it is not guaranteed that both CCDF match.

9. Several blind methods exist to overcome this loss.

However, to overcome the high PAPR issue of OFDMA, the data set $\{A_{k,z}\}$, $k \in \{0, \cdots, M-1\}$ of user $z \in \{0, \cdots, Z-1\}$ are spread [10] by means of a M-DFT [36]. Then multiple access resource allocation precede the N-IFFT modulator like in OFDMA.

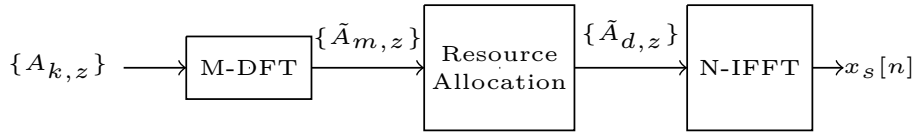



Figure 2.9 – SC-FDMA block diagram.

Figure 2.9 depicts the SC-FDMA block diagram where the spread data set is given by:

$$\tilde{A}_{m,z} = \frac{1}{\sqrt{M}} \sum_{k=0}^{M-1} A_{k,z} \exp\left\{-i2\pi \frac{km}{M}\right\}, \quad 0 \leq m \leq M-1$$

The resource allocation has two different modes:
(i) the localized mode, also referred as LFDMA and,
(ii) the interleaved mode, referred as IFDMA.

In the LFDMA mode, the available subcarriers $d \in \{0, 1, \cdots N-1\}$ are partitioned in $Z = \frac{N}{M}$ consecutive sets of subcarriers and are assigned to each user as follows:

$$\tilde{A}_{d,z} = \begin{cases} \tilde{A}_{m,z}, & d = m + Mz \\ 0, & \text{otherwise} \end{cases}$$

In the IFDMA mode the subcarriers are assigned by regularly spacing them as follows:

$$\tilde{A}_{d,z} = \begin{cases} \tilde{A}_{m,z}, & d = mZ + z \\ 0, & \text{otherwise} \end{cases}$$

The choice of the resource allocation method will depend mainly on the scenario. The LFDMA mode is preferred in most cases because with Channel Dependent Scheduling (CDS), the Base Station can exploit the frequency selectivity of the channel to assign terminals to frequency bands with good propagation conditions. However, in the particular high mobility users scenario, IFDMA is more suitable because it provides frequency diversity since the transmitted signal is spread over the entire bandwidth [42].

Finally, the time-domain SC-FDMA symbol is computed as:

$$x_s[n] = \frac{1}{N} \sum_{d=0}^{N-1} \tilde{A}_{d,z} \exp\left\{i2\pi \frac{dn}{N}\right\}, \quad 0 \leq n \leq N-1$$

The PAPR is independent of the resource allocation mode. There is no theoretical CCDF of the PAPR in the literature for SC-FDMA.

10. This operation is also referred as pre-coding.

The Carrier Interferometry OFDM is an equivalent version to SC-FDMA, where the precoding is obtained through an IDFT rather than a DFT (which seems to be the same since it will just permute the corresponding operations to be performed at the transmitter and the receiver sides: a DFT rather than an IDFT).

## 2.3 GreenOFDM: a new PAPR reduction scheme

The GreenOFDM [13, 14] is a method inspired from the PAPR reduction technique SLM-OFDM presented above. As we have seen, the key point in SLM-OFDM is to generate as much candidates ($C = U$) [11] as possible from the same data set $\{A_k\}$. The GreenOFDM technique will seek to increase the number $C$ of candidates obtained by SLM-OFDM at a fixed number ($U$) of IFFTs.

As it will become apparent shortly, the GreenOFDM technique generates $C = \frac{U^2}{4}$ OFDM symbol candidates using no more than $U$ IFFTs at the transmitter.

### 2.3.1 Generation of the GreenOFDM waveform candidates

The GreenOFDM begins, alike SLM-OFDM, by randomizing $U$ copies of the $N$ subcarriers symbols set $\{A_k\}$, with $0 \leq k \leq N-1$, through element-wise multiplication with $U$ different random phase sequences $\{\phi_{k,u}\}$ $(0 \leq u \leq U-1)$ to obtain $\{A_{k,u}\} = \{A_k.\phi_{k,u}\}$.

The randomized copies are divided into two groups:

$$\begin{aligned} \{A_{k,g_1}\} &= \{A_k.\phi_{k,g_1}\} \text{ with } 0 \leq g_1 \leq \frac{U}{2} - 1, \quad 0 \leq k \leq N-1 \\ \{A_{k,g_2}\} &= \{A_k.\phi_{k,g_2}\} \text{ with } \frac{U}{2} \leq g_2 \leq U-1, \quad 0 \leq k \leq N-1 \end{aligned} \tag{2.9}$$

Next, $L$ being the oversampling factor, $U$ $LN-$IFFTs are computed, one for each $(L-1)N$ zero-padded set $\{A_{k,g_j}\}$ $(j = 1, 2)$ to obtain the oversampled time-domain waveforms $\{x_{g_1}[n]\}$ and $\{x_{g_2}[n]\}$ with $0 \leq n \leq LN-1$.

And last, each time-domain waveform from the first group $\{x_{g_1}[n]\}$ is summed with each time-domain waveform from the second group $\{x_{g_2}[n]\}$. We introduce a factor $\sqrt{2}$ in the following definition of the GreenOFDM samples to maintain the same Euclidean distances between the $\{A_k\}$, and thus the same average power (see the next section 2.3.2).

The GreenOFDM candidate samples are thus given by:

$$x_{g_1,g_2}[n] = \frac{x_{g_1}[n] + x_{g_2}[n]}{\sqrt{2}}$$

11. $C$ is here introduced to differentiate between the $U$ computed IFFTs and the $C$ generated candidates.

### 2.3.2 Choice of the random sequences

The way the random phase sequences $\{\phi_{k,g_1}\}$ and $\{\phi_{k,g_2}\}$ are chosen is very important. A wrong choice could (i) change the average power of each candidate waveform and/or (ii) cancel data subcarriers and change the euclidean distance between the constellation points.

**Average power preservation.** To guarantee $\mathbb{E}\left\{|x_{g_1,g_2}[n]|^2\right\} = \mathbb{E}\left\{|A_k|^2\right\}$:

$$\begin{aligned}\mathbb{E}\left\{|x_{g_1,g_2}[n]|^2\right\} &= \frac{1}{2N}\sum_{k=0}^{N-1}\sum_{l=0}^{N-1}\mathbb{E}\left\{\left(A_{k,g_1}+A_{k,g_2}\right)\left(A^*_{l,g_1}+A^*_{l,g_2}\right)\right\}\exp\left\{i2\pi\frac{(k-l)n}{N}\right\}\\ &= \frac{1}{2N}\sum_{k=0}^{N-1}\mathbb{E}\left\{|A_k|^2\left(|\phi_{k,g_1}|^2+|\phi_{k,g_2}|^2+\phi^*_{k,g_1}\phi_{k,g_2}+\phi_{k,g_1}\phi^*_{k,g_2}\right)\right\}\end{aligned}$$

When the random phase sequences satisfy

$\mathbb{E}\{\phi_{k,g_1}\} = \mathbb{E}\{\phi_{k,g_2}\} = 0$ and

$\mathbb{E}\left\{|\phi_{k,g_1}|^2\right\} = \mathbb{E}\left\{|\phi_{k,g_2}|^2\right\} = 1$

which are the same conditions as in SLM-OFDM (see section 2.2.2), the average power $\mathbb{E}\left\{|x_{g_1,g_2}[n]|^2\right\} = \mathbb{E}\left\{|A_k|^2\right\}$ is preserved.

**About data subcarriers and euclidean distance preservation.** Random sequences must not be in the same set (*e.g.*, $\{\phi_{k,g_1}\},\{\phi_{k,g_2}\} \in \{\pm 1\}$). Otherwise, assuming that they were in the same set, *e.g.* $\{\phi_{k,g_1}\}$ and $\{\phi_{k,g_2}\} \in \{\pm 1\}$, it might then happen that for a certain subcarrier index $\kappa$, $0 \leq \kappa \leq N-1$: $\phi_{\kappa,g_1} = -\phi_{\kappa,g_2}$. In that case, the data at subcarrier $\kappa$ would then be lost since $A_{\kappa,g_1g_2} = A_{\kappa}.(\phi_{\kappa,g_1}+\phi_{\kappa,g_2}) = 0$.

To guarantee that data subcarriers are not lost and that the euclidean distances of the constellation points in frequency domain are not changed when generating the different OFDM waveform candidates, it must be verified that $\left|A_k.\frac{\left(\phi_{k,g_1}+\phi_{k,g_2}\right)}{\sqrt{2}}\right| = |A_k|\ \forall k$, *i.e.*:

$$\left|\frac{\phi_{k,g_1}+\phi_{k,g_2}}{\sqrt{2}}\right| = 1 \qquad \forall k$$

Changing the notation, $\phi_{k,g_1} = e^{i\theta_1}$ and $\phi_{k,g_2} = e^{i\theta_2}$ with $\theta_1$ and $\theta_2 \in [0, 2\pi)$ we have:

$$\begin{aligned}\left|\frac{e^{i\theta_1}+e^{i\theta_2}}{\sqrt{2}}\right|^2 &= (1+\cos(\theta_1-\theta_2)) = 1\\ &= \cos(\theta_1-\theta_2) = 0\\ &\iff \theta_1-\theta_2 = \frac{\pi}{2} \mod(\pi)\end{aligned}$$

For the sake of simplicity without loosing generality, if we set $\phi_{k,g_1} \in \{\pm 1\}$ ($\theta_1 = 0$ or $\pi$) hence $\phi_{k,g_2} \in \{\pm i\}$ ($\theta_2 = -\frac{\pi}{2}$ or $\frac{\pi}{2}$).

### 2.3.3 Putting all together

Figure 2.10 represents the basic building blocks of GreenOFDM transmitter where the dotted blue square contains the symbols from the 'first' group $\{A_{k,g_1}\}$ and the dotted red square contains the symbols from the 'second' group $\{A_{k,g_2}\}$. It is straightforward to see that for all the combinations of the symbols from the first group ($x_{g_1}[n]$) with the symbols from the second group ($x_{g_2}[n]$), a total of $C = \frac{U^2}{4}$ OFDM symbol waveform candidates are generated by computing $U$ IFFTs.

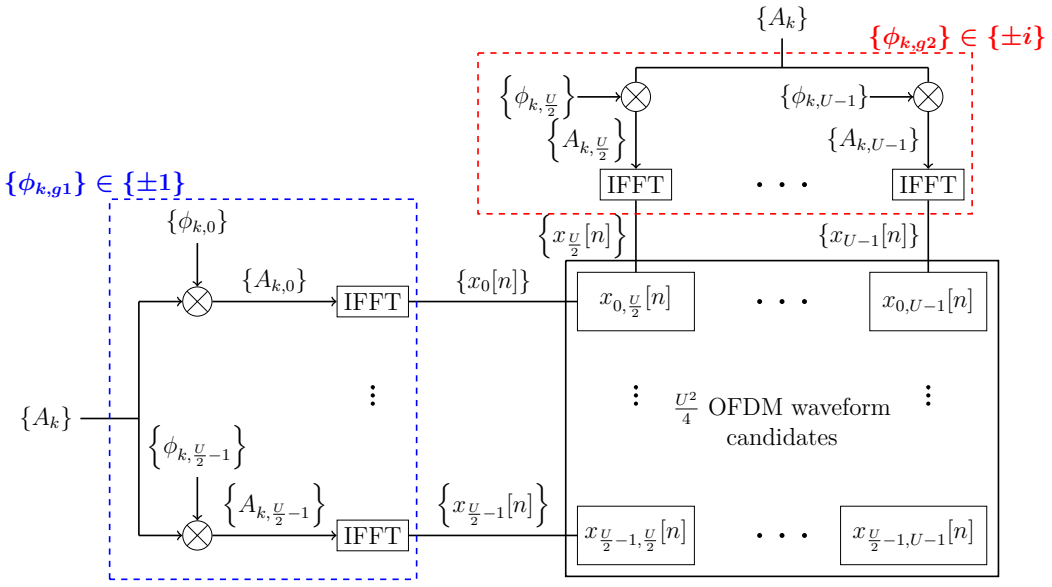



Figure 2.10 – The basic building blocks of GreenOFDM transmitter.

As in SLM-OFDM, the time-domain waveform candidate $x_{\tilde{g}_1,\tilde{g}_2}[n]$ with the lowest PAPR is selected for transmission, where:

$$\begin{aligned}(\tilde{g}_1, \tilde{g}_2) &= \underset{g_1,g_2}{\operatorname{argmin}} \{\mathrm{PAPR}_{g_1,g_2}\} \\ &= \underset{g_1,g_2}{\operatorname{argmin}} \left\{ \max_n \left\{ \frac{|x_{g_1,g_2}[n]|^2}{\mathbb{E}\{|x_{g_1,g_2}[n]|^2\}} \right\} \right\}\end{aligned} \tag{2.10}$$

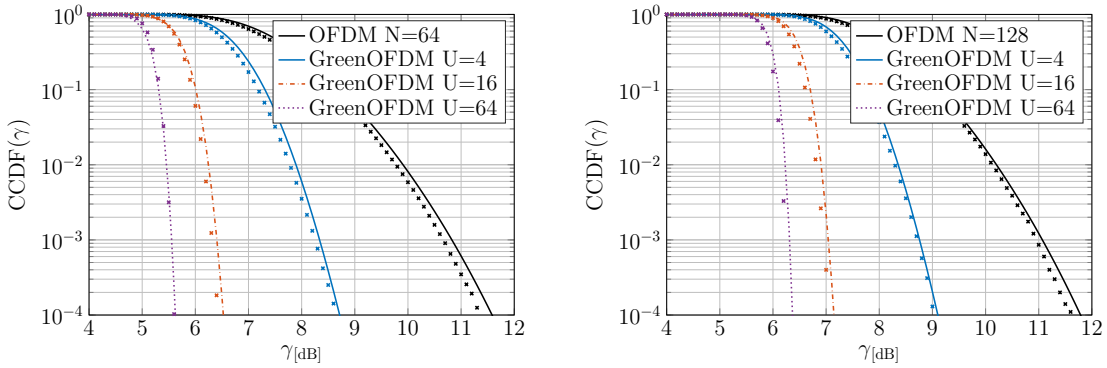
where $\mathbb{E}\{|x_{g_1,g_2}[n]|^2\} = \mathbb{E}\{|A_k|^2\}$ for $A_k$ i.i.d. random variables ($A_k \in 2^n$−QAM and $n = 2m, m \in \mathbb{N}^*$) and where $\mathrm{PAPR}_{g_1,g_2}$ corresponds to the PAPR of the symbol $x_{g_1,g_2}[n]$.

### 2.3.4 CCDF of PAPR in GreenOFDM

As for SLM (Eq.(2.8)), the CCDF of the oversampled GreenOFDM symbols is well approximated by:

$$\mathrm{CCDF}(\gamma) \approx \left(1 - (1 - \exp\{-\gamma\})^{2.8N}\right)^C \tag{2.11}$$

with $C = \frac{U^2}{4}$.

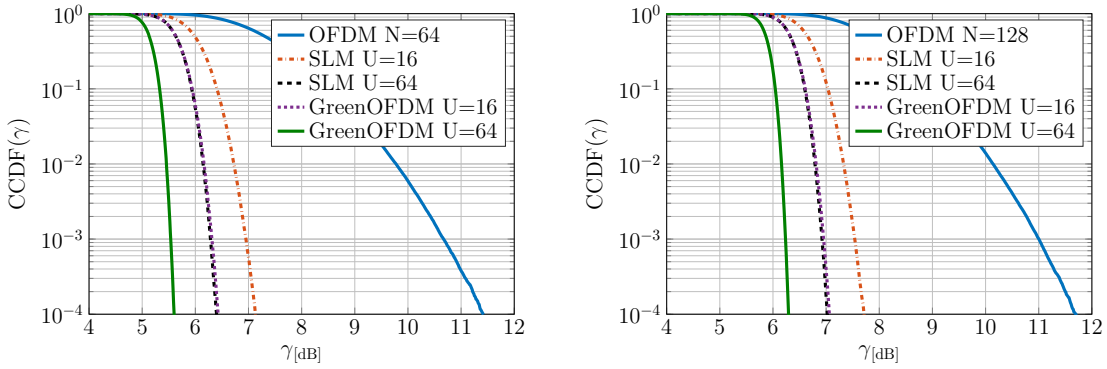



(a) (b)

Figure 2.11 – CCDF of PAPR for $10^6$ GreenOFDM & OFDM symbols with different number of subcarriers: (a) $N = 64$ subcarriers and (b) $N = 128$. Oversampling factor $L = 4$ and QPSK subcarrier mapping.

Figure 2.11 depicts the result of computer simulations for $10^6$ GreenOFDM symbols at each value of $U$ ($U = 4, 16$ and 64 are used as for the classical SLM-OFDM in section 2.2.2). The simulated CCDFs (represented by the crosses) are very close [12] to Eq.(2.11) (represented by the solid lines). As in SLM-OFDM, increasing values of $U$ improve the CCDF($\gamma$).

We now present some computer simulations to compare the CCDF of PAPR for GreenOFDM, SLM-OFDM, conventional OFDM and SC-FDMA. We carried out simulations with two different numbers of subcarriers: $N = 64$ and $N = 128$, oversampling factor $L = 4$, $QPSK$ mapping and $U = 16$ and $U = 64$; $10^6$ symbols are generated.




(a) (b)

Figure 2.12 – CCDF of PAPR for OFDM, SLM-OFDM and GreenOFDM: (a) $N = 64$ subcarriers and (b) $N = 128$, $U = 16$ and $U = 64$, $L = 4$ and QPSK subcarrier mapping.

Figure 2.12 depicts the simulation results for the CCDFs of PAPR. Better PAPR reduction

12. As previously mentioned, this is validated until $10^{-4}$. Below this probability it is not guaranteed that both CCDF match.

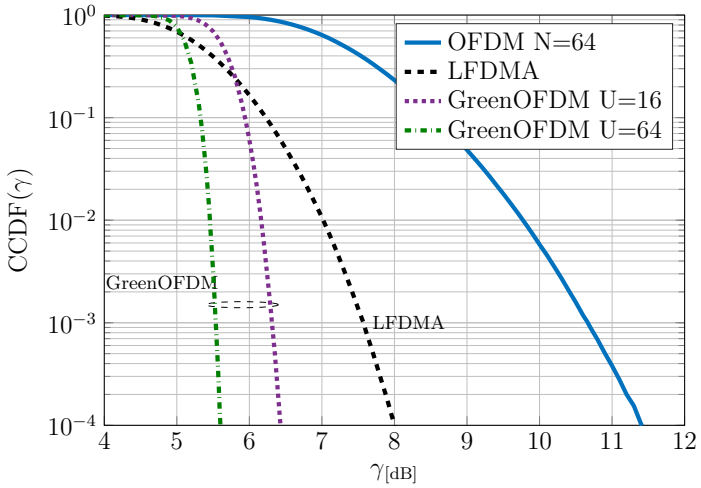



Figure 2.13 – Results of computer simulations of CCDF PAPR for OFDM, GreenOFDM and LFDMA. $N = 64$ QPSK subcarriers, $L = 4$ and $U = 16, 64$.

is obtained with GreenOFDM when compared to SLM-OFDM for the same number of $U$ computed IFFTs. On the other hand, the same PAPR reduction could be obtained by the two methods if $\frac{U^2}{4}$ IFFTs were computed for SLM-OFDM while only $U$ IFFTs were computed for GreenOFDM (this is the case of $U_{SLM} = 64$ in SLM-OFDM compared with $U_{Green} = 16$ in GreenOFDM).

We have also compared GreenOFDM with SC-FDMA with the same parameters, *i.e.* $N = 64$, $L = 4$, QPSK subcarrier mapping and $U = 16$ and $U = 64$. Results depicted in Figure 2.13 show that GreenOFDM presents better results for PAPR reduction compared with LFDMA.

The essential benefit of GreenOFDM is related to the computational burden which is proportional to $U$ so as to generate $\frac{U^2}{4}$ candidates while SLM-OFDM, with the same performance, needs $\frac{U^2}{4}$ IFFTs.

However, the drawback of GreenOFDM is related to data rate loss. Indeed, robust methods are required to send to the receiver the so-called Side Information (SI). This SI is needed to correctly de-randomize the received data. To send the SI, dedicated subcarriers are required. The more we want to reduce the PAPR the larger the SI will be (because of the required higher values of $U$) and hence more subcarriers must be dedicated to the SI.

## 2.4 Conclusion of the Chapter

In this chapter we have described the PAPR of OFDM baseband symbols and analyzed its distribution (CCDF).

Then, we presented the PAPR of the I and Q components, required to dimension the

DACs. We saw that, to avoid any clipping, the DACs have to be dimensioned considering that the PAPR of the I and Q components is twice the PAPR in baseband.

The same analysis for the RF symbols led us to conclude that the PAPR in RF is twice the PAPR in baseband.

Afterwards, we presented some of the existing PAPR reduction techniques:

(i) the Clipping and Filtering consisting in getting rid of the time-domain symbol peaks that are higher than a predefined threshold,

(ii) the SLM-OFDM consisting in generating several time-domain versions of the data set $\{A_k\}$, each version exhibiting a different PAPR and the best one being finally chosen for transmission,

(iii) the SC-FDMA consisting in spreading the data prior to the IFFT by means of a DFT over the data set $\{A_k\}$.

Finally we introduced the GreenOFDM that is inspired by the SLM-OFDM. We described the method and we obtained the resulting PAPR distribution. We saw that the CCDF of PAPR for GreenOFDM outperforms the PAPR reduction performance of both SLM-OFDM and SC-FDMA. The reason why the GreenOFDM outperforms the PAPR reduction capabilities of SLM-OFDM at a given number $U$ of IFFTs is due to a higher number of available candidates $C$. For SLM-OFDM, $U$ IFFTs generate $C = U$ candidates whereas for GreenOFDM $U$ IFFTs generate $C = \frac{U^2}{4}$ candidates.

Chapter 3

# Implementation and energy consumption

## Contents



With the new proposed GreenOFDM scheme, questions have arisen regarding (i) the computation burden of the transmitter side and (ii) the overall energy consumption. In this chapter, these two essential questions are answered.

Firstly, the transmitter computation burden is reduced by implementing a method [43] based on (i) the fact that, for GreenOFDM and SLM, the CCDF(PAPR) decreases rapidly as a function of the number of candidates $C$, (ii) The particular structure of the GreenOFDM modulation that implies that, for each additional IFFT, more candidates are generated than for SLM.

Secondly, the total number of computation operations is calculated for the proposed GreenOFDM scheme.

Then, the energy consumption of the digital part of the implementation is compared with the High Power Amplifier (HPA) energy consumption.

The overall energy consumption of GreenOFDM is compared with the LoRa Orthogonal Signaling scheme that is by itself energy efficient.

## 3.1 GreenOFDM, computational complexity reduction

It was already shown through computer simulations that the GreenOFDM needs much less computation operations than SLM-OFDM for the same amount of PAPR reduction. Moreover, this large reduction of GreenOFDM computational charge as compared with SLM-OFDM can further be improved, as explained in what follows.

A brute force implementation of GreenOFDM requires to calculate all the $\frac{U^2}{4}$ OFDM waveform candidates and their corresponding PAPR in order to select the one with the lowest PAPR.

Finding the lowest PAPR among all the OFDM waveform candidates forbids to find a method with substantial computational charge reduction.

**The key idea to reduce the complexity** The CCDF($\gamma$) for the GreenOFDM (Figure 3.1 plots an example) is obtained by finding the symbol with the lowest PAPR among the whole set of $C = U^2/4$ candidates. We have seen in the previous chapter that CCDF($\gamma$) is well approximated by Eq.(2.11) for $10^{-4} < \text{CCDF}(\gamma) < 1$.

For a given value of $p_0$, generally chosen between $10^{-3}$ and $10^{-4}$, we obtain $\gamma_{p_0}$ as a function of $C$ (associated with $U$). This value $\gamma_{p_0}^C$ is used to dimension the system circuitry [1] (the HPA and the DACs).

Since Eq.(2.11) was validated through computer simulations (down to $\text{CCDF}(\gamma) = 10^{-4}$), then it can be used to determine for a given $p_0$:

$$p_0 = \text{CCDF}(\gamma_{p_0}) \approx \left(1 - \left(1 - \mathrm{e}^{-\gamma_{p_0}}\right)^{2.8N}\right)^C \tag{3.1}$$

1. Recall paragraph "**Maximum PAPR with a given probability**" in Chapter 2.

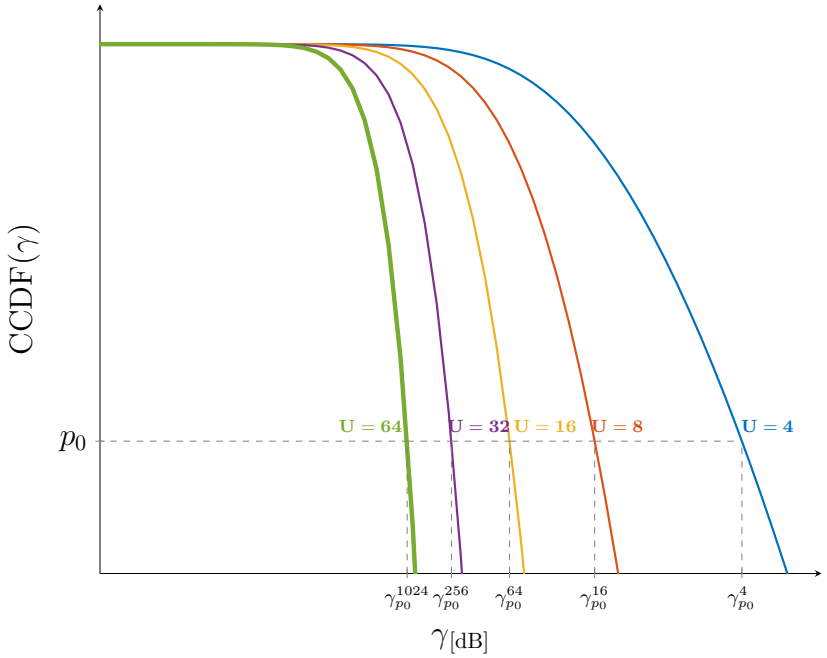



Figure 3.1 – CCDF($\gamma$) for GreenOFDM with a given number of subcarriers $N = 64$ and a given set of values of $U = 4, 8, 16, 32$ and 64.

the value of $\gamma_{p_0}$ [2]:

$$\gamma_{p_0} \approx -\ln\left(1 - \left(1 - \exp\left\{\frac{\ln(p_0)}{C}\right\}\right)^{\frac{1}{2.8N}}\right) \tag{3.2}$$

where $C = \frac{U^2}{4}$ and, as mentioned before, $p_0$ is chosen to be equal to $10^{-3}$ or $10^{-4}$.

An important remark at this point is that all symbol candidates whose PAPR are lower than $\gamma_{p_0}$ are worth for transmission, there is no need to find the GreenOFDM candidate symbol with the lowest PAPR, we only need to find one candidate whose PAPR is lower than $\gamma_{p_0}$. In the following section, we present a method to reduce the computational charge based on this remark.

2. Using simple arithmetic we have:

$$\ln(p_0) \approx C \times \ln\left(1 - \left(1 - \mathrm{e}^{-\gamma_{p_0}}\right)^{2.8N}\right) \rightarrow \gamma_{p_0} \approx -\ln\left(1 - \exp\left\{\frac{\ln\left(1 - \exp\left\{\frac{\ln(p_0)}{C}\right\}\right)}{2.8N}\right\}\right)$$

$$\gamma_{p_0} \approx -\ln\left(1 - \left(1 - \exp\left\{\frac{\ln(p_0)}{C}\right\}\right)^{\frac{1}{2.8N}}\right)$$

### 3.1.1 The GreenOFDM with IFFTs-on-demand

In the proposed method [43], the IFFTs are progressively computed to generate the candidates one by one in a specific way that will become clear shortly. For each generated candidate, the $\text{PAPR}_{g_1,g_2}$ (*cf.* Eq.(2.10)) is computed and compared with the predefined threshold $\gamma_{p_0}$. Once a candidate waveform with acceptable PAPR is found (*i.e.*, $\text{PAPR}_{g_1,g_2} \leq \gamma_{p_0}$), then it is selected for transmission and there is no need to comptute other IFFTs.

#### 3.1.1.1 Principle of GreenOFDM with IFFTs-on-demand

Let us see in details the way the IFFTs are progressively computed.

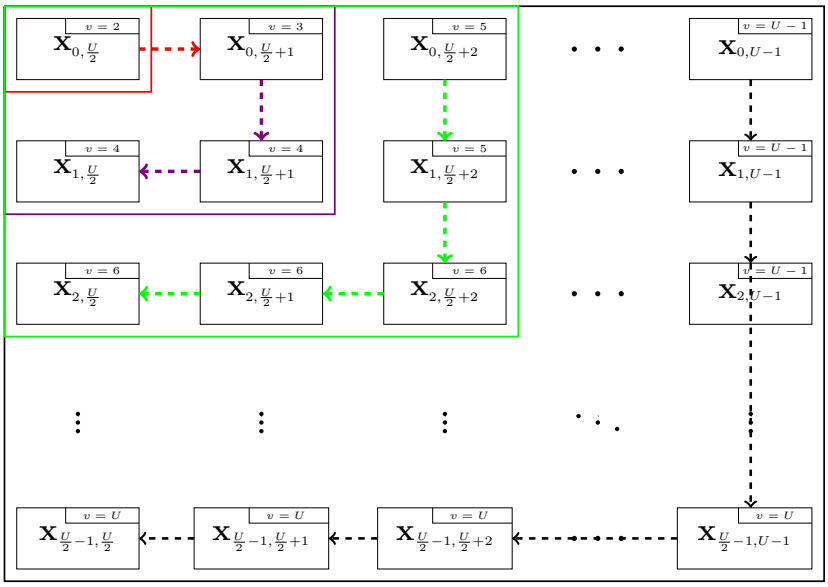



Figure 3.2 – Block diagram description of progressive IFFTs.

**The first step of the method.** The GreenOFDM IFFTs-on-demand begins by computing $v = 2$ IFFTs to obtain the candidate $\{x_{0,\frac{U}{2}}[n]\}$ depicted in Figure 2.10 (see Chapter 2) as well as in vector notation $\mathbf{x}_{0,\frac{U}{2}}$ in Figure 3.2.

The PAPR of this candidate is evaluated and, if $\text{PAPR}_{0,\frac{U}{2}} \leq \gamma_{p_0}$ then the symbol is selected for transmission and the process stops here after the computation of 2 IFFTs. Otherwise the process goes on.

**The second step of the method.** Since $\text{PAPR}_{0,\frac{U}{2}} > \gamma_{p_0}$, a third IFFT is computed (red arrow in Figure 3.2, $v = 3$) to obtain the candidate $\mathbf{x}_{0,\frac{U}{2}+1}$.

The PAPR of this candidate is evaluated and, if $\text{PAPR}_{0,\frac{U}{2}+1} \leq \gamma_{p_0}$ then the symbol is

chosen for transmission and the process stops here after the computation of 3 IFFTs. Otherwise the same process goes on.

Since $\text{PAPR}_{0,\frac{U}{2}+1} > \gamma_{p_0}$, a fourth IFFT is computed (purple arrows in Figure 3.2, $v = 4$) allowing us to generate two additional candidates $\mathbf{x}_{1,\frac{U}{2}+1}$ and $\mathbf{x}_{1,\frac{U}{2}}$. So far we get $C_v = 4$ candidates (within the purple square in Figure3.2) having computed $v = 4$ IFFTs.

At this point, the complexity is similar to that of SLM-OFDM: 4 IFFTs to generate 4 candidates.

If at least one of the previously generated candidates satisfies the condition PAPR$\leq \gamma_{p_0}$ then the process stops here after the computation of 4 IFFTs. Otherwise the same process goes on.

**The next steps of the method.** From now on, the benefit of GreenOFDM computational charge reduction starts.

Indeed, since none of the 4 previously generated candidates satisfies the condition PAPR$\leq \gamma_{p_0}$, one additional IFFT ($v = 5$) is computed to generate 2 additional candidates $\mathbf{x}_{0,\frac{U}{2}+2}$ and $\mathbf{x}_{1,\frac{U}{2}+2}$.

The main difference with respect to SLM-OFDM appears: in SLM-OFDM, one additional IFFT generates only one additional candidate while, in GreenOFDM, one additional IFFT generates several additional candidates.

Further, a sixth IFFT might be computed to generate 3 additional candidates, $\mathbf{x}_{2,\frac{U}{2}+2}$, $\mathbf{x}_{2,\frac{U}{2}+1}$ and $\mathbf{x}_{2,\frac{U}{2}}$. At this point, 6 IFFTs are computed to generate $C_v = 9$ candidates. The process should now be clear, and it goes on in the same way until we find one candidate symbol with PAPR under the threshold.

According to this IFFTs-on-demand scheme, the number of generated candidates is $C_v = \frac{v^2}{4}$ for $v$ even and $C_v = \frac{v^2-1}{4}$ for $v$ odd; that provides $v \propto 2\sqrt{C_v}$ computed IFFTs.

#### 3.1.1.2 Performance of GreenOFDM with IFFTs-on-demand

The number $v$ of computed IFFTs is a random variable since it is directly related to the PAPR and the chosen $\gamma_{p_0[dB]}$. Assuming that the candidates are independent to each other, we can derive the probability mass function of the random variable $v$ as detailed in Appendix A.

$\Pr\{V = v\}$ is given by:

$$\Pr\{V = v\} = \begin{cases} \left(1 - e^{-\gamma_{p_0}}\right)^{2.8N} \cdot \sum_{c=0}^{\frac{v}{2}-1} \left(1 - \left(1 - e^{-\gamma_{p_0}}\right)^{2.8N}\right)^{\frac{v}{2}\left(\frac{v}{2}-1\right)+c}, & \text{if } v \text{ even} \\ \left(1 - e^{-\gamma_{p_0}}\right)^{2.8N} \cdot \sum_{c=0}^{\frac{v-1}{2}-1} \left(1 - \left(1 - e^{-\gamma_{p_0}}\right)^{2.8N}\right)^{\frac{(v-1)^2}{4}+c}, & \text{if } v \text{ odd} \end{cases} \tag{3.3}$$

Computer simulation were carried out to validate the obtained result in Eq.(3.3). Figure 3.3 depicts $\Pr\{V = v\}$ using the parameters listed in Table 3.1.

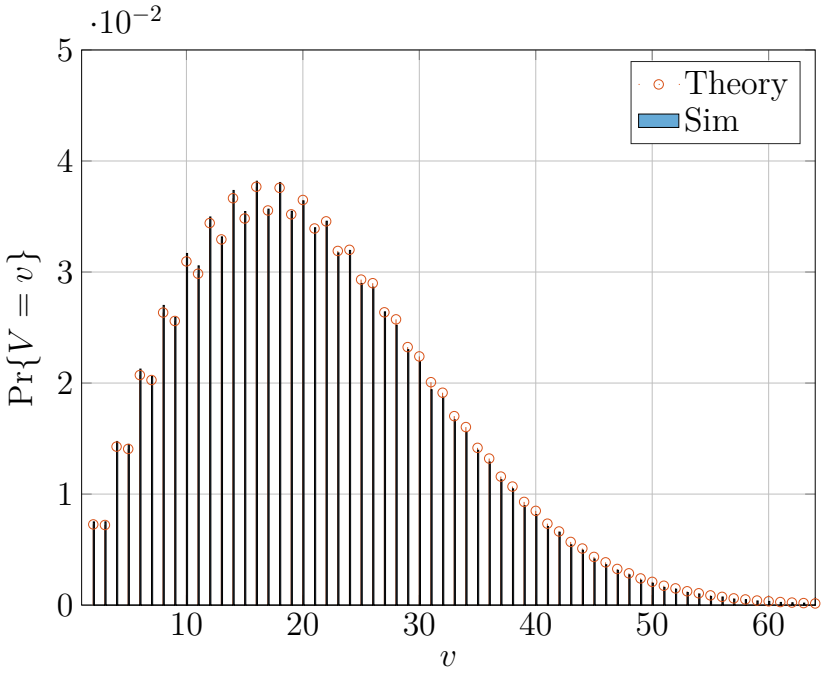



Figure 3.3 – Theoretical and simulation results of the probability mass function of the number of computed IFFTs $\Pr\{V = v\}$.

| Parameter | Value |
|---|---|
| $N$ | 64 |
| $L$ | 4 |
| $U$ | 64 |
| $p_0 = \text{CCDF}(\gamma_{p_0})$ | $10^{-3}$ |
| $\gamma_{p_0[\text{dB}]}$ | 5.55dB (*cf.* Figure2.12a) |
| Total GreenOFDM symbols | $10^6$ |

Table 3.1 – Simulation parameters for complexity reduction of GreenOFDM.

Despite the fact that the assumption of independence between the candidates is not very realistic, it can be seen that the theoretical expression in Eq.(3.3) matches the empirical histogram.

#### 3.1.1.3 Complexity reduction based on a fine tuning of $\gamma_p$

It is noticeable that the probability to be close to $U = 64$ is very low; this is quite interesting from an implementation point of view, since it permits to make it possible to reduce the number of IFFTs to be calculated by setting a maximum number $v_{\max}$ below which $V$ remains with a

high probability $1 - \Pr\{V > v\}$. The CCDF($v$) associated to Figure 3.3 is depicted in Figure 3.4.

These results show that at CCDF($v$) = $10^{-3}$ (*i.e.* with a probability of 0.999), no more than 61 IFFTs have to be computed, and at CCDF($v$) = $10^{-2}$ (*i.e.* with a probability of 0.99), this number reduces to 49. The limit being $U = 64$, we obtain that rarely we have to compute a number of IFFTs close to $U$.

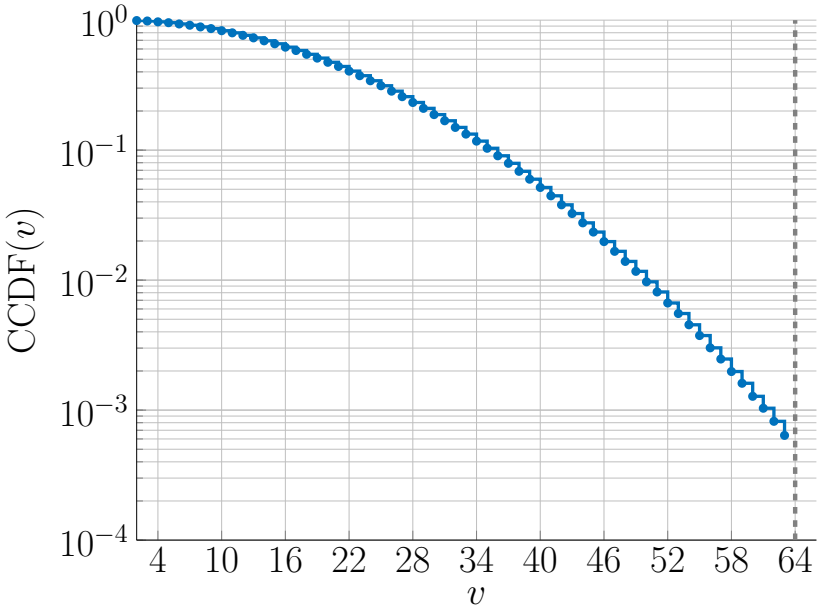



Figure 3.4 – CCDF of the required number ($v$) of IFFTs from Figure 3.3 and for $p_0 = 10^{-3}$.

The idea now is to determine what would have happened if we had kept the same theoretical ideal $p_0$ goal but stopped the IFFT calculus by using a different threshold who is slightly higher than $\gamma_{p_0}$, *i.e.* we tolerate slightly higher PAPR values. We introduce the threshold $\gamma_{p_i}$ with a tolerance margin as follows:

$$\gamma_{p_i[\text{dB}]} = \gamma_{p_0[\text{dB}]} + 0.1 \times i[\text{dB}] \text{ with } i = 1, 2, \cdots$$

Applying the same process in order to obtain the CCDF($v$) for different values of $\gamma_{p_i}$ through computer simulations gives as a result the distributions depicted in Figure 3.5. Here it can be seen that for a CCDF($v$) = $10^{-3}$ (*i.e.* 99.9% of the simulated cases) a maximum of 49 IFFTs have to be computed for $\gamma_{p_1}$, a maximum of 40 IFFTs have to be computed for $\gamma_{p_2}$, a maximum of 33 IFFTs for $\gamma_{p_3}$, a maximum of 27 IFFTs for $\gamma_{p_4}$ and a maximum of 23 IFFTs for $\gamma_{p_5}$.

The maximum number of IFFTs, for a given probability $1 - \text{CCDF}(v)$, reduces highly when $\gamma_{p_i}$ increases. At this point, two situations are presented regarding the analog circuitry:

(i) We can still perform the analog dimensioning with the threshold $\gamma_{p_0}$ at the same time that the digital complexity is being reduced with the threshold $\gamma_{p_i}$. The obtained GreenOFDM

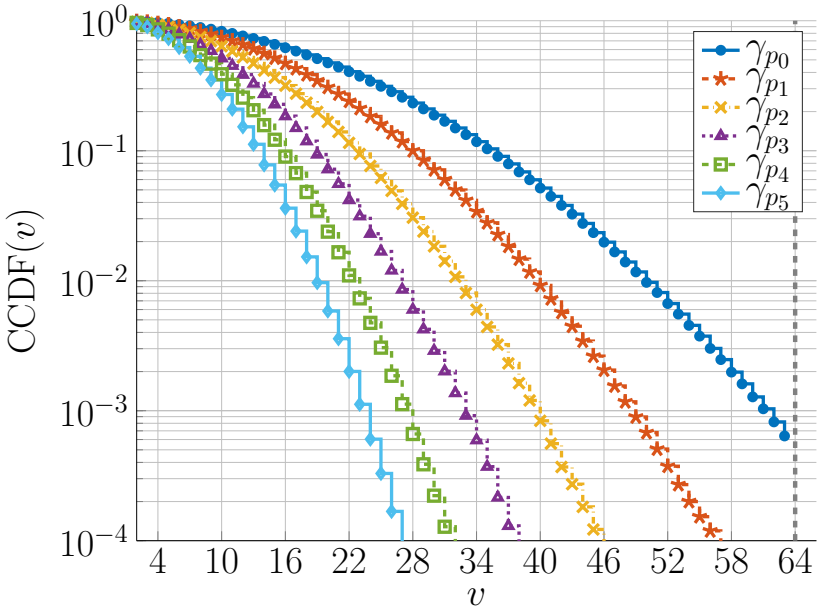



Figure 3.5 – CCDF of the required number ($v$) of IFFTs with different thresholds $\gamma_{p_i[\text{dB}]}$.

symbols in this case are sometimes going to be slightly clipped ($\gamma_{p_i} > \gamma_{p_0}$) inducing distortion. If we choose this situation, distortion is to be tolerated.

(ii) We dimension the analog with the threshold $\gamma_{p_i}$ at the same time that the digital complexity is being reduced with the same threshold. The High Power Amplifier energy consumption in this case is increased ($\gamma_{p_i} > \gamma_{p_0}$). If we choose this situation, a slightly higher energy consumption is to be tolerated.

**The resulting CCDF($\gamma$)?** Imposing a threshold and finding the first symbol with PAPR below the threshold changes the CCDF($\gamma$), as compared with the GreenOFDM without any computational charge reduction. Figure 3.6 depicts the corresponding CCDF($\gamma$) with the same parameters from Table 3.1 and with thresholds $\gamma_{p_0}$, $\gamma_{p_2}$ and $\gamma_{p_4}$. For the "threshold" GreenOFDM, the CCDF($\gamma$) stops at $\gamma_{p_i}$ since at each simulated symbol, a candidate symbol with PAPR lower or equal to $\gamma_{p_i}$ exists.

From the obtained results, we can see that the thresholded GreenOFDM symbols that are worth for transmission are guaranteed with a high probability [3] to be below $\gamma_{p_i}$. This CCDF($\gamma$) is very important to be considered regarding the circuitry dimensioning.

**Complexity reduction factor: GreenOFDM and SLM.** It is this result that has a significant impact on reducing the complexity compared to the conventional GreenOFDM in [14]. We define the Computational Complexity Reduction Factor (CRF) in terms of the

3. 1−CCDF($\gamma$)

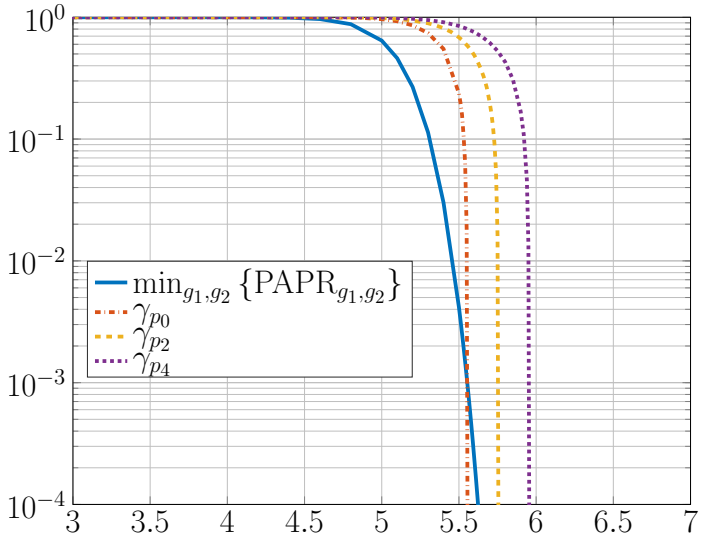



Figure 3.6 – CCDF($\gamma$) of the GreenOFDM with brute force scheme compared with implemented threshold $\gamma_{p_i}$.

number of computed IFFTs as:

$$\mathrm{CRF}_v = \frac{\mathrm{N}^{\mathrm{bre}}\ \mathrm{IFFT}_{\mathrm{after}}}{\mathrm{N}^{\mathrm{bre}}\ \mathrm{IFFT}_{\mathrm{before}}} \times 100(\%) \tag{3.4}$$

where $\mathrm{IFFT}_{\mathrm{before}} = U$ (Conventional GreenOFDM) and $\mathrm{IFFT}_{\mathrm{after}}$ is the number of IFFT at $\mathrm{CCDF}(v) = 10^{-3}$.

| **Threshold** | $\boldsymbol{v_0^{\mathrm{Green}}}$ | $\boldsymbol{U}$ | $\mathbf{CRF}_v$ |
|---|---|---|---|
| $\gamma_{p_2}$ | 40 | 64 | 62.5% |
| $\gamma_{p_3}$ | 33 | | 51.6% |
| $\gamma_{p_4}$ | 27 | | 42.2% |
| $\gamma_{p_5}$ | 23 | | 35.9% |

Table 3.2 – $\mathrm{CRF}_v$ between GreenOFDM with the proposed method and conventional GreenOFDM.

The $\mathrm{CRF}_v$ between GreenOFDM with the proposed method and the conventional GreenOFDM listed in Table 3.2 shows that the CRF decreases as $\gamma_{p_i}$ increases. This is mainly because we are relaxing the constraint (in other words, OFDM candidates with slightly higher PAPR are allowed) and hence less IFFTs have to be computed to find a candidate worth for transmission.

Furthermore, as a fair comparison, the SLM-OFDM as defined in [28] is also implemented to obtain the same PAPR reduction (*i.e.*, using the same thresholds $\gamma_{p_i}$). However, for the SLM-OFDM technique, the number of IFFTs to compute is proportional to the number of candidates (*i.e.*, $v = C_v$). It is found through computer simulations that for the SLM-OFDM to cover 99.9% of the simulated cases (*i.e.*, for $\mathrm{CCDF}(v) = 10^{-3}$) a maximum of 376 IFFTs

have to be computed for $\gamma_{p_2}$, a maximum of 255 IFFTs for $\gamma_{p_3}$, a maximum of 179 IFFTs for $\gamma_{p_4}$ and a maximum of 128 IFFTs for $\gamma_{p_5}$. Results are depicted in Figure 3.7 in logarithmic scale together with the results of the GreenOFDM.

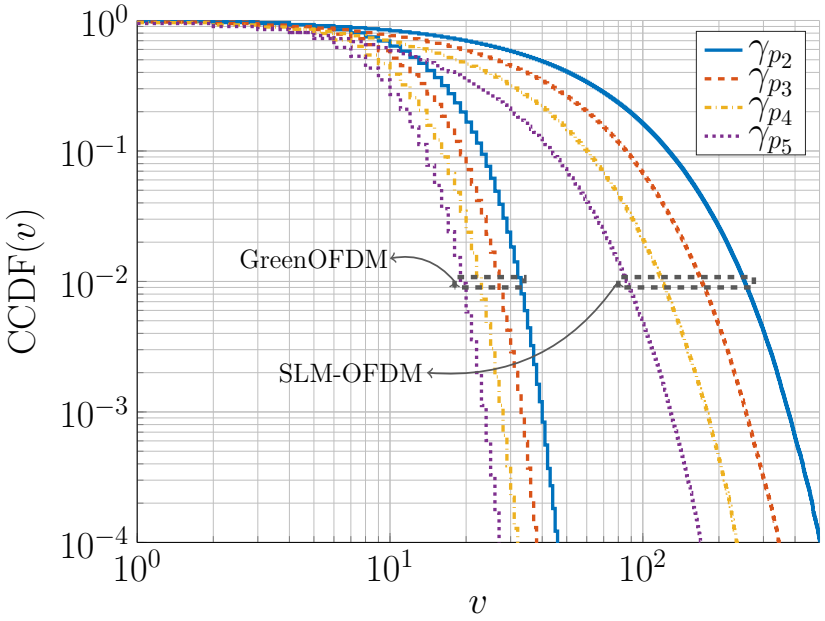



Figure 3.7 – CCDF of the required number $v$ of IFFTs with different thresholds $\gamma_{p_i}$[dB] for both GreenOFDM and SLM-OFDM.

The $\text{CRF}_v$ between GreenOFDM with the proposed method and the SLM-OFDM with the implemented threshold are listed in Table 3.3. It can be observed that the $\text{CRF}_v$ decreases as $\gamma_{p_i}$ decreases. This is because for lower values of $\gamma_{p_i}$, $C_v$ must increase and with SLM-OFDM the required number of IFFTs is proportional to $C_v$ while with GreenOFDM it is proportional to $2\sqrt{C_v}$.

| **Threshold** | $\boldsymbol{v_0^{\text{Green}}}$ | $\boldsymbol{v_0^{\text{SLM}}}$ | $\textbf{CRF}_v$ |
|---|---|---|---|
| $\gamma_{p_2}$ | 40 | 376 | 10.6% |
| $\gamma_{p_3}$ | 33 | 255 | 12.9% |
| $\gamma_{p_4}$ | 27 | 179 | 15.1% |
| $\gamma_{p_5}$ | 23 | 128 | 17.9% |

Table 3.3 – $\text{CRF}_v$ between GreenOFDM with the proposed method and threshold SLM-OFDM.

### 3.1.2 Hierarchical sampling

Our aim is to determine as fastly as possible if a candidate satisfy the PAPR constrainst.

It is noteworthy that:

(i) The method introduced in the current section can be used independently or as a complement to the IFFTs-on-demand method.

(ii) In this chapter, an oversampling factor $L = 4$ is used in order not to miss-out any peak that might appear in the analog domain.

The symbol candidate $x_{g_1,g_2}[n], n = 0, \cdots, 4N-1$ is sample-by-sample generated. For each sample, $\text{PAPR}_{g_1,g_2}[n] = \frac{|x_{g_1,g_2}[n]|^2}{\mathbb{E}\{|x_{g_1,g_2}[n]|^2\}}$ is computed and compared with the threshold $\gamma_{p_0}$ (or $\gamma_{p_i}$).

If there is a sample $n$ such that $\text{PAPR}_{g_1,g_2}[n] > \gamma_{p_0}$ (or $\text{PAPR}_{g_1,g_2}[n] > \gamma_{p_i}$), then the OFDM candidate symbol is discarded and the process continues with the computation of the next candidate; that way, we avoid to compute all the samples of a symbol that is going to be discarded anyway.

The method proposed [44] here is a particular order in which the samples $\{x_{g_1,g_2}[n]\}$ are scanned.

As a matter of fact, if a candidate waveform sampled at the Nyquist rate ($L = 1$) has at least one sample with a $\text{PAPR}_{g_1,g_2}[n]$ exceeding the threshold, then the oversampled version ($L = 2$ or $L = 4$) will also have one (or more) sample(s) with $\text{PAPR}_{g_1,g_2}[n]$ exceeding the threshold.

Hence, in hierarchical sampling each OFDM waveform candidate $\{x_{g_1,g_2}[n]\}$ is analyzed sample-by-sample as follows:

(i) First, $x_{g_1,g_2}[4m]$ with $m = 0, \cdots, N-1$ is computed on a sample-by-sample basis as its $|x_{g_1,g_2}[4m]|^2$ is compared with $\gamma_{p_i}$ and if $\gamma_{p_i}$ is exceeded at a certain sample, then the symbol is discarded and the next symbol is computed. However if for all $m$ $|x_{g_1,g_2}[4m]|^2$ is below $\gamma_{p_i}$ then the method continues with the samples from (ii).

(ii) The samples $x_{g_1,g_2}[4m+2]$ with $m = 0, 1, \cdots, N-1$ are computed step-by-step and for each value of $m$, $|x_{g_1,g_2}[4m+2]|^2$ is compared with $\gamma_{p_i}$. As before, if $\gamma_{p_i}$ is exceeded at a certain sample, then the symbol is discarded and the next symbol is computed. On the other hand if for all $m$ $|x_{g_1,g_2}[4m+2]|^2$ is below $\gamma_{p_i}$ then the method continues with the samples from (iii).

(iii) Finally, the samples $x_{g_1,g_2}[2m+1]$ with $m = 0, 1, \cdots, 2N-1$ are computed step-by-step. Also, $|x_{g_1,g_2}[2m+1]|^2$ is computed and compared with $\gamma_{p_i}$ and the very first found sample exceeding $\gamma_{p_i}$ causes the symbol to be discarded and we jump to the next symbol. But if for all samples $|x_{g_1,g_2}[2m+1]|^2$ is below $\gamma_{p_i}$, then the symbol is chosen for transmission.

The CCDF of the amount of processed samples $\varrho$ was obtained by computer simulations with the same parameters from Table 3.1 and with different values of $\gamma_{p_i}$. Results are depicted in Figure 3.8.

As can be seen, for 99.9% of cases (*i.e.*, for $\text{CCDF}(\varrho) = 10^{-3}$): with $\gamma_{p_2}$ a maximum of

14 840 samples were processed, with $\gamma_{p_3}$ a maximum of 11 136 samples were processed, with $\gamma_{p_4}$ a maximum of 8 320 samples were processed and with $\gamma_{p_5}$ a maximum of 6 528 samples were processed. Without this proposed computational reduction algorithm a total of $C \times LN = \frac{U^2}{4} \times 4 \times N = 262\,144$ samples would have been processed.

**Applying the method to SLM-OFDM.** This method is, as mentioned before, not specific to GreenOFDM. Indeed, applying it to SLM-OFDM with the same performances (*i.e.*, same thresholds $\gamma_{p_i}$[dB]) have shown equivalent results. Computer simulations were carried out obtaining that for 99.9% of the simulated cases a maximum of 14 660 samples were processed for $\gamma_{p_2}$, a maximum of 10 796 samples were processed for $\gamma_{p_3}$, a maximum of 8 180 samples were processed for $\gamma_{p_4}$ and a maximum of 6 375 samples were processed for $\gamma_{p_5}$.

**Comparing the hierachical and the sequential sampling.** A remark at this point is what this method adds as compared with scanning the samples $n$ successively from $n = 0$ to $n = 4N - 1$. To show the difference we obtained through computer simulation the CCDF($\varrho$) for this successive scanning of the samples $n$. The results depicted in Figure 3.9 show that at CCDF($\varrho$) $= 10^{-3}$ $\varrho$ is equal to 28 768 samples for $\gamma_{p_2}$, 20 736 samples for $\gamma_{p_3}$, 15 392 samples for $\gamma_{p_4}$ and 12 128 samples for $\gamma_{p_5}$, respectively.

Comparing with the previous results depicted in Figure 3.8 we can remark that a factor very close to 2 is obtained in the reduction of the number of processed samples. This result shows why the hierarchical sampling must be applied rather than scanning the samples successively. The reason behind this is that the samples scanned successively are highly correlated while in hierarchical sampling we begin by analysing successively the (uncorrelated) samples at the Nyquist rate.

### 3.1.3 Total number of operations for GreenOFDM and SLM

In order to implement the obtained results with a digital signal processor, an analysis of the total number of operations is required. In what follows, we present this analysis for both the GreenOFDM and the SLM. In the previous sections, we saw that better results were obtained when using the threshold $\gamma_{p_i}$ with $i > 0$. In the presented results we use the same thresholds.

#### 3.1.3.1 GreenOFDM

For a digital implementation of the GreenOFDM with the proposed algorithm, the total number of operations must be considered. Combining IFFTs-on-demand and hierarchical sampling reduces significantly the total number of operations. This number includes $\frac{LN}{2}\log_2(LN)$ complex multiplications and $LN\log_2(LN)$ complex additions for each IFFT (radix-2 FFT algorithm), a complex addition to generate each sample $x_{g_1,g_2}[n]$, two real multiplications and a

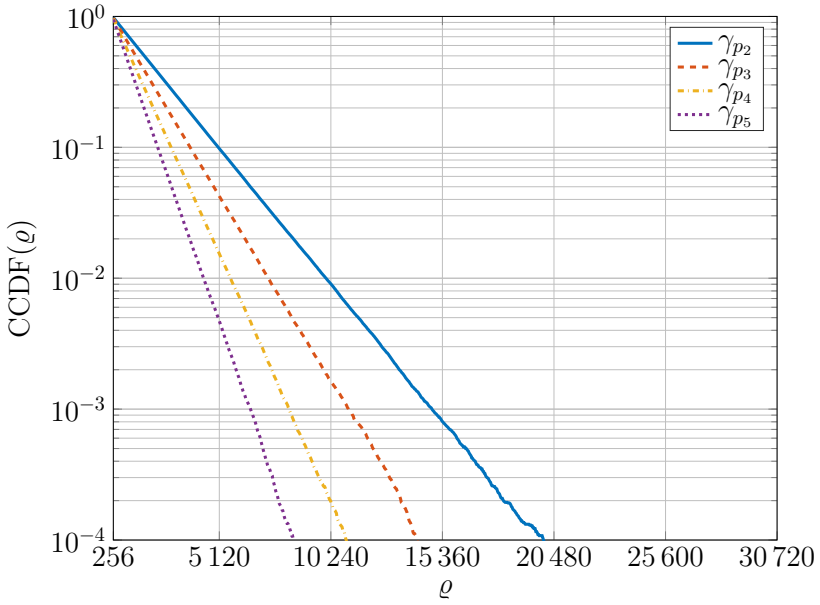



Figure 3.8 – CCDF of the amount of processed samples $\varrho$ with different thresholds $\gamma_{p_i}$ using the proposed method hierarchical sampling.

real addition to obtain $|x_{g_1,g_2}[n]|^2$ of each sample and one modulus comparison with $\gamma_{p_i}$ (note that here the normalization factor $\sqrt{2}$ and $\mathbb{E}\{|x_{g_1,g_2}[n]|^2\}$ required to obtain the $\text{PAPR}_{g_1,g_2}[n]$ can be bypassed when included into the threshold).

Decomposing the operations, we have that each complex addition is equivalent to two real additions, and each complex multiplication is equivalent to four real multiplications and two real additions.

The total amount of operations $\mathcal{M}$ required to find the adequate GreenOFDM symbol is then the sum of:

(i) The operations to compute the $v$ required IFFTs: $v\times\big(6\times\frac{LN}{2}\log_2(LN)+2\times LN\log_2(LN)\big)$.

(ii) The operations to compute the $\varrho$ processed samples: $6\varrho$.

Hence, $\mathcal{M}=v\times 5\times LN\log_2(LN)+6\varrho$.

The CCDF of the number of required operations $\mathcal{M}$ was obtained by computer simulations with the parameters of Table 3.1 and $\gamma_{p_i}$ with $i=2,3,4$ and 5, respectively.

From the obtained results (*cf.* Figure 3.10) it is found that 99.9% of the simulated cases (*i.e.* for $\text{CCDF}(\mathcal{M})=10^{-3}$) were covered by a maximum of 498 112 operations for $\gamma_{p_2}$dB, 407 296 operations for $\gamma_{p_3}$dB, 332 992 operations for $\gamma_{p_4}$dB, and 280 704 operations for $\gamma_{p_5}$dB. It is noteworthy that the total number of operations without these two proposed methods the GreenOFDM would require $U\times 5\times LN\log_2(LN)+6\times\frac{U^2}{4}\times LN\cong 2.2\ MOPS$ (Mega-Operations).

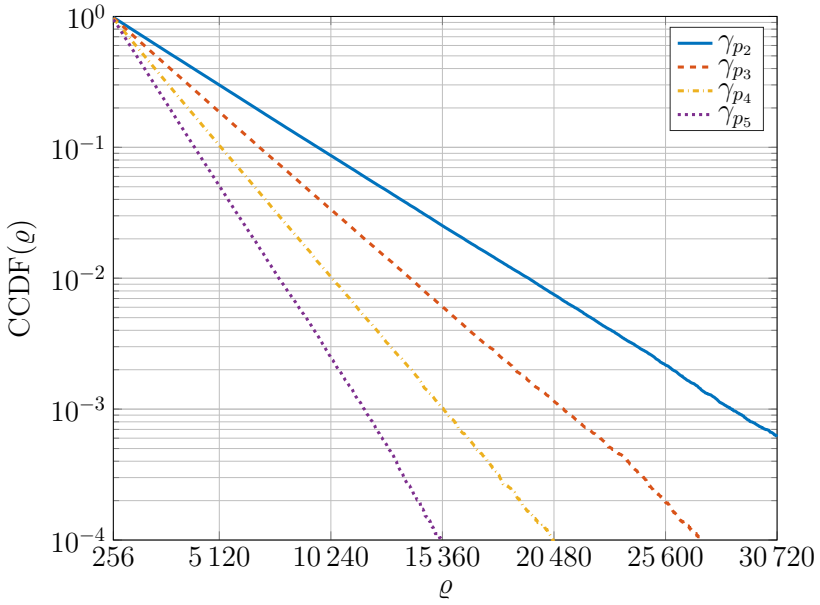



Figure 3.9 – CCDF of the amount of processed samples $\varrho$ with different thresholds $\gamma_{p_i}$ using a successive scanning of $n$ from 0 to $4N-1$.

#### 3.1.3.2 SLM

For comparison purpose, the number of operations required for SLM-OFDM to obtain the same performance (*i.e.*, the same thresholds $\gamma_{p_i}$) are also obtained by computer simulations (to clarify, the IFFTs are computed and the hierarchical sampling, which is not specific to GreenOFDM, is applied too).

The results were that for 99.9% of the simulated cases, a maximum of 3 954 624 operations are computed for $\gamma_{p_2}$ dB, a maximum of $2,674,944$ for $\gamma_{p_3}$ dB, a maximum of 1 874 112 for $\gamma_{p_4}$ dB and a maximum of 1 345 728 for $\gamma_{p_5}$ dB. It is noteworthy that the total number of operations without complexity reduction the SLM-OFDM would require $\frac{U^2}{4} \times 5 \times LN\log_2(LN) + 4 \times \frac{U^2}{4} \times LN \cong 11.5\ MOPS$.

The Computational Complexity Reduction Factor of GreenOFDM as compared with SLM-OFDM by applying the two methods together (the IFFTs-on-demand and the hierarchical sampling) is given by:

$$\mathrm{CRF}_{\mathcal{M}} = \frac{\mathcal{M}_{\mathrm{Green}}(\gamma_{p_i})}{\mathcal{M}_{\mathrm{SLM}}(\gamma_{p_i})} \times 100\% \tag{3.5}$$

Table 3.4 lists the $\mathrm{CRF}_{\mathcal{M}}$ as a function of $\gamma_{p_i}$.

It is clear that the number of required operations for GreenOFDM is significantly lower than that required for SLM-OFDM with the same performance. The main reason is because, as mentioned in the analysis of the $\mathrm{CRF}_v$, for the same performance the same amount of candidates $C_v$ must be available and in SLM-OFDM the required number of IFFTs is proportional to $C_v$ while in GreenOFDM it is proportional to $2\sqrt{C_v}$. In addition, when $\gamma_{p_i}$ decreases, $C_v$

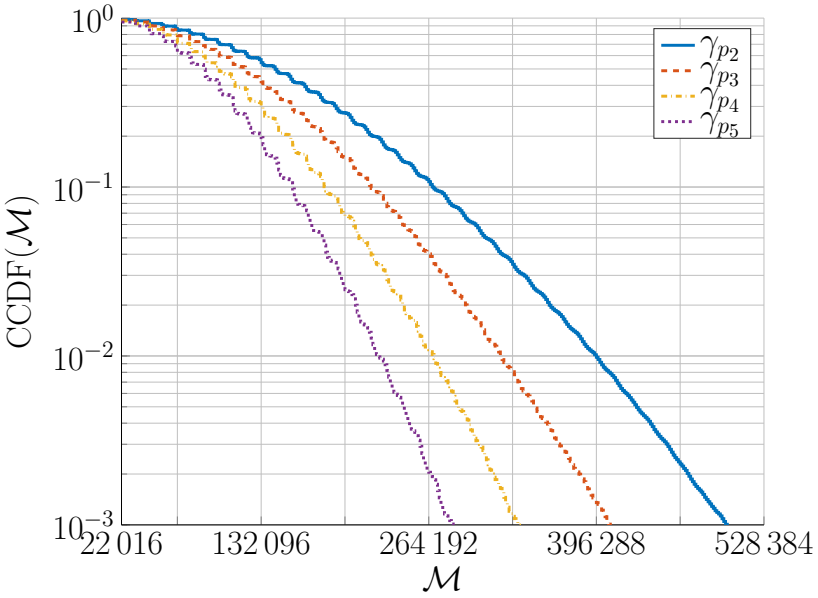



Figure 3.10 – CCDF of the number of operations $\mathcal{M}$ with different thresholds $\gamma_{p_i}$.

| **Threshold** | $\mathcal{M}_{\mathbf{Green}}(\gamma_{p_i})$ | $\mathcal{M}_{\mathbf{SLM}}(\gamma_{p_i})$ | $\mathbf{CRF}_{\mathcal{M}}$ |
|---|---|---|---|
| $\gamma_{p_2}$ | 498 112 | 3 954 624 | 12.6% |
| $\gamma_{p_3}$ | 407 296 | 2 674 944 | 15.2% |
| $\gamma_{p_4}$ | 332 992 | 1 874 112 | 17.8% |
| $\gamma_{p_5}$ | 280 704 | 1 345 728 | 20.9% |

Table 3.4 – $\text{CRF}_{\mathcal{M}}$ between GreenOFDM and SLM-OFDM.

must be increased resulting in a lower CRF approaching 10%.

#### 3.1.3.3 Real time constraint.

To conclude this section, notice that actual implementation of GreenOFDM requires to fix the operating clock frequency of the digital processor. Indeed, to maintain real-time transmission we can choose $f_{\text{clock}} = \frac{\mathcal{M}_0}{T}$ where $T = \frac{N-1}{B}$ (*i.e.*, $T$ is the symbol time duration) and $\mathcal{M}_0$ is the total number of operations for a given probability $\text{CCDF}(\mathcal{M}_0)$ (*e.g.*, $\text{CCDF}(\mathcal{M}_0) = 10^{-3}$).

However, the $\text{CCDF}(\mathcal{M})$ in Figure 3.10 shows that it might happen (with a probability lower than $\text{CCDF}(\mathcal{M}_0)$) that selecting a compliant GreenOFDM candidate requires more operations than $\mathcal{M}_0$.

A solution to this issue with $f_{\text{clock}} = \frac{\mathcal{M}_0}{T}$ is to introduce a FIFO (Firt-In-First-Out) buffer of $K$ selected symbols. With this buffer, real-time operation can be maintained even for symbols requiring up to $K\mathcal{M}_0$ operations.

## 3.2 Energy consumption

In this section, we analyze the energy consumption of the GreenOFDM both for digital and analog implementation to determine the respective impact on the overall consumption of the transmitter. We compare then the consumption of GreenOFDM to other multicarrier schemes to highlight the advantage of implementing GreenOFDM. Finally, we compare the consumption of GreenOFDM with the LoRa modulation.

### 3.2.1 Analysis of the GreenOFDM energy consumption

The analysis of the GreenOFDM energy consumption is decomposed in two parts: the digital implementation consumption and the analog consumption. In the latter, the analog consumption is dominated by the Analog Front-end Power Amplifier [45].

#### 3.2.1.1 The GreenOFDM digital implementation energy consumption

A recurrent thought would be that the GreenOFDM remains a heavy computational charge method and this would arise questions about the energy consumed by the GreenOFDM modulator. In this section we show that the energy required by the GreenOFDM with computational charge reduction remains low as compared with the Power Amplifier (PA) Energy Consumption.

To do so, we have used the dedicated processor from GreenWaves Technologies SAS: the GAP8 IoT processor [46, 47].

As an example we assume a wireless communication with a channel bandwidth of $B = 250$ kHz (hence the symbol time duration is $T = \frac{N-1}{B} = 252\ \mu s$). From Table 3.4 with $\gamma_{p_2}$, we have $\mathcal{M}_{\text{Green}}(\gamma_{p_2}) \approx 500\ kOPS$. When implemented in GAP8, the required number of cycles is reduced by a factor very close to 16 thanks to (i) the 8 programmable computation cores of the processor and (ii) the 32-bit vectorization capability that can be split in two simultaneous 16-bit computations. The total number of cycles when implemented in GAP8 is thus $\frac{500\times10^3}{16} = 31\,250$ and the corresponding required clock frequency is then $f_{\text{clock}} = \frac{31\,250}{T} = 124$ MHz. At this frequency, the GAP8 operating at $1.2V$ has an average power consumption of 60 $mW$. Then the total energy consumption of the digital implementation of the GreenOFDM running on GAP8 is given by:

$$E_{\text{BB}}^{\text{GreenOFDM}} = 60\ 10^{-3}\ T_{\text{signal}}\ (Joules\ [J]) \tag{3.6}$$

where $T_{\text{signal}}$ is the time duration of a GreenOFDM signal composed of several symbols.

#### 3.2.1.2 The PA energy consumption

The PA power consumption is given by [48, 49]:

$$P_{\text{DC}} = \frac{P_{\text{out,avg}}}{\eta_{\text{Class}_\text{X}}} \tag{3.7}$$

where $P_{out,avg}$ is the average power to be sent through the transmission channel and $\eta_{\text{Class}_\text{X}}$ is the PA Class X Power Efficiency (with X being A, B, AB, C, etc, are the acronyms of the different PA that can be found on the market).

A Class A PA (the most linear PA) can deliver a maximum of 50% of power efficiency. The efficiency of a Class A PA in the linear region is given by [49, 50]:

$$\eta_{\text{Class}_\text{A}} = \frac{0.5}{\text{PAPR}} \tag{3.8}$$

where the PAPR is associated to the baseband signal. From Eq.(3.7) and Eq.(3.8), the power consumption of a Class A PA is thus given by:

$$P_{DC}^{(A)} = 2\ P_{\text{out,avg}}\ \text{PAPR} \tag{3.9}$$

With the values from the Long Term Evolution LTE Up-Link, the maximum average power to be sent is +23dBm $\approx 200mW$ [51]. Hence, the power consumption of the Class A PA is given by $P_{\text{DC}}^{(A)} = 2 \times\ 200mW \times \gamma_{\text{p}} = 1.43\ W$.

Assuming that the PA and the digital processor work during the same amount of time [4] $T_{\text{signal}}$, the energy consumption of the PA is hence given by:

$$E_{DC}^{(A)} = 2\ P_{\text{out,avg}}\ \text{PAPR}\ T_{\text{signal}} = 1.43\ T_{\text{signal}}([J]) \tag{3.10}$$

**Digital implementation versus PA energy consumption** The ratio between the energy consumption from the GreenOFDM running on the GAP8 processor (Eq.(3.6)) and the Class A PA (Eq.(3.10)) is given by $\frac{60\times10^{-3}T_{\text{signal}}}{1.43\ T_{\text{signal}}} \approx 4.2 \times 10^{-2}$ (*i.e.*, 4.2%). This means that the digital energy consumption is readily negligible as compared with the PA energy consumption which is the dominant contributor of the transmitter's energy consumption.

#### 3.2.1.3 Overall GreenOFDM energy consumption

Now that we have verified that the digital energy consumption can be neglected as compared with the analog one (the PA is the most energy consuming device in the analog chain), we now study the PA energy consumption to amplify a packet carrying a certain number of bits.

4. The time duration of the frame of symbols to be transmitted.

A GreenOFDM symbol composed of $N$ subcarriers, occupying a bandwidth $B$, has a time duration of $T_{\text{GreenOFDM}} = \frac{N-1}{B}$. The number of bits carried by each GreenOFDM symbol is given by the expression $\alpha N \log_2(M)$ where $\alpha$ corresponds to the portion of the $N$ subcarriers reserved for data transmission ($\alpha \leq 1$) and $M$ is the cardinality of the constellation mapping per subcarrier ($M - QAM$ or $M - PSK$).

The time duration of the concatenated GreenOFDM symbols carrying a total of $L_b$ bits is given by:

$$t_{\text{GreenOFDM}}(L_b) = \frac{L_b}{\alpha N \log_2(M)} \; T_{\text{GreenOFDM}} \; \beta \tag{3.11}$$

where $\beta$ is the total time-domain duration extension due to the time-domain increase because of the addition of the Cyclic Prefix to improve the robustness against ISI.

The energy consumed by a Class A PA (see Eq.(3.9)) to send the concatenated GreenOFDM symbols carrying $L_b$ bits is given by:

$$\mathrm{E}^{(A)}_{\text{GreenOFDM}} = 2 \; P_{\text{out,avg}} \; \gamma_p \; \frac{L_b}{\alpha N \log_2(M)} \; \frac{N-1}{B} \; \beta \tag{3.12}$$

where $\gamma_p$ is the PAPR of the GreenOFDM with $U$ that provides $\text{CCDF}(\gamma_p) = p$ (*cf.* Eq.(3.2)). Changing the value of $\gamma_p$ to the corresponding PAPR of OFDM or SC-FDMA at the given probability $p$ gives the corresponding energy consumed by OFDM $\mathrm{E}_{\text{OFDM}}$ and SC-FDMA $\mathrm{E}_{\text{SC-FDMA}}$.

It becomes obvious from the results of $\text{CCDF}(\gamma)$ obtained in Chapter 3 (*cf.* Figure 2.13) that the PA energy consumption in the case of GreenOFDM is lower than in the case of OFDM and SC-FDMA.

### 3.2.2 LoRa energy consumption

Comparing the GreenOFDM with other OFDM based schemes shows that the GreenOFDM achieves lower energy consumption (because lower PAPR). Comparing the GreenOFDM with orthogonal signaling schemes like LoRa is more challenging.

#### 3.2.2.1 LoRa's digital implementation energy consumption

The energy consumption in the case of a digital implementation of the LoRa chirps should be very low since its modulator consists of selecting 1 among $N$ chirps in a set of chirps $c_{k_0}[n]$ (see 1.3.2) as function of the binary data that must be sent.

In LoRa, the digital consumption of the modulation is to be neglected as compared with the PA energy consumption.

#### 3.2.2.2 Overall LoRa energy consumption

The LoRa modulation encodes $SF$ bits through chirps occupying a bandwidth of $B$ and with time duration of $T_{\text{LoRa}} = \frac{2^{SF}}{B}$. The time duration of the concatenated LoRa symbols carrying a total of $L_b$ bits is given by:

$$t_{\text{LoRa}}(L_b) = \frac{L_b}{SF}\frac{2^{SF}}{B} \tag{3.13}$$

Therefore, from Eq.(3.9) the energy consumed by a Class A PA to send the concatenated LoRa symbols is given by:

$$\text{E}^{(A)}_{\text{LoRa}} = 2P_{\text{out,avg}}\frac{L_b}{SF}\frac{2^{SF}}{B} \tag{3.14}$$

Note that the PAPR of LoRa is 0dB because it is a constant enveloppe modulation.

### 3.2.3 Energy consumption GreenOFDM vs LoRa

The Energy Consumption Ratio (ECR) between the GreenOFDM and the LoRa modulation is given by the ratio between their PA's energy consumption. Considering the same available bandwidth $B$ for both schemes and the same output power $P_{\text{out,avg}}$ to send the same amount of bits $L_b$, the ECR between the two PA's thus reads:

$$\begin{aligned} ECR_{\text{PA}} &= \frac{\text{E}^{(A)}_{\text{LoRa}}}{\text{E}^{(A)}_{\text{GreenOFDM}}} \\ &= \frac{2P_{\text{out,avg}}\ \frac{2^{SF}\ L_b}{SF\ B}}{2P_{\text{out,avg}}\ \gamma_p\ \frac{L_b\ (N-1)\ \beta}{\alpha\ N\ \log_2(M)\ B}} \\ &= \frac{\alpha\ N\ \log_2(M)\ 2^{SF}}{\gamma_p\ SF\ (N-1)\ \beta} \end{aligned} \tag{3.15}$$

An example of the obtained ECR is displayed in Table 3.5 using the parameters from Table 3.1, with $SF \in \{7, \cdots, 12\}$ and, to simplify, with $\alpha = 1$ and $\beta = 1$.

| | **SF** | | | | | |
|---|---|---|---|---|---|---|
| | 7 | 8 | 9 | 10 | 11 | 12 |
| $\mathbf{ECR_{PA}}$ | 10 | 18 | 32 | 57 | 104 | 191 |

Table 3.5 – Energy Consumption Ratio between the LoRa modulation and the GreenOFDM at the transmitter side.

Though the GreenOFDM has a PAPR higher than that of the LoRa modulation, the obtained ECR shows that for GreenOFDM less energy is consumed at the transmitter side. This is mainly because of the spectral efficiency. The same amount of $L_b$ bits requires more LoRa symbols (where each symbol carries $SF$ bits) than in GreenOFDM (where each symbol

carries $\alpha N \log_2(M)$ bits). The PA in the case of the LoRa transmitter is active for a longer period of time, which explains the higher energy consumption.

Note that the previous analysis is simplified as, in practice, LoRa is a constant envelope signal for which a more efficient than class A amplification can be used.

Moreover, to have a totally fair comparison, one needs to compare the systems consumption with the same conditions at the receiver side (*i.e.* at a given range and at a given error probability), as studied in the sequel.

**Comparison for the same error rate and the same coverage.** The obtained ECR considers the transmitter side only. To consider the receiver side too, we have to keep in mind that the comparison is held between different modulation schemes. Because of that, we need to consider the implication of the following settings: (i) the same BER for both schemes and (ii) the same coverage.

To attain the same BER [5] at the receiver side, the required $\frac{E_b}{N_0}$ [6] needs to be determined. Figure 3.11 depicts the spectral efficiency $\eta$ [7] of different modulation schemes as a function of the required $\frac{E_b}{N_0}$ in decibels to attain a BER $= 10^{-5}$. In this figure, the representation of the limit of the Shannon-Hartley theorem is also depicted [8].

On the other hand, to attain the same coverage we introduce a simple example. The "Friis model" [52] of free space propagation in Line-of-Sight (LoS) between the transmitter and the receiver expresses the distance as follows:

$$d^2 = \frac{P_t G_t G_r}{P_r} \left(\frac{\lambda}{4\pi}\right)^2 \tag{3.16}$$

where $P_t$ is the transmitted power, $P_r$ the received power, $G_t$ the transmitting antenna gain, $G_r$ the receiving antenna gain and $\lambda$ the wavelength.

To guarantee the same coverage, the distances for GreenOFDM (say $d_1$) and LoRa (say $d_2$) must be the same. Assuming the same antennas and the same carrier frequency are used

5. ***B**it **E**rror **R**ate*
6. The Signal-to-Noise Ratio (SNR) per bit.
7. The spectral efficiency $\eta$ is the capacity $C$ divided by the available bandwidth $B$: $\eta = \frac{C}{B}$[bits/s/Hz].
8.
$$C \leq B \log_2(1 + SNR)$$
$$\eta \leq \log_2(1 + \frac{E_b}{N_0}\eta)$$
$$\frac{E_b}{N_0} \geq \frac{2^\eta - 1}{\eta}$$

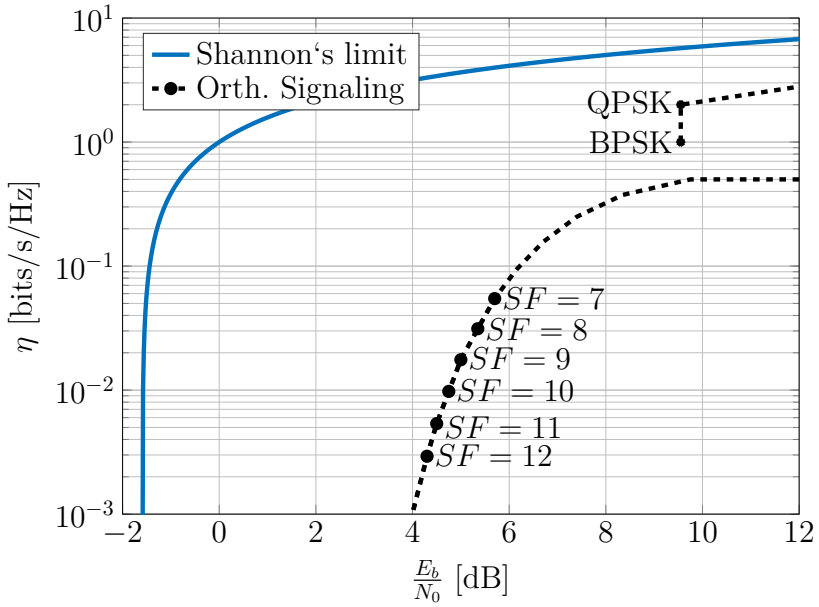



Figure 3.11 – The spectral efficiency as a function of the required SNR per bit to attain a BER = $10^{-5}$ for orthogonal signaling modulations and BPSK/QPSK modulation.

for both GreenOFDM and LoRa, the same coverage is attained for:

$$d_1 = d_2$$
$$\sqrt{\frac{P_{t_1} G_t G_r}{P_{r_1}}} \frac{\lambda}{4\pi} = \sqrt{\frac{P_{t_2} G_t G_r}{P_{r_2}}} \frac{\lambda}{4\pi}$$
$$\frac{P_{t_1}}{P_{r_1}} = \frac{P_{t_2}}{P_{r_2}}$$

Then, with the definition of the SNR at the receiver, *i.e.*, SNR = $\frac{P_r}{P_n}$ and with SNR = $\frac{E_b}{N_0}\eta$ we have:

$$P_{t_1} = P_{t_2} \frac{E_b/N_{0_1}\ \eta_1}{E_b/N_{0_2}\ \eta_2} \tag{3.17}$$

Including $P_{t_1}$ in Eq.(3.17) instead of $P_{\text{out,avg}}$ in the expression of $\mathrm{E}^{(A)}_{\text{GreenOFDM}}$ and $P_{t_2}$ instead of $P_{\text{out,avg}}$ in the expression of $\mathrm{E}^{(A)}_{\text{LoRa}}$ let us obtain a new expression for the ECR with the same BER and the same coverage:

$$\begin{aligned} ECR &= \frac{P_{t_2}\ \alpha\ N\ \log_2(M)\ 2^{SF}}{P_{t_1}\ \gamma_p\ SF\ (N-1)\ \beta} \\ &= \frac{E_b/N_{0_2}\ \eta_2\ \alpha N\ \log_2(M)\ 2^{SF}}{E_b/N_{0_1}\ \eta_1\ \gamma_p\ SF\ (N-1)\ \beta} \end{aligned} \tag{3.18}$$

Table 3.6 summarizes the values from Figure 3.11 of the spectral efficiency as a function of the required $\frac{E_b}{N_0}$ to attain the BER = $10^{-5}$. The following Table 3.7 displays the ECR

| **Modulation Scheme** | $\frac{E_b}{N_0}$**[dB]** | $\eta$ |
|---|---|---|
| LoRa $SF = 7$ | 5.7 | 0.055 |
| LoRa $SF = 8$ | 5.35 | 0.031 |
| LoRa $SF = 9$ | 5 | 0.018 |
| LoRa $SF = 10$ | 4.75 | 0.0098 |
| LoRa $SF = 11$ | 4.5 | 0.0054 |
| LoRa $SF = 12$ | 4.3 | 0.0029 |
| BPSK | 9.55 | 1 |
| QPSK | 9.55 | 2 |

Table 3.6 – Required $\frac{E_b}{N_0}$[dB] for $BER = 10^{-5}$ and the corresponding spectral efficiency $\eta$ of the modulation schemes depicted in Fig.3.11.

with the corresponding values of $\eta$, $\frac{E_b}{N_0}$ and using the parameters from Table 3.1, with $SF \in \{7, \cdots, 12\}$ and with $\alpha = 1$ and $\beta = 1$. These results show that the energy efficiency of LoRa modulation overcomes by more than an order of magnitude the overall energy consumption of the GreenOFDM at the receiver side. This shows that the energy-efficiency schemes benefit is at the advantage of the LoRa scheme.

| | **SF** | | | | | |
|---|---|---|---|---|---|---|
| | 7 | 8 | 9 | 10 | 11 | 12 |
| **ECR** | 0.11 | 0.11 | 0.098 | 0.092 | 0.087 | 0.083 |

Table 3.7 – ECR between the LoRa modulation and the GreenOFDM for the same BER and the same coverage at the receiver side.

## 3.3 Conclusion of the Chapter

In this chapter we have introduced a method named IFFTs-on-demand to reduce, in a probabilistic manner, the GreenOFDM computational burden (number of IFFTs). This method requires to choose adequately a threshold $\gamma_{p_i}$ that impacts the performance of computation reduction. We obtained through computer simulations the distribution of the number of computed IFFTs as a function of the chosen threshold $\gamma_{p_i}$ and showed that the number of IFFTs to compute is lower than $U$.

Then we introduced another method, named hierarchical sampling. The goal of this method is to reduce the number of samples that are to be calculated for the candidates that are not worth for transmission. This reduction is obtained thanks to a particular scanning order of the samples.

We presented the total number of operations needed when IFFTs-on-demand and hierarchical sampling are combined.

In the last section of the chapter we focused on the energy consumption. The first analysis

led us to conclude that the energy consumption of the digital implementation of the threshold GreenOFDM can be neglected as compared with the PA energy consumption. This fact led us to conclude also that the comparison of GreenOFDM with other multicarrier schemes in terms of energy consumption becomes a PAPR comparison. In terms of PAPR, the GreenOFDM was shown to outperform both SLM-OFDM and SC-FDMA (*cf.* Chapter 2).

Finally, we compared the GreenOFDM with LoRa. We obtained that in all cases, LoRa's energy consumption is lower than GreenOFDM's energy consumption. This is mainly because the LoRa modulation belongs to the family of orthogonal signaling schemes that is more efficient from an energy point of view.

Chapter 4

# Implementation of couple of OFDM based IoT schemes on GAP8

## Contents



In this Chapter, we focus on the implementation on the GreenWaves Technologies SAS GAP8 IoT processor.

First, we present some details about: (i) IEEE 802.15.4g and (ii) 3GPP NB-IoT.

Implementations of GreenOFDM, IEEE 802.15.4g and 3GPP NB-IoT were part of the European Commission program H2020 SMEINST HEASIT project N$^o$ 709841 granted to GreenWaves Technologies SAS.

Then we present the overview and some figures about the implementation of the IEEE 802.15.4g and the GreenOFDM modems on the GAP8 processor.

## 4.1 The two studied OFDM based communication schemes

In this section, we present both the IEEE 802.15.4g and the 3GPP NB-IoT as they were both studied in detail in order to implement them on the GAP8 processor.

### 4.1.1 The IEEE 802.15.4g MR-OFDM

The IEEE 802.15.4 [53] is an IEEE standard for Low-Rate Wireless Personal Area Networks (LR-WPAN) composed of several task groups. Among them there is the IEEE 802.15.4g task group which is specific to Smart Utility Network (SUN) applications. This task group proposes 3 different multi-band and multi-rate PHY [1] layers:

— The MR-FSK (for Multi-Rate Frequency Shift Keying).
— The MR-OFDM (for Multi-Rate Orthogonal Frequency Division Multiplexing).
— The MR-O-QPSK (for Multi-Rate Offset Quadrature Phase Shift Keying).

The MR-OFDM was chosen for study and implementation in GreenWaves Technologies SAS GAP8 processor since it is OFDM based. The MR-OFDM is the PHY of IEEE 802.15.4g with the highest speed rate among the 3 different PHY.

The basic MR-OFDM building blocks are displayed on Figure 4.1. Here, several general blocks can be found: the channel coding (Encoder), the digital modulation, the analog Front-end, etc.

A global overview of the MR-OFDM PHY block diagram is presented.

**The global overview.** The MR-OFDM is defined by two main parameters: the Options (ranging between 1 to 4 for various occupied bandwidths and so for different number of channels) and the Modulation and Coding Scheme or MCS (ranging between 0 to 6 for various bit rates). Once both parameters are set, all the PHY parameters of the OFDM modulation chain are defined. The Options define: IFFT size, data carriers, pilote carriers, bandwidth. The MCSs define: constellation mapping, coding rate, spreading factor. Table 4.1 summarizes all the MR-OFDM parameters as a function of the Options and the MCSs.

The MR-OFDM signal is composed by a concatenation of OFDM symbols carrying the data bits that come from the IEEE 802.15.4 MAC [2] layer. This MR-OFDM signal has three categories of OFDM symbols,

— **Synchronization HeadeR (SHR) symbols** which are used for the fine and the coarse synchronization at the receiver side, *i.e.* to correctly estimate the beginning of the frame and the frequency domain channel impulse response. This set of symbols is referred as the *Synchronization Header*.

1. Stands for PHYsical layer.
2. Medium Access Control.

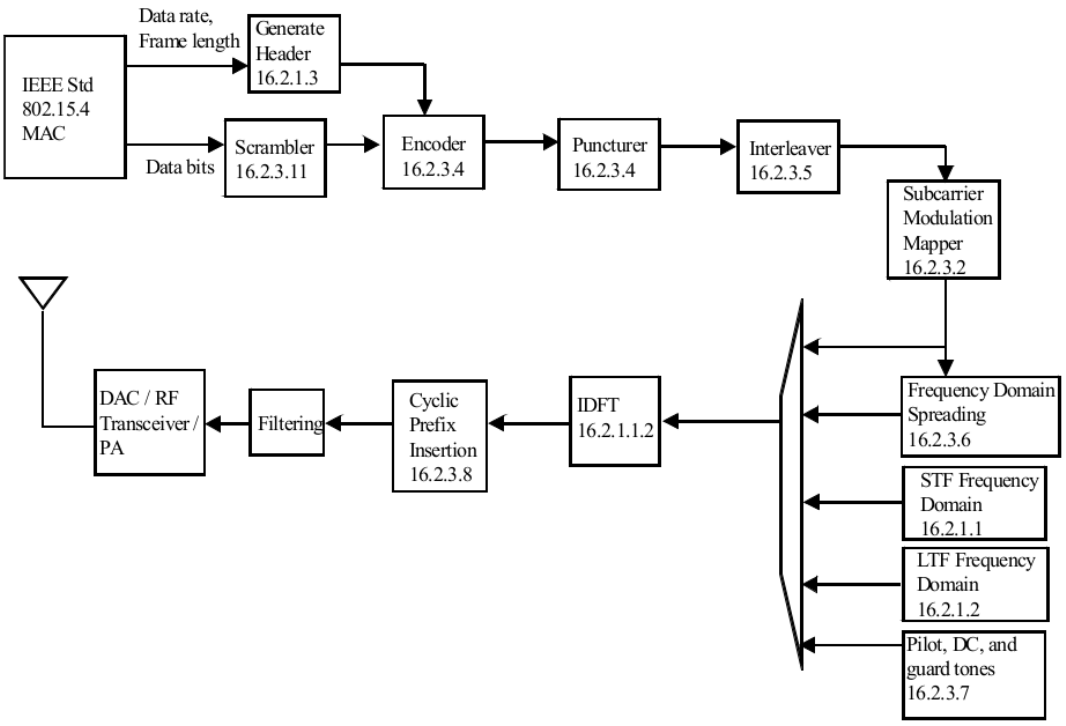



Figure 4.1 – Block diagram of the IEEE 802.15.4g MR-OFDM physical layer (from [53], p. 81).

| Parameter | OFDM Option 1 | OFDM Option 2 | OFDM Option 3 | OFDM Option 4 |
|---|---|---|---|---|
| Nominal bandwidth (kHz) | 1094 | 552 | 281 | 156 |
| Channel spacing (kHz) | 1200 | 800 | 400 | 200 |
| FFT size | 128 | 64 | 32 | 16 |
| Active tones | 104 | 52 | 26 | 14 |
| # Pilot tones | 8 | 4 | 2 | 2 |
| # Data tones | 96 | 48 | 24 | 12 |
| MCS0 (kb/s) (BPSK rate 1/2 4x spreading) | 100 | 50 | — | — |
| MCS1 (kb/s) (BPSK rate 1/2 2x spreading) | 200 | 100 | 50 | — |
| MCS2 (kb/s) (QPSK rate 1/2 2x spreading) | 400 | 200 | 100 | 50 |
| MCS3 (kb/s) (QPSK rate 1/2) | 800 | 400 | 200 | 100 |
| MCS4 (kb/s) (QPSK rate 3/4) | — | 600 | 300 | 150 |
| MCS5 (kb/s) (16-QAM rate 1/2) | — | 800 | 400 | 200 |
| MCS6 (kb/s) (16-QAM rate 3/4) | — | — | 600 | 300 |

Table 4.1 – Parameters of 802.15.4g MR-OFDM

— **PHY HeadeR (PHR) symbols** that carry the information about the modulation mapping (BPSK, QPSK, ...), the total number of data symbols, the coding rate, etc. This set of symbols is referred as the *Physical Header*.
— **PHY Payload symbols** that carry, as indicated by its name, the useful data bits. This set of symbols is referred as the *PHY Payload*.

#### 4.1.1.1 Synchronization HeadeR (SHR) symbols

The SHR symbols are calculated according to the chosen Option. Two kinds of symbols are defined:

(i) The Short Training Field (STF) symbols, composed of 4 OFDM symbols used at the receiver side for the so-called coarse-time synchronization and

(ii) The Long Training Field (LTF) symbols, composed of 2 OFDM symbols used at the receiver side for the so-called fine synchronization and channel estimation.

The frequency domain OFDM symbol for the STF and LTF are predefined in look-up tables in the standard IEEE 802.15.4g specification document [53]. The IFFT of the SHR look-up tables calculates the OFDM symbol in time, then Cyclic Prefix (CP) is prepended. For all the OFDM symbols (Synchronization symbols, Header symbols and Data symbols), the CP corresponds to $\frac{1}{4}$ of the time-domain samples of the OFDM symbol.

#### 4.1.1.2 PHY HeadeR (PHR) and PHY Payload

The PHR symbols and the PHY Payload symbols follow the same steps as depicted in Figure 4.1. They begin however with a different block for each of them, the binary Header generation in the PHR case (starting from block "Generate Header 16.2.1.3") and a scrambling operation in the PHY Payload case (starting from block "Scrambler 16.2.3.11"). In the following paragraphs we present each block beginning with the PHR symbols path and then we present the other block present in the PHY Payload, *i.e.* the scrambler.

**PHR composition.** The PHR OFDM symbols are generated by first composing the binary PHR as shown in Table 4.2. Here the Reserved fields are not used and are set to 0. For the other fields we have:

— the Rate field that corresponds to the binary value of the MCS,
— the Frame Length field that corresponds to the number of bytes in the data payload,
— the Scrambler field that corresponds to the index of the 4 different seeds used for scrambling the data payload, and
— the HCS field that corresponds to a particular CRC[3]-8 over the 22 first bits of the PHR.

3. CRC stands for Cyclic Redundancy Check and is used to detect errors on the receiver side.

| **Bit string index** | **0-1** | **5** | **6-16** | **17-18** | **19-20** | **21** | **22-29** | **30-35** |
|---|---|---|---|---|---|---|---|---|
| **Bit mapping** | $RA_4$-$RA_0$ | R | $L_{10}$-$L_0$ | $R_1$-$R_0$ | $S_1$-$S_0$ | R | $H_7$-$H_0$ | $T_5$-$T_0$ |
| **Field name** | Rate | Reserved | Frame Length | Reserved | Scrambler | Reserved | HCS | Tail |

Table 4.2 – PHR fields.

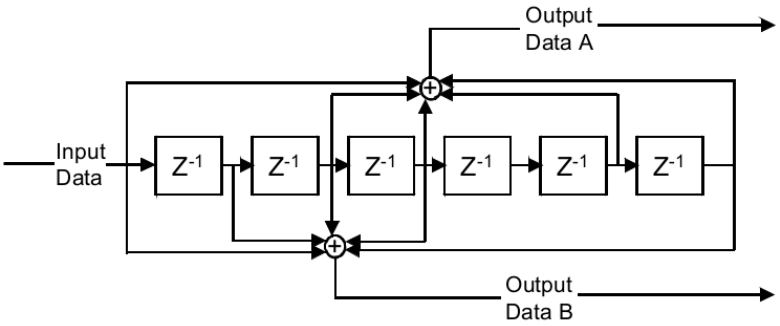



Figure 4.2 – Convolutional Encoder.

**Convolutional Encoder and Puncturing.** The PHR is encoded by the Convolutional Encoder of rate $\frac{1}{2}$ as depicted in Figure 4.2. Puncturing [4] is applied in order to achieve a coding rate of $\frac{3}{4}$, though it is only performed in Data Payload symbols with the corresponding MCS (see Table 4.1). PHR symbols are always encoded using the lowest MCS of the chosen Option.

**Interleaver.** Prior to subcarrier constellation mapping, encoded bits are interleaved. This means that the encoded binary data is permuted according to a predefined formula of permutation indexes. A single permutation is applied if the constellation mapping is BPSK or QPSK, a second permutation is applied if constellation mapping is 16-QAM. The permutation formulaes are defined in the specification for all cases [5].

**Spreading.** Before the frequency domain OFDM symbol is generated, the data output of the constellation mapper is frequency spreaded (only if the MCS is lower than 3). The frequency spreading consists in a repetition of the subcarriers prior to phase change (the changes of the phases are $\in \{\pm\pi/2, \pi\}$). The goal of this repetition is to have a frequency diversity at the receiver side.

**Frequency domain symbol generation.** Finally, the frequency domain OFDM symbol is generated by adding the pilot tones to a set of predefined frequency indexes according to the

4. Consisting in deleting the 4th and 5th bit every 6 output bits.
5. A single permutation is applied in PHR symbols because the lowest MCS is always used.

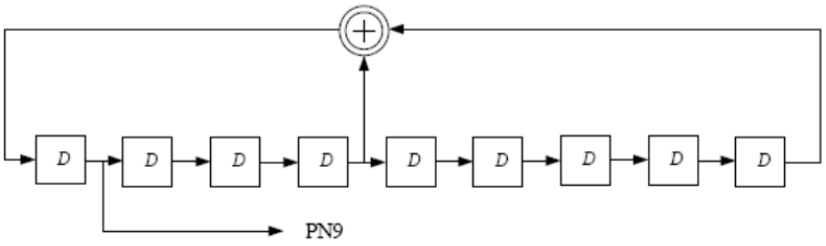



Figure 4.3 – The PN9 sequence.

chosen Option. Each pilot is BPSK modulated and its values are generated using the PN9 sequence of Figure 4.3 initialized with all 1's.

**Scrambler.** The data payload OFDM symbols follow the same blocks as previously presented. Besides, an additional block appears prior to convolutional coding: the scrambler.

The inputs to the scrambler are the data bits, coming from the MAC layer, followed by tail bits and pad bits. The tail bits (six consecutive 0 bits) are used to return the convolutional encoder to the "zero state" and the pad bits are needed to have an integer number of OFDM symbols. The scrambler uses a PN9 sequence as shown in Figure 4.3. The seed for the scrambler is initialized from a look up table of four different seeds. The seed to be used is chosen according to bits 19 and 20 of the PHR field (see Table 4.2).

On the receiver side, the dual operations for each block are to be applied.

### 4.1.2 NB-IoT

The Narrow Band IoT (NB-IoT) [54, 55] was introduced in the 3GPP's[6] 13$^{th}$ Release as a PHY layer for Low Power Wide Area Networks (LPWAN). The NB-IoT shares the same structure than the 3GPP LTE (Long Term Evolution). Two modulation schemes are proposed for uplink communications: the SC-FDMA (as in 4G LTE) and the single carrier BPSK/QPSK.

**The band modes.** NB-IoT communications are available in three different band modes: (i) the in-band mode, where the NB-IoT uses the Physical Resource Blocks (PRB) within the existing LTE carrier, (ii) the guard band mode, where the NB-IoT uses the guard bands of the LTE carrier and (iii) the standalone mode, where the NB-IoT re-uses GSM and UMTS available carriers.

**The Downlink and Uplink channels.** The downlink communication (transmitted by the Base Station — or BS — side) uses OFDMA with convolutional codes for broadcasting

6. 3$^{rd}$ Generation Partnership Project

frames to all the UE (User Equipment). Several "channels" are used to carry different kinds of information. The downlink radio frame (which is not going to be detailed here as it is not the goal of this section) can be summarized by the following subframes:

— NPBCH (Narrowband Physical Broadcast CHannel): carries master information.
— NPDCCH (Narrowband Physical Downlink Control CHannel): carries scheduling information and HARQ (Hybrid Automatic Repeat reQuest) acknowledgement.
— NPDSCH (Narrowband Physical Downlink Shared CHannel): carries data, paging, system information and RAR (Random Access Response).
— NPSS (Narrowband Primary Synchronization Signal): this frame is used by the UE for cell search and time and frequency synchronization.
— NSSS (Narrowband Secondary Synchronization Signal): same goal as NSSS. This signal is only present on even numbered frames.

In the Uplink (UE side) the communication happens through two channels: (i) the NPRACH (Narrowband Physical Random Access CHannel) that is used for initial random access to the Base Station (BS) and (ii) the NPUSCH (Narrowband Physical Uplink Shared CHannel) that is used to send data and control information.

The NPRACH channel uses single tone frequency hopping to achieve the initial synchronization with the BS. The BS answers with the Resource Block (RB) allocation for the UE. Then, NPUSCH can be used to send the data payload using the allocated RB. NPUSCH uses SC-FDMA or single tone BPSK/QPSK according to the number of subcarriers allocated by the BS.

After studying each channel, we have found that the NPUSCH channel is the most computational demanding if we consider its implementation. Further details are presented in Annex B for the NPUSCH channel. Estimations about the NPUSCH running on GAP8 show that the processor limits are exceeded, reason why the NB-IoT was not implemented on GAP8.

## 4.2 Implementation on GAP8

The implementation of the MR-OFDM and GreenOFDM on the GAP8 processor [47] followed three different stages: (i) First, we began with a MATLAB model for the main digital blocks of the transmitter and the receiver, this model was then used as a reference for the other stages. (ii) We ported the MATLAB model into C-code in fixed-point arithmetic running on a PC before (iii) porting all the code to fit on the GAP8 processor (memory wise, adapting routines to run in parallel, etc).

The GAP8 processor, at that time was not ready so we had a Xilinx FPGA emulation of the GAP8 processor. This emulation of the GAP8 processor was another deliverable of the same H2020 project.

For a transmit-receive validation, two FPGA's were required. We had also a Radio Frequency Front-end (this RF Front-end is also another deliverable of the H2020 project).

### 4.2.1 About the implemented algorithms

So far we gave a global overview of the transmitter block chain. Before summarizing the results of the implementation on the GAP8 processor, we describe the implemented algorithms, where some of them are common for the different communication schemes. Also, the corresponding algorithms implemented on the receiver side are described.

#### 4.2.1.1 The transmitter side

First of all, it is noteworthy to mention that for all the schemes, the starting point was a chain of bytes to be transmitted since we only implemented the physical layer. Then, the following functions where implemented:

**The header** Since the transceiver is expected to be configurable, the corresponding header was built considering the different parameters (e.g. as in MR-OFDM [7]) of the system.

**The scrambler** The shift register in figure 4.3 was implemented taking into account the corresponding seed. The output bits of this shift register was used to scramble the binary data through exclusive or operation.

**The convolutional code** The GAP8 processor's signal processing library had already a convolutional code implementation to which we only had to give the boolean polynomials (corresponding to $g_0 = 133_8$ and $g_1 = 171_8$ according to the MR-OFDM specification document).

**The permutations** For the permutations some look-up tables were built containing the indexes resulting from the equations described in the standard.

**The subcarrier mapping** For the subcarrier mapping, also look-up tables were used, one for each constellation (BPSK, QPSK and 16-QAM).

**The frequency spreading** The frequency spreading operation described in MR-OFDM is basically a frequency repetition of the data that is first multiplied by a given coefficient set $\in \{1, i, -1, -i\}$. This was simply implemented by a concatenation of the data prior coefficient multiplication.

7. an equivalent header was implemented for the GreenOFDM

**The pilot tones** The pilot tones positions were implemented through look-up tables too. The modulated value on each subcarrier were generated with BPSK modulation over a binary data at the output of the same PN-sequence as for the scrambler using a seed set to all ones.

**The synchronization symbols** The synchronization symbols, as they are constant to every frame, were implemented through look-up tables of the time domain samples (for both the STF and the LTF).

**The other functions** The remaining transmitter side functions include:

— The construction of the frequency domain symbol with the data subcarriers indexes implemented with look-up tables, the pilot tones, the null subcarriers and, in the particular case of GreenOFDM, some subcarriers dedicated to send the side information.

— The IFFT over the frequency domain symbols in order to generate the time domain OFDM symbols. Here the Radix-2/4 FFT algorithms that were already implemented in the GAP8 processor's signal processing library were used.

— The appending of the CP: a copy of the last quarter of the time domain symbols

— The concatenation of the OFDM symbols in the right order to generate an OFDM signal that was then copied into a buffer and sent to the LVDS interface developed by other colleagues for the RF transmission.

In the particular case of GreenOFDM, the methods described in Chapter 3 for the computational charge reduction were also implemented (IFFTs-on-demand and hierarchical sampling using the threshold $\gamma_{p_2}$).

#### 4.2.1.2 The receiver

Here, the corresponding transmitter reverse functions were implemented. However, the first function corresponds to an algorithm that is not described in the standards and for which no special processing was applied in the transmitter side. It is the synchronization algorithm. Here we describe the basic principle of the synchronization algorithms before briefly summarizing the remaining functions.

The starting point is a buffer of RF demodulated In-phase and Quadrature-phase digital data. The RF demodulation was developed by other colleagues. This buffer was the starting point to perform the digital synchronization.

The synchronization header is composed of two main fields as presented before, the STF and the LTF. Each of them is going to be used for a specific task in the synchronization process. These tasks are divided into two: (i) the coarse synchronization (where the STF symbols are used) and (ii) the fine synchronization (where the LTF symbols are used).

**The coarse synchronization** The goal of the coarse synchronization is to detect the beginning of a frame. To do so, the correlation properties of the STF symbols are exploited.

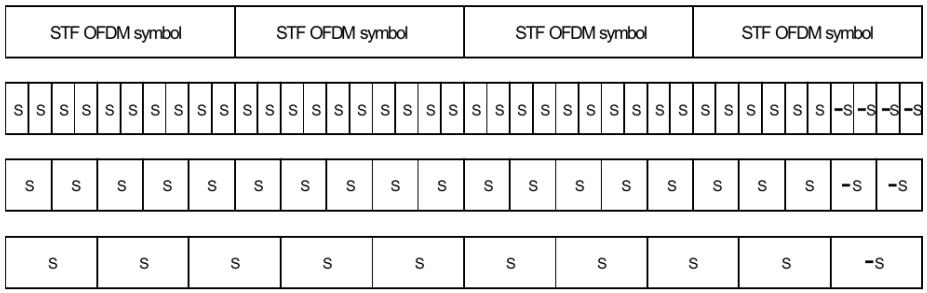



Figure 4.4 – Short Training Field time domain representation.

Figure 4.4 depicts the time domain structure of the STF composed of 4 OFDM symbols. These symbols are, as it could be seen, the result of repeating an element denoted by 's'. At the receiver side, the correlation is computed over a sliding window containing twice the amount of samples of this element 's'. The result of this correlation, at the presence of a frame, is depicted in Figure 4.5 where only the modulus is represented.

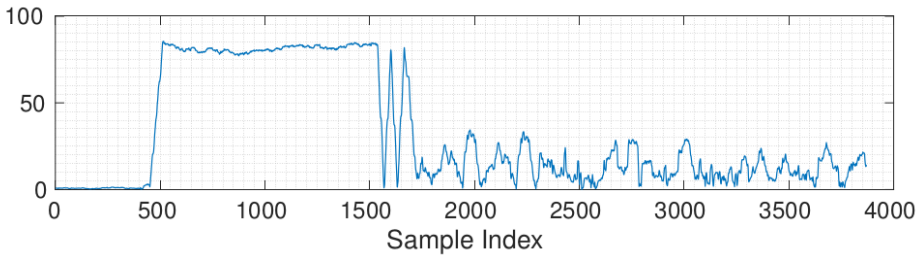



Figure 4.5 – The modulus of the correlation computed at the receiver side to obtain the coarse synchronization.

This result is exploited with some thresholding to obtain a more precise estimation of the beginning of the frame as well as any phase offset to be corrected (see Figure 4.6). This eventual phase error could be seen to be constant during the STF. The estimation of this value is then used to correct the phase of the received frame.

The sychronization process continues with the samples that come after the point where the correlation of the STF drops.

**The fine synchronization** The goal of the fine synchronization is to define with more precission the beginning of the received frame and to achieve initial frequency channel synchronization, *i.e.* to have an estimation of the channel frequency response. To do so, the FFT is computed over the LTF OFDM symbols and with the frequency domain LTF look-up tables, the initial channel estimation $\{H_k\}$ is obtained.

Then, through IFFT the Channel Impulse Response is calculated. The CIR is useful to know if we are synchronized with the most representative path of the multi-path channel.

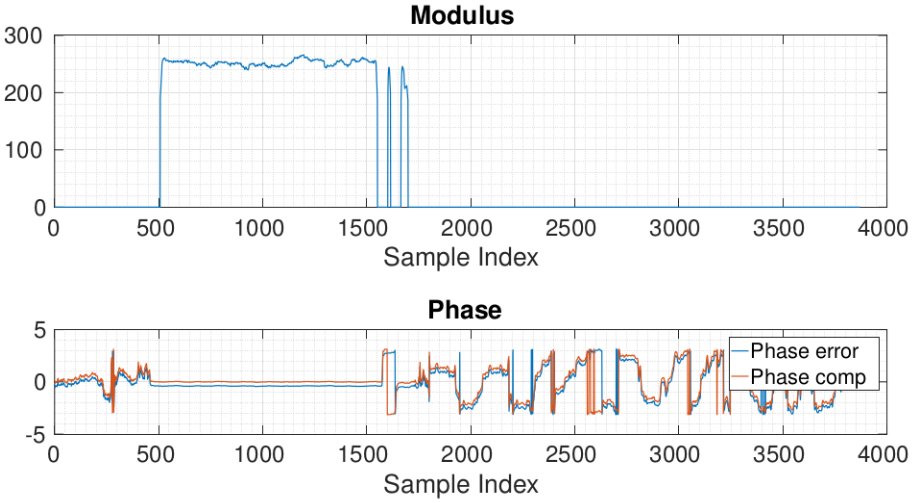



Figure 4.6 – The modulus and the phase of the correlation computed at the receiver side to obtain the coarse synchronization.

Figure 4.7 depicts an example for which the beginning of the frame must be around 10 samples later than that obtained from the coarse sychronization. Once this start of the frame is corrected, the channel frequency estimation should be updated too (the delay of samples in time is represented by a complex coefficient multiplication in frequency so no other IFFT needs to be calculated).

Once these two synchonization functions are computed, the received frame can be demodulated.

**The other functions** The remaining functions are the reverse operations of the previously described methods of the transmitter that include:

- CP suppression: bypass the amount of samples that correspond to the CP in the received symbol window.
- Correction of the transmit/receive phase offset with the estimation from the coarse synchronization algorithm.
- FFT of the time-domain OFDM symbol demodulation.
- Channel re-estimation and interpolation (track the channel evolution in time). To achieve this goal, the receiver uses its LUT of pilot symbols to re-estimate the channel coefficients at the corresponding sub-carrier indexes. The remaining sub-carriers were obtained through linear interpolation.
- Equalization through zero-forcing algorithm ($\frac{\hat{H}_k^*}{|\hat{H}_k|^2}$).
- In the case where a frequency spreading is applied on the transmitter side, a de-spreading operation was implemented consisting in an addition of the repeated subcarriers.
- Demapping: the lowest distance of each received sub-carrier with respect to the constellation points. In the particular case of GreenOFDM, the dedicated subcarriers for Side

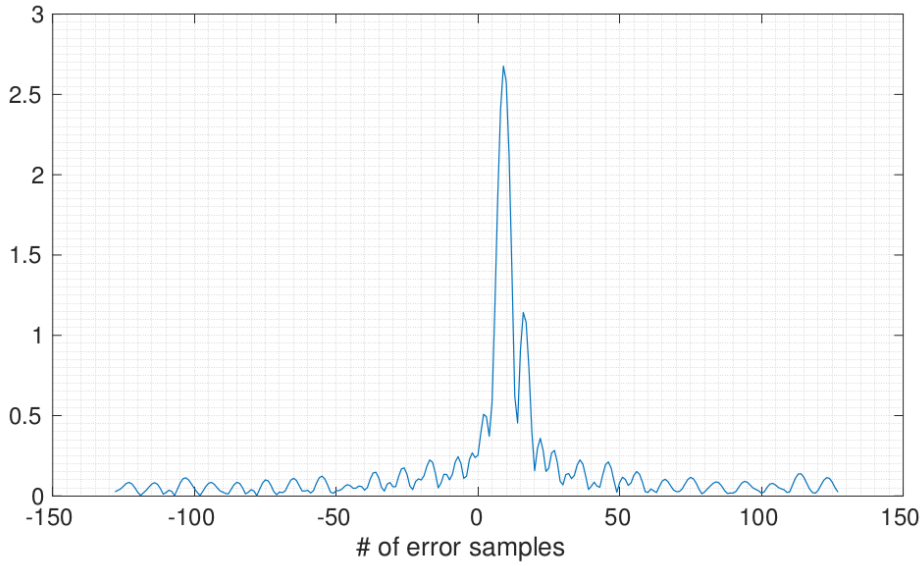



Figure 4.7 – Channel Impulse Response obtained through the fine synchronization estimation.

Information are first demodulated in order to de-randomize the received subcarriers.

— Permutation of the received bits to have them in the right order, using the corresponding indexes LUTs to those used at the transmitter side.
— Viterbi decoder using the already existing GAP8 processor's signal processing library with hard decision.
— Header symbols data extraction concerning the subcarrier mapping used in data symbols, total number of symbols, etc.
— De-scrambling using the same shift register as in the transmitter side.

Now that the main algorithms were overviewed, we continue with the profiling of the implementations.

### 4.2.2 IEEE 802.15.4g MR-OFDM on GAP8

After a detailed study of the MR-OFDM specification [53], we proceeded with the implementation. As previously mentioned, the first step was a Transmit/Receive model using MATLAB. This model was validated by generating the same frame example as given in the MR-OFDM specification document (these examples appear on Annexes L and M of [53]) in order to be sure that the implementation was standard compliant.

Then, the MATLAB floating-point model was ported into C-code in fixed-point arithmetic running on a PC. This intermediate step eased up the final porting to GAP8 that supports fixed point arithmetic and where an additional effort was required to parallelize some routines in order to exploit the cluster of 8 cores. After coding the transmitter and the corresponding receiver functions on GAP8, a RF transmission between two FPGA [8] cards with RF Front-end

8. At that moment, the GAP8 chip was emulated on FPGA before it was taped-out.

| Function | Avg # Cycles | Clock Freq. |
|---|---|---|
| Before IFFT | 3593 | 30 MHz |
| IFFT + CP | 2348 | 19.6 MHz |
| **Total** | **5941** | **49.6 MHz** |

Table 4.3 – Benchmark for the Header Symbols.

| Function | Avg # Cycles | Clock Freq. |
|---|---|---|
| Before IFFT | 3543 | 29.5 MHz |
| IFFT + CP | 2203 | 18.4 MHz |
| **Total** | **5746** | **47.9 MHz** |

Table 4.4 – Benchmark for the Data Symbols.

was achieved. We present next the profiling of the number of cycles and the required clock frequency for the transmitter and the receiver side on the GAP8 processor.

#### 4.2.2.1 The transmitter

The profiling of the transmitter was performed with the example given in the Annex of the MR-OFDM specification (Option 2 and MCS 3). An interesting aspect of the MR-OFDM is that all the OFDM symbols have a constant time duration of 120 $\mu s$ (96 $\mu s$ for the OFDM symbol + 24 $\mu s$ for the CP) regardless of the chosen Option.

The profiling was divided in two parts: (i) the operations that took place before IFFT (Constellation Mapping, Spreading, Pilots indexes and data indexes set, frequency domain OFDM symbol construction) and (ii) the IFFT + CP. The results for the average number of cycles and the equivalent clock frequency to maintain real time for 120 $\mu s$ symbols can be found on Tables 4.3 and 4.4. We can see from the obtained results that a clock frequency of 50 MHz is sufficient to guarantee real-time operation.

#### 4.2.2.2 The receiver

The receiver includes the dual operations of the transmitter plus some specific operations such as the frame synchronization and the channel equalization. We decided to divide the receiver in three main stages:

— Coarse Synchronization stage which is a correlation based operation in which the starting point of a received frame is estimated as well as the phase offset coming from Transmitter-Receiver clock impairments. It is performed over the Short Training Field symbols.

— Fine Synchronization stage that is triggered by the coarse synchronization results. This operation performs the channel estimation and improves the estimate of the beginning of the frame. These calculations are performed over the Long Training Field symbols.
— The final stage corresponds to the Physical Header and Data Payload OFDM symbols demodulation. Here several operations are held including phase offset compensation, CP suppression, FFT demodulation, channel equalization, interpolation, QAM demapping, Viterbi decoding, etc.

The Coarse and Fine Synchronization are common stages of the GreenOFDM receiver which was first ported into GAP8. Given so, the same code was used with just a slight change (coming from the preamble length). Then the inverse operations of the transmitter were ported to GAP8.

In what follows we detail in three parts the benchmarking: (i) the synchronization (coarse and fine synchronization) part, (ii) the Header Symbols demodulation part and (iii) the Data Symbols demodulation part.

**Synchronization.** Table 4.5 shows the average number of cycles required to perform the coarse synchronization and the fine synchronization. Regarding the coarse synchronization, the number of cycles that appear in the table corresponds to a total of 4862 processed samples before finding the beginning of a received frame. Since an oversampling factor of 4 was used in the receiver to perform the synchronization, then the 4862 processed samples would be the equivalent of approximately 15 symbols [9]. The clock frequency to maintain real-time is then obtained (recall that each symbol has a time duration of 120 $\mu s$). Regarding the fine synchronization, recall that it is computed over two OFDM symbols.

| Function | Avg # Cycles | Clock Freq. |
|---|---|---|
| Coarse Synchronization | 222828 | 122.21 MHz |
| Fine Synchronization | 25062 | 104.42 MHz |

Table 4.5 – Benchmark for the coarse and the fine synchronization.

**Header demodulation.** The average number of cycles and the required clock frequency for the header symbols demodulation are displayed in Table 4.6. The average is computed over 6 OFDM symbols that compose the PHR at each received frame. The results show that a clock frequency of 98.51 MHz is sufficient to guarantee real time operation.

**Data demodulation.** The average number of cycles and the required clock frequency for the data symbols demodulation are resumed in Table 4.7. The average is computed over 13 OFDM symbols that compose the PHY payload at each received frame.

9. Each symbol in Option 2 has 64 samples plus $\frac{1}{4}$ for the CP, that is 80 samples in total. With the 4 times oversampling we have then 320 samples per symbol.

| Function | Avg # Cycles | Clock Freq. |
|---|---|---|
| Phase Offset | 2947 | 24.56 MHz |
| CP sup. + FFT | 3071 | 25.6 MHz |
| Interpolation & Equalization | 456 | 3.8 MHz |
| Header de-mapping | 5347 | 44.56 MHz |
| **Total** | 11821 | **98.51** $MHz$ |

Table 4.6 – Benchmark for the header symbols demodulation.

An important remark is that the benchmark obtained for the header and the data demodulation differ significantly for the de-mapping process which includes the Viterbi decoder. In the case of the header, the Viterbi decoder is only activated once (so after demodulating 6 OFDM symbols) while in the case of the data, the Viterbi decoder is activated at each OFDM symbol.

The results show that a clock frequency of 165.16 MHz is sufficient to guarantee real time operation. It can be concluded that the data demodulation is the most demanding task with regards to the required clock frequency, so that setting the GAP8 processor running at 165.16 MHz guarantees to handle the MR-OFDM.

| Function | Avg # Cycles | Clock Freq. |
|---|---|---|
| Phase Offset | 2918 | 24.31 MHz |
| CP sup. + FFT | 3304 | 27.53 MHz |
| Interpolation & Equalization | 464 | 3.86 MHz |
| Data de-mapping | 13133 | 109.44 MHz |
| **Total** | 19819 | **165.16** $MHz$ |

Table 4.7 – Benchmark for the data symbols demodulation.

### 4.2.3 The GreenOFDM on GAP8

In the case of the GreenOFDM, we decided to implement some common functions with the MR-OFDM (the synchronization, the QPSK mapping, the convolutional encoder and decoder). Regarding the implementation of the GreenOFDM on GAP8 we have chosen the same values as in the previous chapter (ref. Table 3.1). Given $B = 250$ kHz, $T \approx 256\mu s$, $\frac{1}{4}$ samples for the CP, the symbol duration is hence $320\mu s$. For the transmitter, the profiled function was the GreenOFDM modulator by itself while for the receiver we profiled the synchronization and the symbol demodulation.

#### 4.2.3.1 The transmitter

The GreenOFDM symbol generation is by far the most computational demanding block of the transmitter. In Section 3.2.1.1 we have shown that 31 250 cycles are required to generate a satisfying GreenOFDM symbol. The required clock frequency must be of $\frac{31250}{320\times10^{-6}} = 97.65$ MHz in that case in order to guarantee real time. The system bandwidth, hence the data rate, could be increased since we are still not at the GAP8 limiting operating frequency (around 250 MHz).

#### 4.2.3.2 The receiver

The GreenOFDM receiver benchmark is summarized in Table 4.8. Here, three main stages can be seen (as for MR-OFDM): (i) the coarse synchronization, (ii) the fine synchronization and (iii) the symbols demodulation which includes phase offset correction, the GreenOFDM symbol demodulator [10] (FFT, de-mapping, error correcting decoder) and the channel equalization. The number of cycles were averaged over a symbol time duration of $320\mu s$. The obtained results show that a clock frequency lower than 50 MHz is sufficient to guarantee real-time operation at the receiver side.

| **Function** | **Avg # Cycles/Symbol** | **Clock Freq.** |
|---|---|---|
| Coarse synchronization | 12230 | 38.22 MHz |
| Fine synchronization | 12402 | 38.75 MHz |
| Phase Offset | 2956 | 46.1 MHz |
| Equalization | 443 | |
| Symbol Demodulation | 11355 | |

Table 4.8 – Benchmark for the GreenOFDM receiver.

## 4.3 Conclusion of the Chapter

In this chapter we have summarized the two standards that were implemented using the GreenWaves Technologies SAS GAP8 IoT processor.

For one of them, the 3GPP NB-IoT, we presented estimations that show that an implementation on GAP8 is impossible.

In the last section, we showed the profiling (number of cycles and required clock frequency) of the IEEE 802.15.4g MR-OFDM and the GreenOFDM that were implemented on GAP8.

10. Here, dedicated subcarriers were used in the transmitter side to send the so-called Side Information to de-randomize the data

In the case of MR-OFDM we saw that the GAP8 can run the transmitter with a clock frequency of 50 MHz and the receiver with a clock frequency of 165.16 MHz.

In the case of GreenOFDM, the GAP8 can run the transmitter with a clock frequency of 97.65 MHz and the receiver with a clock frequency of 50 MHz.

It is noteworthy that for GreenOFDM, the transmitter is more demanding than the receiver, contrarily to the case of MR-OFDM where the receiver is more demanding that the transmitter. This is mainly due to the required number of IFFTs in the GreenOFDM modulator.

# Conclusion and Perspectives

In this work, the OFDM modulation has been the central theme of research and engineering. A general description of the OFDM modulation was presented in chapter 1. Then in chapter 2, the problem of high PAPR in OFDM has been studied and a new technique has been proposed as an alternative to reduce the PAPR. This technique, called GreenOFDM, has yielded interesting results in terms of PAPR reduction compared to other techniques proposed in the literature.

Afterwards, motivated by the implementation of GreenOFDM in the GreenWaves Technologies SAS GAP8 IoT processor, we studied the computational charge of the proposed technique. In the first part of chapter 3 two computational charge reduction methods have been proposed. The results obtained from the proposed methods served to implement the GreenOFDM technique efficiently in the GAP8 IoT processor.

Later in the second part of chapter 3, intrigued by the question of energy consumption, the consumption of the GreenOFDM digital implementation in the processor GAP8 IoT processor has been studied. We have concluded that the energy consumption of this digital implementation is negligible compared to the energy consumption of the analog part of the transmitter, where the PA represents the most energy-consuming component.

Finally, we have compared the energy consumption of PA when transmitting GreenOFDM signals with respect to other multicarrier signals. The conclusion is that this energy comparison is equivalent to a comparison of CCDF(PAPR). As it has already been shown, the GreenOFDM provides interesting results in terms of CCDF(PAPR) reduction. These results are also translated to the energy consumption plane.

This relative energy efficiency of the GreenOFDM led us to consider that it can be implemented in different applications with low energy requirements (knowing that the benefits of OFDM are kept). Among the possible applications, the emerging (and slowly becoming omnipresent) IoT appears as one of the possible fields where GreenOFDM can provide its contributions, either for the uplink communications ('things' that send their data to the BS) or the downlink communications (*e.g.*, the BS that broadcast updated software to the edge IoT devices).

Since GreenOFDM can be considered as a communication scheme for the IoT, it has been decided to compare its energy consumption with one of the most popular techniques in the IoT, namely the LoRa modulation (that was described in the last part of chapter 1). The result is that GreenOFDM cannot beat LoRa in terms of energy consumption and the reason lies in the fact that LoRa is part of the so-called energy-efficient modulations while GreenOFDM is part of the so-called spectral-efficient modulations. GreenOFDM must be considered for situations in which the spectral-efficient modulations are required (the case of IEEE 802.15.4g MR-OFDM and the 3GPP NB-IoT).

Finally, for the engineering activities, details about the implementation on the GAP8 IoT processor of two studied communication schemes (the IEEE 802.15.4g MR-OFDM and the 3GPP NB-IoT) and of the GreenOFDM were presented in chapter 4.

## Main Contributions

The contributions of the thesis are summarized in order of appearance in the manuscript. There are three main contributions and they are summarized as follows:

(i) In the first chapter we contributed to the formalization of the LoRa modulation description. A digital and analog analysis was presented based on linear chirp signals. The orthogonality was studied and it was found that the analog LoRa symbols are not orthogonal. A possible reason for this non-orthogonality was given by going through a description of an overlapped Chirped-FSK scheme.

(ii) In the second chapter we contributed to the novel GreenOFDM technique [13, 14]. Based on the SLM-OFDM technique where the key point is to generate $C = U$ candidates through the computation of $U$ IFFTs, in GreenOFDM we changed the SLM-OFDM scheme to increase the number of candidates and hence generate $C = \frac{U^2}{4}$ candidates with the same number $U$ of IFFTs. Computer simulations of the CCDF($\gamma$) have shown that the GreenOFDM provides improvements compared to SLM-OFDM or SC-FDMA.

(iii) In chapter three we contributed to the GreenOFDM with IFFTs-on-demand method [43] which reduces the number of IFFTs to compute in a probabilistic manner. The distribution of the number of computed IFFTs $v$ was shown to vary as a function of the chosen PAPR threshold $\gamma_p$. As a complement, the hierarchical sampling method [44] was proposed to reduce the total number of samples to scan in candidates that have their PAPR above the threshold $\gamma_p$. An article presenting both methods has been submitted to Annals of Telecommunications, pending a decision.

## Perspectives

Based on what have been presented and the main contributions that are product of this work, perspectives for future works can be summarized as follows:

— The remarked non-orthogonality of the LoRa modulation impacts the performance on the receiver side if there is not a perfect synchronization. A perspective is to study what happens in such situations and what solutions can be proposed to overcome this non-orthogonality. In addition to this, a "TurboCSS" scheme that combines coding and modulation, inspired by [12] where very interesting results were obtained, can be proposed.

— In the thesis we have focused only on the transmitter side. Another perspective is to study the Bit Error Rate (BER) of the GreenOFDM in a multi-path wireless channel transmission and to determine how the Side Information (SI) errors impact the overall error rate performance.

Furthermore, since the SI degrades the GreenOFDM data rate compared to conventional OFDM, a GreenOFDM scheme with "Pilot Assisted Side Information" can be proposed with the study of its BER performance, as it is done in existing multiple signaling methods [41, 56, 57].
To complement this section of SI, another idea is to explore the plane of spatial diversity. One clue to study is a "MIMO-GreenOFDM" scheme and see how to increase the robustness of the SI with such a scheme.
On the other hand, another idea is to explore the inclusion of quaternions when randomizing information with PRN sequences. In particular, we would like to know if the SI can be implicitly transported and found in the receiver-side by exploiting the properties of quaternions.

— Finally, the combination of GreenOFDM with other PAPR reductions techniques can be studied, *e.g.,* apply Clipping (and Filtering) to the GreenOFDM symbols or to apply a sort of Green-SC-FDMA inspired from other SLM-SC-FDMA schemes [58, 59].

To conclude this section, we emphasize that the GreenOFDM symbols are OFDM symbols with the same properties (although with reduced PAPR). This means that GreenOFDM has the same benefits as conventional OFDM and can therefore be applied to other OFDM-based methods, *e.g.* Vectorized-OFDM, Filtered-OFDM, ZP-OFDM, hierarchical modulation (as used in DVB-T).

Appendix A

# Distribution of the number of IFFTs.

To obtain its probability mass function $\Pr\{V = v\}$ we split its study into two cases; (a) for $v$ even and, (b) for $v$ odd.

## (a) The probability $\Pr\{V = v\}$ for $v$ even

To attain $v$ computed IFFTs before finding an adequate candidate means that one of the $\frac{v}{2}$ available candidates after computing the $v^{\text{th}}$ IFFT has satisfied the condition. This requires that all the previous candidate waveforms have $\text{PAPR}_{g_1,g_2} > \gamma_{p_0}$. For example for $v = 4$ $\Pr\{V = 4\}$ is given by:

$$\begin{aligned}\Pr\{V=4\} =&\Pr\Big\{\Big(\text{PAPR}_{0,\frac{U}{2}} > \gamma_{p_0} \cap \text{PAPR}_{0,\frac{U}{2}+1} > \gamma_{p_0} \cap \text{PAPR}_{1,\frac{U}{2}+1} \leq \gamma_{p_0}\Big)\\ &\cup \Big(\text{PAPR}_{0,\frac{U}{2}} > \gamma_{p_0} \cap \text{PAPR}_{0,\frac{U}{2}+1} > \gamma_{p_0} \cap \text{PAPR}_{1,\frac{U}{2}+1} > \gamma_{p_0} \cap \text{PAPR}_{1,\frac{U}{2}} \leq \gamma_{p_0}\Big)\Big\}\end{aligned}$$

Assuming that the different waveforms are independent of each other, and using Eq.(2.3) we have [1]:

$$\Pr\{V=4\} \approx \left(\left(1-\left(1-e^{-\gamma_{p_0}}\right)^{2.8N}\right)^2 + \left(1-\left(1-e^{-\gamma_{p_0}}\right)^{2.8N}\right)^3\right).\left(1-e^{-\gamma_{p_0}}\right)^{2.8N}$$

This expression can be read as the probability to have computed $v = 4$ IFFTs to find the adequate candidate to be the sum of the probabilities of each available candidate $c$ (among $\frac{v}{2}$) with $c = 0, \cdots, \frac{v}{2} - 1$ that its PAPR is below the threshold while the previous $\frac{v}{2}.\left(\frac{v}{2}-1\right) + c$ candidate's PAPRs exceed the threshold, *i.e.*:

$$\Pr\{V = v_{\text{even}}\} = \left(1-e^{-\gamma_{p_0}}\right)^{2.8N}.\sum_{c=0}^{\frac{v}{2}-1}\left(1-\left(1-e^{-\gamma_{p_0}}\right)^{2.8N}\right)^{\frac{v}{2}\left(\frac{v}{2}-1\right)+c} \tag{A.1}$$

1.

$$\begin{aligned}\Pr\{V=4\} =&\Pr\left\{\text{PAPR}_{0,\frac{U}{2}} > \gamma_{p_0}\right\}\Pr\left\{\text{PAPR}_{0,\frac{U}{2}+1} > \gamma_{p_0}\right\}\Pr\left\{\text{PAPR}_{1,\frac{U}{2}+1} \leq \gamma_{p_0}\right\}\\ &+\Pr\left\{\text{PAPR}_{0,\frac{U}{2}} > \gamma_{p_0}\right\}\Pr\left\{\text{PAPR}_{0,\frac{U}{2}+1} > \gamma_{p_0}\right\}\Pr\left\{\text{PAPR}_{1,\frac{U}{2}+1} > \gamma_{p_0}\right\}\Pr\left\{\text{PAPR}_{1,\frac{U}{2}} \leq \gamma_{p_0}\right\}\\ \approx&\left(1-\left(1-e^{-\gamma_{p_0}}\right)^{2.8N}\right)^2\left(1-e^{-\gamma_{p_0}}\right)^{2.8N} + \left(1-\left(1-e^{-\gamma_{p_0}}\right)^{2.8N}\right)^3\left(1-e^{-\gamma_{p_0}}\right)^{2.8N}\end{aligned}$$

## (b) The probability $\Pr\{V = v\}$ for $v$ odd

Applying the same analysis for $v$ odd, we find that there are $\frac{v-1}{2}$ possibilities to attain the $v$ computed IFFTs. As before, each possibility occurs if all the previous candidate waveforms exceed the threshold $\gamma_{p_0}$ and the actual waveform satisfy the imposed condition $\text{PAPR}_{g_1,g_2} \leq \gamma_{p_0}$. The probability $\Pr\{V = v_{\text{odd}}\}$, under the same assumptions (different candidates are independent between them, and using Eq.(2.3)), is given by the sum of the probabilities for each possible candidate $c$ among $\frac{v-1}{2}$ with $c = 0, \cdots, \frac{v-1}{2} - 1$ that its PAPR is below the threshold while the previous $\frac{(v-1)^2}{4} + c$ candidate's PAPRs exceed the threshold, *i.e.*:

$$\Pr\{V = v_{\text{odd}}\} = \left(1 - e^{-\gamma_{p_0}}\right)^{2.8N} \cdot \sum_{c=0}^{\frac{v-1}{2}-1} \left(1 - \left(1 - e^{-\gamma_{p_0}}\right)^{2.8N}\right)^{\frac{(v-1)^2}{4}+c} \tag{A.2}$$

And hence $\Pr\{V = v\}$ reads:

$$\Pr\{V = v\} = \begin{cases} \left(1 - e^{-\gamma_{p_0}}\right)^{2.8N} \cdot \sum_{c=0}^{\frac{v}{2}-1} \left(1 - \left(1 - e^{-\gamma_{p_0}}\right)^{2.8N}\right)^{\frac{v}{2}\left(\frac{v}{2}-1\right)+c}, & \text{if } v \text{ even} \\ \left(1 - e^{-\gamma_{p_0}}\right)^{2.8N} \cdot \sum_{c=0}^{\frac{v-1}{2}-1} \left(1 - \left(1 - e^{-\gamma_{p_0}}\right)^{2.8N}\right)^{\frac{(v-1)^2}{4}+c}, & \text{if } v \text{ odd} \end{cases} \tag{A.3}$$

Appendix B

# NB-IoT details about profiling

The overview of the NB-IoT presented in Chapter 4 have shown that the most demanding element is the NPUSCH channel. We present here some of its details and some estimations about its implementation.

## B.1 The NPUSCH Channel

The NPUSCH supports, as previously mentioned, two modulation schemes: (i) single tone BPSK/QPSK when the PRB is composed of only one subcarrier (in this case the subcarrier spacing $\Delta f = 3.75$ kHz) and (ii) SC-FDMA otherwise (in this case $\Delta f = 15$ kHZ).

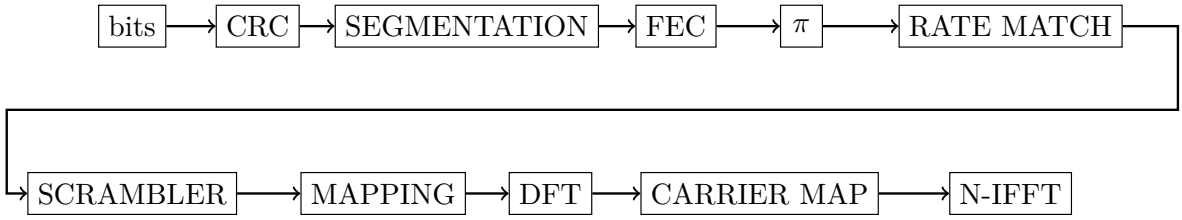



Figure B.1 – The NPUSCH block diagram.

The NB-IoT NPUSCH block diagram is depicted in Figure B.1. First a 24 bits CRC is calculated over the data payload bits by a predefined cyclic generator polynomial. Then the block of bits at the output of the CRC is segmented if the size is bigger than a threshold Z (Z = 6144 bits) and an additional CRC is calculated and attached to each segment. Channel coding and interleaving are the same as in LTE: Turbo Coding with coding rate $\frac{1}{3}$ (Interleaver and encoder constituent are well defined in TS 36.212 [54]). Finally interleaving and rate matching takes place.

Then data is scrambled using a 31-bits sequence. This sequence is initialized using two different m-sequences and an initial 31 bit value result of the combination of several parameters (the Radio Network Temporary Identifier, number of the slot, number of the radio frame, number of the Cell Identifier, etc).

The output of the scrambler is then mapped to BPSK/QPSK constellation. Then, in the case of SC-FDMA, data is spread by means of DFT (Discrete Fourier Transform) over groups

of constellation points equivalent to the number of assigned subcarriers. Prior N-IFFT, data is mapped to the subcarriers indexes assigned by the BS.

Finally, N-IFFT computes each time-domain symbol (with $N = 8192$ if only one subcarrier has been assigned and $N = 2048$ otherwise) and a Cyclic Prefix is prepended to each symbol prior concatenation in order to construct the signal to be transmitted.

## B.2 Estimation of the NB-IoT complexity on GAP8

The overview of the NB-IoT was not presented with more detail since it was not possible to be implemented in the processor version we had at the time (2018).

Indeed, some estimations of the NPUSCH only that ran on a virtual version of the processor (called GV-SoC for GAP8 Virtual System-on-Chip) showed to exceed the processor limits. The following table represents the obtained number of cycles and required clock frequency to maintain real time (symbols of $70\mu s$ duration) for the NPUSCH encoder turning on GAP8.

| **Function** | **Avg # Cycles** | **Clock Freq.** |
|---|---|---|
| Turbo Code | 151 | 2.16 MHz |
| Scrambler | 48 | 685.7 kHz |
| Mapper | 59 | 843 kHz |
| DFT spread | 1728 | 24.7 MHz |
| IFFT | 15928 | 228.5 MHz |
| **Total** | **17914** | **256 MHz** |

Table B.1 – Benchmark for the NPUSCH symbols with full RB allocation (12 subcarriers).

Conclusion for NB-IoT was that it was at the limit of what GAP8 could support. The goal however was to show that GAP8, whose main target are smart sensors algorithms like sound, image and/or vibration analysis, is flexible to handle also with some of the existing digital communication schemes. Something that we achieved to demonstrate with the implementation of the GreenOFDM and the MR-OFDM point-to-point communication.

# Bibliography


[1] J.G. Proakis and M. Salehi. *Digital Communications, 5th edition*. McGraw-Hill Higher Education, 2008. URL: https://books.google.fr/books?id=tOpKAQAACAAJ (cit. on p. 7).

[2] Inc Bell Communications Research et al. *Telecommunications Transmission Engineering*. Telecommunications Transmission Engineering vol. 2. Telcordia Technologies, 1990 (cit. on p. 11).

[3] S. Kay. *Intuitive Probability and Random Processes using MATLAB®*. Intuitive Probability and Random Processes Using MATLAB. Springer US, 2006. URL: https://books.google.fr/books?id=aSjX2tTpmFcC (cit. on p. 11).

[4] R. W. Chang. "Synthesis of band-limited orthogonal signals for multichannel data transmission". In: *The Bell System Technical Journal* 45.10 (1966), pp. 1775–1796 (cit. on p. 14).

[5] Semtech Corporation. *LoRa$^{TM}$ Modulation Basics*. AN1200.22 Rev2, available at www.semtech.com/uploads/documents/an1200.22.pdf. 2015 (cit. on pp. 19, 21).

[6] C. Goursaud and J. M. Gorce. "Dedicated networks for IoT: PHY / MAC state of the art and challenges". In: *EAI Endorsed Transactions on Internet of Things* 15.1 (Oct. 2015) (cit. on p. 19).

[7] L. Vangelista. "Frequency Shift Chirp Modulation: The LoRa Modulation". In: *IEEE Signal Processing Letters* 24.12 (2017), pp. 1818–1821 (cit. on p. 19).

[8] M. Winkler. "Chirp signals for communications". In: *IEEE WESCON Convention Record* 14.2 (1962) (cit. on p. 19).

[9] P. Courmontagne, G. Fages, and P. Beaujean. "A chirp FSK improvement for communications in shallow water using bandwidth overlapping". In: *OCEANS 2008*. 2008, pp. 1–7 (cit. on p. 24).

[10] R. Dutta et al. "Performance of chirped-FSK and chirped-PSK in the presence of partial-band interference". In: *2011 18th IEEE Symposium on Communications and Vehicular Technology in the Benelux (SCVT)*. 2011, pp. 1–6 (cit. on p. 24).

[11] J.G. Proakis and M. Salehi. *Digital Communications, 5th Edition*. 5th ed. McGraw-Hill, 2007 (cit. on p. 29).

[12] Y. Roth. "The physical layer for low power wide area networks : a study of combined modulation and coding associated with an iterative receiver". Thèse. Université Grenoble Alpes, July 2017. URL: https://hal.archives-ouvertes.fr/tel-01568794 (cit. on pp. 29, 94).

[13] D. Mestdagh, J. Cambonie, and J. Gulfo. "OFDM transmission systems with reduced Peak-to-Average Power Ratio". Patent WO/2017/013467 (FR). Jan. 2017. URL: patentscope.wipo.int/search/en/detail.jsf?docId=WO2017013467&_cid=P22-JY2IVR-09946-2 (cit. on pp. 31, 45, 94).

[14] D. J. G. Mestdagh, J.L. Gulfo Monsalve, and J.-M. Brossier. "GreenOFDM: a new selected mapping method for OFDM PAPR reduction". In: *IET Electronics Letters* 54.7 (2018), pp. 449–450 (cit. on pp. 31, 45, 58, 94).

[15] J. Palicot and Y. Louët. "Power Ratio definitions and analysis in single carrier modulations". In: *2005 13th European Signal Processing Conference*. 2005, pp. 1–4 (cit. on p. 32).

[16] C. Tellambura. "Computation of the continuous-time PAR of an OFDM signal with BPSK subcarriers". In: *IEEE Communications Letters* 5.5 (2001), pp. 185–187 (cit. on p. 35).

[17] R. van Nee and A. de Wild. "Reducing the peak-to-average power ratio of OFDM". In: *Vehicular Technology Conference, 1998. VTC 98. 48th IEEE*. Vol. 3. 1998, 2072–2076 vol.3 (cit. on p. 35).

[18] H. Ochiai and H. Imai. "On the distribution of the peak-to-average power ratio in OFDM signals". In: *IEEE Transactions on Communications* 49.2 (2001), pp. 282–289 (cit. on p. 35).

[19] Sanjay Singh, M.Sathish Kumar, and Mruthyunjaya H.s. "Effect of Peak-to-Average Power Ratio Reduction on the Multicarrier Communication System Performance Parameters". In: *International Journal of Electrical and Computer Engineering (IJECE)* 4 (Jan. 2009), pp. 779–786 (cit. on p. 36).

[20] Mikael Gustavsson, J. Jacob Wikner, and Nianxiong Nick Tan. *CMOS Data Converters for Communications*. Norwell, MA, USA: Kluwer Academic Publishers, 2000 (cit. on p. 36).

[21] Bernard Picinbono. "On Circularity". In: *IEEE Transactions on Signal Processing* 42.12 (1994), pp. 3473–3482 (cit. on p. 37).

[22] D.J.G Mestdagh, P.M.P. Spruyt, and B.B.F. Biran. "Analysis of clipping effect in DMT-based ADSL systems." In: *ANSI/T1E1.4/93-129* (1993) (cit. on p. 39).

[23] D.J.G. Mestdagh. "Reduction of Peak-to-Average-Power-Ratio of OFDM Systems: A Tutorial Overview". In: (2014). URL: `www.academia.edu/9538402/` (cit. on p. 40).

[24] Y. Rahmatallah and S. Mohan. "Peak-To-Average Power Ratio Reduction in OFDM Systems: A Survey And Taxonomy". In: *IEEE Communications Surveys Tutorials* 15.4 (2013), pp. 1567–1592 (cit. on p. 40).

[25] Xiaodong Li and L. J. Cimini. "Effects of clipping and filtering on the performance of OFDM". In: *1997 IEEE 47th Vehicular Technology Conference. Technology in Motion*. Vol. 3. 1997, 1634–1638 vol.3 (cit. on p. 40).

[26] J. Armstrong. "Peak-to-average power reduction for OFDM by repeated clipping and frequency domain filtering". In: *IET Electronics Letters* 38.5 (2002), pp. 246–247 (cit. on p. 40).

[27] Xiao Huang et al. "Companding transform for reduction in peak-to-average power ratio of OFDM signals". In: *IEEE Transactions on Wireless Communications* 3.6 (2004), pp. 2030–2039 (cit. on p. 40).

[28] D. J. G. Mestdagh and P. M. P. Spruyt. "A method to reduce the probability of clipping in DMT-based transceivers". In: *IEEE Transactions on Communications* 44.10 (1996), pp. 1234–1238 (cit. on pp. 40, 59).

[29] R. W. Bauml, R. F. H. Fischer, and J. B. Huber. "Reducing the peak-to-average power ratio of multicarrier modulation by selected mapping". In: *IET Electronics Letters* 32.22 (1996), pp. 2056–2057 (cit. on pp. 40, 42).

[30] Seung Hee Han, J. M. Cioffi, and Jae Hong Lee. "Tone injection with hexagonal constellation for peak-to-average power ratio reduction in OFDM". In: *IEEE Communications Letters* 10.9 (2006), pp. 646–648 (cit. on p. 40).

[31] J. Tellado and J. Cioffi. "Peak Power Reduction for Multicarrier Transmission". In: *IEEE Communication Theory Mini-Conference (GLOBECOM)*. 1998, pp. 219–224 (cit. on p. 40).

[32] L. Wang and C. Tellambura. "Analysis of Clipping Noise and Tone-Reservation Algorithms for Peak Reduction in OFDM Systems". In: *IEEE Transactions on Vehicular Technology* 57.3 (2008), pp. 1675–1694 (cit. on p. 40).

[33] B. S. Krongold and D. L. Jones. "PAR reduction in OFDM via active constellation extension". In: *IEEE Transactions on Broadcasting* 49.3 (2003), pp. 258–268 (cit. on p. 40).

[34] S. H. Muller and J. B. Huber. "OFDM with reduced peak-to-average power ratio by optimum combination of partial transmit sequences". In: *IET Electronics Letters* 33.5 (1997), pp. 368–369 (cit. on p. 40).

[35] N. Taspinar, A. Kalinli, and M. Yildirim. "Partial Transmit Sequences for PAPR Reduction Using Parallel Tabu Search Algorithm in OFDM Systems". In: *IEEE Communications Letters* 15.9 (2011), pp. 974–976 (cit. on p. 40).

[36] H. G. Myung, J. Lim, and D. J. Goodman. "Peak-To-Average Power Ratio of Single Carrier FDMA Signals with Pulse Shaping". In: *2006 IEEE 17th International Symposium on Personal, Indoor and Mobile Radio Communications*. 2006, pp. 1–5 (cit. on pp. 40, 44).

[37] H. G. Myung. "Introduction to single carrier FDMA". In: *2007 15th European Signal Processing Conference*. 2007, pp. 2144–2148 (cit. on p. 40).

[38] D. A. Wiegandt, C. R. Nassar, and Zhiqiang Wu. "Overcoming peak-to-average power ratio issues in OFDM via carrier-interferometry codes". In: *IEEE 54th Vehicular Technology Conference. VTC Fall 2001. Proceedings (Cat. No.01CH37211)*. Vol. 2. 2001, 660–663 vol.2 (cit. on p. 40).

[39] K. Anwar et al. "Simplified Realization of Pseudo-Orthogonal Carrier Interferometry OFDM by FFT Algorithm". In: *Multi-Carrier Spread-Spectrum*. Ed. by Khaled Fazel and Stefan Kaiser. Dordrecht: Springer Netherlands, 2006, pp. 167–174 (cit. on p. 40).

[40] G. T. Zhou and L. Peng. "Optimality Condition for Selected Mapping in OFDM". In: *IEEE Transactions on Signal Processing* 54.8 (2006), pp. 3159–3165 (cit. on p. 42).

[41] S. A. Adegbite, S. McMeekin, and B. G. Stewart. "Performance of a new joint PAPR reduction and SI estimation technique for pilot-aided SLM-OFDM systems". In: *2014 9th International Symposium on Communication Systems, Networks Digital Sign (CSNDSP)*. 2014, pp. 308–313 (cit. on p. 95).

[42] H. G. Myung et al. "Channel-Dependent Scheduling of an Uplink SC-FDMA System with Imperfect Channel Information". In: *2008 IEEE Wireless Communications and Networking Conference*. 2008, pp. 1860–1864 (cit. on p. 44).

[43] D. Mestdagh, J. Gulfo, and E. Flamand. "Method for reducing the number of operations for calculating the PAPR in selective mapping OFDM, using an IFFT on-demand procedure". Patent WO/2017/194985 (FR). Nov. 2017. URL: `patentscope.wipo.int/search/en/detail.jsf?docId=WO2017194985&_cid=P22-JY2IVR-09946-2` (cit. on pp. 51, 54, 94).

[44] D. Mestdagh, J. Gulfo, and E. Flamand. "Method for reducing the number of operations for calculating the PAPR in selective mapping OFDM, using hierarchichal sample evaluation". Patent WO/2017/194987 (FR). Nov. 2017. URL: `patentscope.wipo.int/search/en/detail.jsf?docId=WO2017194987&_cid=P22-JY2IVR-09946-2` (cit. on pp. 61, 94).

[45] Shuguang Cui, A. J. Goldsmith, and A. Bahai. "Energy-constrained modulation optimization". In: *IEEE Transactions on Wireless Communications* 4.5 (2005), pp. 2349–2360 (cit. on p. 66).

[46] GreenWaves Technologies SAS. *company website*. `www.greenwaves-technologies.com`. 2019 (cit. on p. 66).

[47] E. Flamand et al. "GAP-8: A RISC-V SoC for AI at the Edge of the IoT". In: *2018 IEEE 29th International Conference on Application-specific Systems, Architectures and Processors (ASAP)*. July 2018, pp. 1–4 (cit. on pp. 66, 82).

[48] S.C. Cripps. *RF Power Amplifiers for Wireless Communications*. Artech House microwave library. Artech House, 1999 (cit. on p. 67).

[49] R. J. Baxley and G. T. Zhou. "Power savings analysis of peak-to-average power ratio in OFDM". In: *IEEE Transactions on Consumer Electronics* 50.3 (2004), pp. 792–798 (cit. on p. 67).

[50] P. Arno. "Conception et caractérisation d'amplificateurs d'émission RF optimisés en rendement et linéarité pour applications en téléphonie mobile". PhD thesis. Grenoble-INP, 2006 (cit. on p. 67).

[51] P. Joshi et al. "Output Power Levels of 4G User Equipment and Implications on Realistic RF EMF Exposure Assessments". In: *IEEE Access* 5 (2017), pp. 4545–4550 (cit. on p. 67).

[52] H. T. Friis. "A Note on a Simple Transmission Formula". In: *Proceedings of the IRE* 34.5 (1946), pp. 254–256 (cit. on p. 70).

[53] "IEEE Standard for Local and Metropolitan area networks. Part 15.4: Low-Rate Wireless Personal Area Networks (LR-WPANs). Amendment 3: Physical Layer (PHY) Specifications for Low- Data-Rate, Wireless, Smart Metering Utility Networks". In: *IEEE-SA Standards Board* (2012) (cit. on pp. 76, 77, 79, 87).

[54] *3GPP Release 13: TS-36-211 TS-36-212 TS-36-213*. 2017 (cit. on pp. 81, 99).

[55] Y.-P. Eric Wang et al. "A Primer on 3GPP Narrowband Internet of Things (NB-IoT)". In: *CoRR* abs/1606.04171 (2016). URL: `http://arxiv.org/abs/1606.04171` (cit. on p. 81).

[56] A. T. Ho, J.F. Hélard, and Y. Nasser. "A novel combined PAPR reduction and channel estimation approach for OFDM systems". In: *2010 IEEE International Symposium on Broadband Multimedia Systems and Broadcasting (BMSB)*. 2010, pp. 1–5 (cit. on p. 95).

[57] R. J. Baxley, J. E. Kleider, and G. T. Zhou. "A Method for Joint Peak-to-Average Power Ratio Reduction and Synchronization in OFDM". In: *MILCOM 2007 - IEEE Military Communications Conference*. 2007, pp. 1–6 (cit. on p. 95).

[58] A. Khelil et al. "SLM localised SC-FDMA performance evaluation based on 30 GHz channel measurement for 5G". In: *IET Electronics Letters* 52.18 (2016), pp. 1573–1574 (cit. on p. 95).

[59] Imran Baig, Muhammad Ayaz, and Varun Jeoti. "A SLM based localized SC-FDMA uplink system with reduced PAPR for LTE-A". In: *Journal of King Saud University - Engineering Sciences* 25.2 (2013), pp. 119 –123 (cit. on p. 95).

**Résumé** — Ce travail est consacré à l'étude de la modulation OFDM et plus particulièrement au problème de son PAPR élevé.

Une solution pour la réduction de PAPR, appelée GreenOFDM, est proposée. Ses performances sont analysées et comparées avec d'autres techniques existantes dans la littérature, les résultats obtenus sont très prometteurs. La complexité calculatoire de cette technique est analysée en vue de sa mise en œuvre. Deux méthodes sont proposées pour réduire le nombre total d'opérations de la technique GreenOFDM ; leurs performances sont obtenues par simulation. Nous montrons comment il est possible de réduire considérablement le coût de calcul afin d'obtenir une implémentation numérique efficace. Enfin, pour démontrer cette efficacité, le coût énergétique de la mise en œuvre de GreenOFDM dans un processeur programmable est analysé et comparé à la consommation d'énergie de la partie analogique de l'émetteur. Une comparaison en termes de consommation d'énergie avec d'autres techniques de modulation est également menée à bien.

**Mots clés : OFDM, SLM, GreenOFDM, réduction de PAPR, réduction de la complexité calculatoire, amplificateur de puissance, consommation d'énergie, internet des objets.**

**Abstract** — This work is devoted to the study of the OFDM modulation and more particularly to its high PAPR problem.

A solution for the reduction of the PAPRs, called GreenOFDM, is proposed. Its performance is analyzed and compared with two other techniques available in the literature, the achieved performance of GreenOFDM is very promising. The computational complexity of this technique is analyzed in order to achieve an efficient implementation on a programmable processor. Two methods are proposed to reduce the total number of operations of the GreenOFDM technique; their performance is obtained by computer simulations. We show how it is possible to considerably reduce the number of operations and to obtain an efficient digital implementation. In fine, to demonstrate the efficiency, the energy cost of implementing GreenOFDM in a programmable processor is analyzed and compared to the energy consumption of the analog part of the transmitter. A comparison in terms of energy consumption with other modulation techniques is also carried out.

**Keywords: OFDM, SLM, GreenOFDM, PAPR reduction, Computational Complexity Reduction, Power Amplifier, Energy Consumption, IoT.**

Grenoble-INP GIPSA-lab, 11 Rue des Mathématiques
Saint-Martin-d'Hères