

Rapport Projet MAN

# 2ème Année Génie Systèmes Embarqués et Info. Indus

**CDMA**

**(Code Division Multiple Access)**

**Réalisé par :**

* **MOUHIYEDINE Chaymae**
* **TAYEK Fatima Zahra**

**Encadré par :**

* **Mr.BELKEBIR Hicham**

**Année Universitaire :** 2021/2022

**TABLE DES MATIERES**

1. [Introduction 3](#_bookmark0)
2. [Définitions 3](#_bookmark2)
3. [Modèle mathématique 9](#_bookmark3)
4. [Outil utilisé 12](#_bookmark3)
5. Travail réalisé 14
   1. [Implémentation du CDMA dans GNU Radio 14](#_bookmark1)
   2. Etapes de création de certains [blocs utilisés 14](#_bookmark2)
   3. [Blocs utilisés 15](#_bookmark4)
   4. Spreading Block (Bloc de l’étalement) 16
   5. Code Despreading without code tracking 16
   6. Decoding with code tracking 17
   7. Schéma réalisé dans GNU Radio 17
6. [Conclusion 19](#_bookmark5)
7. [Références 20](#_bookmark5)

# Introduction :

Les développements récents dans la conception de systèmes de communication numérique peuvent être principalement attribués aux algorithmes logiciels plutôt qu'une conception matérielle dédiée.

Les efforts pour conduire la mise en œuvre numérique à l'antenne (sans fil) et le câble (filaire) peut se résumer en SDR, Software Defined Radio. La radio logicielle réalise les problèmes matériels en les convertissant en problèmes logiciels.

Ce travail de notre projet vise à la mise en œuvre en temps réel du code division multiple

accès (CDMA) en SDR en utilisant GNU Radio..

# Définitions :

# SDR

# On assiste à une croissance exponentielle du nombre de façons et de moyens de

# communication entre les personnes entre eux. Ils utilisent par exemple la voix, la vidéo

# et les messages texte, la radio logicielle (SDR) offre un moyen rentable et flexible de

# faire progresser les communications. Elle offre de nombreux avantages pour les

# fournisseurs de services et les utilisateurs finaux.

# SDR est une radio dans laquelle les fonctions de la couche physique sont réalisées

# par logiciel. LE GROUPE IEEE P1900.1 a travaillé pour établir une définition et une

# vue d'ensemble du SDR. SDR (Software Defined Radio) est un ensemble de matériels et

# de logiciels où tout ou partie des fonctions radio sont mises en œuvre par le biais de

# logiciels ou de micrologiciels qui sont programmables.

# Ces dispositifs comprennent des FPGA (Field Programmable Gate Array), des DSP, des

# GPP ou d'autres processeurs programmables. Avec SDR, une nouvelle technologie

# peut être ajoutée à la technologie existante sans grand changement de matériel.

# La figure 1 montre l'architecture fonctionnelle générale de la radio logicielle.

# 

# GNU Radio utilisé dans ce projet est un type de SDR (Software Defined Radio).

## Figure 1: Architecture fonctionnelle générale de SDR

## CDMA :

## Grâce aux techniques d'accès multiple par étalement du spectre, il est possible de transmettre

## simultanément des signaux ayant la même largeur de bande RF (radiofréquence) sans aucune

## interférence. L'AMRC (accès multiple par répartition en code) est un type de technique

## d'étalement du spectre. Les systèmes CDMA utilisant des codes pour différents utilisateurs

## sont connus sous le nom de DS-CDMA (Direct Sequence Code Division Multiple Access).

## Dans ce système, s'il y a N utilisateurs, chaque utilisateur recevra son propre code, gi(t),

## où i=1,2,. ,N.

## En général, les codes utilisés sont orthogonaux, de sorte que la corrélation croisée entre deux

## codes est nulle.

## L'avantage d'utiliser le système CDMA est que tous les utilisateurs peuvent partager le spectre

## complet de manière asynchrone.

## L’accès multiple par répartition de code, ou Code Division Multiple Access (CDMA), est une

## technique de multiplexage définie comme étalement de spectre.Cette dernière était initialement

## destinée aux applications militaires.Elle permet, par l’étalement de la puissance sur une large

## bande de fréquence du canal, de mieux résister aux évanouissements (fading) sélectifs en

## fréquence et de donner au signal à transmettre la forme d’un bruit le rendant difficilement

## détectable par des récepteurs auquel le message n’est pas destiné.

## Un autre avantage est celui

## de la résistance que confère l’étalement de spectre aux brouilleurs pouvant apparaitre en cours

## de transmission.Pour le CDMA, l’utilisation de séquences d’étalement comme codes permettant

## de distinguer les différents utilisateurs donne, de plus, l’avantage d’exploiter simultanément

## l’ensemble de la bande de fréquence est des intervalles de temps. Il en résulte une meilleure

## gestion des ressources disponibles. Les conditions posées sur l’orthogonalité des séquences de

## code permettant de réduire les interférences entre utilisateurs.

## 

## Codes CDMA :

## Les codes CDMA sont la séquence d'étalement partagée par l'émetteur et le récepteur.

## Les données d'entrée sont multipliées par la séquence d'étalement.

## Le débit binaire de la séquence d'étalement est plus élevé que le débit binaire des

## données réelles.

## Lorsque le signal est reçu, il est synchronisé avec le code et multiplié par les codes pour supprimer l'étalement, ce processus est appelé désétalement. Les codes d'étalement doivent ressembler à du bruit. C'est-à-dire que les codes doivent comporter un nombre égal de uns et de zéros. Les codes d'étalement d'un système CDMA doivent avoir de bonnes propriétés de corrélation. Lorsque plusieurs signaux sont reçus, chacun avec un code d'étalement différent, le récepteur doit être capable de sélectionner le signal individuel utilisant ce code particulier. Le signal étalé ne doit pas être corrélé avec les autres signaux ; il doit se comporter comme du bruit et ne doit pas interférer avec le désétalement d'un signal particulier.

## L'étalement des signaux donne lieu à un niveau élevé de redondance qui permet au signal de faire face aux interférences des autres signaux dans la même bande passante.

## CDMA est une méthode d'accès à des canaux multiples obtenue par l'étalement d'un bit de données unique sur une séquence plus longue de bits transmis par l'utilisation de codes d'étalement. Il faut faire très attention en choisissant les séquences ou les codes d'étalement pour que les données puissent être désétalées correctement au niveau du récepteur.

## Par conséquent, la condition idéale pour ces codes d'étalement est qu'ils soient orthogonaux. L'orthogonalité peut être expliquée mathématiquement comme suit si n utilisateurs et des séquences de puces de n bits comme ensemble de vecteurs couvrent l'espace n, alors les séquences sont orthogonales si n'importe quel point de l'espace n est exprimé par point de l'espace n est exprimé par une seule combinaison linéaire des vecteurs de l'espace.

## En pratique, des vecteurs orthogonaux permettront un décodage correct des données transmises. La condition d'orthogonalité est très utile dans la partie suivi de code même si la définition semble un peu artificielle et ralentit l'estimation des résultats.

## L’auto-corrélation (dans quelle mesure un signal est similaire à sa version retardée ?) et la corrélation croisée (la mesure de similarité d'une forme d'onde avec une version retardée d'une forme d'onde différente) doivent être prises en compte.

## L'accès multiple par répartition en code (CDMA) se distingue fortement des techniques de multiplexage en temps et en fréquence.

## Les codes CDMA doivent avoir de bonnes propriétés de corrélation. Les différents types de codes CDMA sont : la séquence PN, M-Sequence Code, Gold Sequence Code, ZCZ Code.

## PN Sequence :

## Le scénario idéal pour avoir une séquence d'étalement serait une séquence aléatoire de zéros et de uns. L'émetteur et le récepteur doivent avoir la même séquence d'étalement. En d'autres termes, l'émetteur et le récepteur doivent disposer d'une méthode permettant de générer le même flux binaire à l'émetteur et au récepteur tout en conservant les propriétés de la séquence aléatoire. Ce travail est effectué par le générateur PN. Le générateur PN génère un code d'étalement avec une séquence aléatoire de zéros et de uns.

## La séquence PN est générée en prenant une certaine valeur initiale connue sous le nom de graine. L'algorithme ne produira pas une séquence aléatoire parfaite mais passera de nombreux tests d'aléa. De tels nombres sont appelés pseudo-bruit ou nombres pseudo-aléatoires. Il n'est pas possible de prédire la séquence si l'algorithme ou la graine est inconnu.

## Les propriétés importantes des séquences PN sont le caractère aléatoire et l'imprévisibilité. Afin de générer une séquence aléatoire, la séquence de nombres doit être aléatoire dans un certain sens statistique. Il existe deux critères permettant de valider une séquence de nombres comme étant aléatoire, à savoir la distribution uniforme et l'indépendance. La distribution uniforme signifie que la séquence de nombres doit être uniforme. C'est-à-dire que la fréquence d'apparition de chaque nombre doit être la même. La propriété d'indépendance signifie qu'aucune valeur ne peut être déduite des autres.

## 

## Figure 2: CDMA utilisant la séquence PN

## La figure 2 décrit la manière dont les systèmes CDMA fonctionnent en utilisant une séquence PN.

## M-Sequence Code :

## Le spectre étalé généré par un générateur PN composé de XOR (OU EXCLUSIF) et de registres à décalage LFSR (Linear Feedback Shift Register).

## LFSR est un dispositif de stockage à 1 bit composé d'une ligne de sortie qui donne la valeur du dispositif actuellement stocké et d'une ligne d'entrée. A un instant discret, la valeur dans le dispositif de stockage est remplacée par la valeur dans la ligne discrète. Ce qui provoque un décalage d'un bit dans l'ensemble du registre.

## Si LFSR contient n bits, il peut y avoir 1 à (n-1) portes. La présence ou l'absence d'une porte est représentée par la présence ou l'absence d'un terme polynomial. L'équation générale du LFSR par des termes XOR est représentée comme suit :

## Bn=A0B0 ⊕ A1B1 ⊕ A2B2 ⊕ ⊕An-1Bn-1

## Pour Ai=0, le circuit XOR correspondant sera supprimé. La figure 3 montre le registre à décalage à 4 bits pour l'équation B3= B0 ⊕B1. La séquence générée par LFSR a des avantages tels que la séquence est aléatoire avec de longues périodes. Ils peuvent être facilement implémentés en matériel à grande vitesse. La vitesse est importante car le taux d'étalement doit être supérieur au débit de données.

## On peut prouver que le LFSR a une période de N=2n-1. La séquence tout à zéro ne sera obtenue que si tous les coefficients sont nuls ou si le contenu initial du LFSR est nul.

## Ce LFSR génère toujours une séquence de période N et ces séquences sont appelées m-séquences (séquence de longueur maximale). Elles peuvent être utilisées dans le FHSS (Frequency Hopping Spread Spectrum) mais pas dans le système CDMA, en raison de leur faible corrélation croisée.

## 

## Figure 3: Mise en œuvre du registre à décalage

## Gold Sequence Codes :

## Les m-séquences (M-Sequence) ne peuvent pas être utilisées dans un système CDMA car les m-séquences (M-Sequence) pour chaque utilisateur n'ont pas une bonne propriété de corrélation croisée.

## Cependant, Gold Sequence satisfait à la propriété de corrélation croisée. Gold Sequence est construite par XOR de deux m-séquences à la même horloge. La figure 4 montre deux paires de registres à décalage générant deux m-séquences qui sont soumises à une opération XOR au niveau du bit.

## L’expression pour le LFSR en haut de la figure 4 est B4=B3⊕B0

## et l'expression pour le LFSR en bas est C4=C3⊕C2⊕C1⊕C0.

## La séquence m de ces deux LFSR est soumise à une opération XOR au niveau des bits pour générer Gold Sequence. La séquence résultante n'est pas maximale. Afin de concevoir un Gold Code souhaité des paires préférées de m-séquences sont utilisées,les paires préférées de m-séquences sont sélectionnées à partir de la table des paires ou générées à partir d’un algorithme.

## 

## Figure 4: Mise en œuvre des registres à décalage de Gold Sequence

## ZCZ Codes :

## Le système DS-CDMA idéal doit avoir des lobes latéraux d'autocorrélation très faibles et des lobes latéraux de corrélation croisée très faibles. Il est prouvé qu'il est impossible de développer une fonction d'autocorrélation impulsive idéale et une fonction de corrélation croisée nulle idéale. C'est-à-dire que l'auto-corrélation a un pic unique à t=0 et des valeurs nulles ou presque nulles à d'autres valeurs de t.

## Pour les codes d'auto-corrélation, l'amplitude sera nulle à toutes les valeurs de t, mais ce n'est pas possible et cela entraînera des interférences entre canaux dans les systèmes CDMA. Bien qu'il ne soit pas possible de développer une fonction de corrélation croisée nulle et une auto-corrélation impulsive, il est possible de développer une ZCZ (Zero Correlation Zone) dans l'auto-corrélation et la corrélation croisée. La ZACZ (Zero Auto-correlation Zone) est une zone où toutes les amplitudes de la fonction d'auto-corrélation seront nulles sauf à t=0 et en dehors de la zone il y aura une amplitude non nulle. De même, ZCCZ (Zero Cross Correlation Zone) est une zone où l'amplitude de la corrélation croisée est nulle dans une zone particulière et où l'amplitude n'est pas nulle en dehors de cette zone.

## Les figures 5 et 6 montrent respectivement la ZACZ (Zero Auto-correlation Zone) et la

## ZCCZ (Zero Cross Correlation Zone).

## Figure 5: Fonction d'autocorrélation et sa zone ZACZ

## (Zero Auto-correlation Zone)

## 

## Figure 6: Fonction de corrélation croisée et sa zone ZCCZ

## (Zero Cross Correlation Zone)

## Modèle mathématique :

## Emetteur :

## Soit bk() les données à transmettre. Les données à transmettre sont codées de manière différentielle comme suit

## Les données codées de manière différentielle sont ensuite étalées à l'aide du code

## d'étalement . Le code d'étalement utilisé dans ce système est un code ZCZ de longueur 64.

## 

## Les données étalées sont ensuite transmises au filtre RRC (Root Raised Cosine) pour la

## mise en forme des impulsions, puis le signal est transmis au modulateur.

## Le signal modulé en bande de base est alors transmis. Le signal transmis est représenté par

## (t) .

## La figure 2 illustre l'étalement des données et le spectre du signal correspondant dans la

## Figure 7.

## 

## Figure 7 : Spectre du signal CDMA

## Récepteur :

## Soit y(t) le signal en bande de base reçu. Il sera une combinaison du signal transmis,

## du bruit et du réponse impulsionnelle du canal.

## 

## où (t) = la réponse impulsionnelle du canal.

## n(t) = bruit

## (t) = le signal transmis

## L'acquisition du code est faite pour identifier le début d'un symbole, et doit être faite par le

## récepteur. Une fois que le début du symbole est identifié, le récepteur peut utiliser le code

## utilisé par l'émetteur pour effectuer le désétalement.

## Le désétalement est effectué en multipliant les données reçues par le code de l'utilisateur

## particulier. Les données multipliées sont intégrées et envoyées au décodeur différentiel.

## 

## Dans le décodeur différentiel, une opération différentielle et un décodage sont effectués

## pour identifier les données.

## Processus de décodage/encodage à spectre étalé CDMA :

## Pour visualiser le fonctionnement du système CDMA à étalement de spectre, la méthode la plus

## simple consiste à montrer par un exemple comment le système fonctionne sous la forme de bits

## de données et comment il est récupéré à partir d'un signal CDMA étalé.

## La partie initiale du système consiste à générer le signal CDMA. Prenons un exemple où 1001

## est la donnée qui doit être transmise, et 0010 est le code de l'étalement du signal.

## Nous avons utilisé la multiplication du code d'étalement entier aux données pour chaque bit de

## données.

## De la même manière, pour tous les bits de données, le signal étalé ou étendu comprend quatre

## bits, comme indiqué dans le tableau 1 suivant :

|  |  |  |
| --- | --- | --- |
| Données à transmettre | Code d’étalement | Signal étalé |
| 1 | 0010 | 1101 |
| 0 | 0010 | 0010 |
| 0 | 0010 | 0010 |
| 1 | 0010 | 1101 |

## Table 1 : Section Emetteur

## Le tableau 2 montre le signal reçu et le code d'étalement nécessaire pour le décoder du côté

## du récepteur.

|  |  |  |  |
| --- | --- | --- | --- |
| Signal reçu | Code d’étalement | Désétalement | Sortie |
| 11010010 |  | 1111 | 1 |
| 0010 | 0010 | 0000 | 0 |
| 0010 | 0010 | 0000 | 0 |
| 1101 | 0010 | 1111 | 1 |

## Table 2 : Section Récepteur NB : 1 ⊕ 1 = 0 1 ⊕ 0 = 1

## De cette manière, on peut voir que les données originales peuvent être récupérées précisément avec le code d'étalement.

## Si un autre code est utilisé pour décoder le signal étalé CDMA, on obtiendra une série aléatoire après le désétalement, qui apparaîtra comme du bruit dans le système.

## Le code d'étalement utilisé dans cet exemple a une longueur de seulement quatre bits, ce qui permet de visualiser plus clairement le processus.

## En général, les codes CDMA ont une longueur de 64 ou 128 bits pour fournir les performances requises.

## Systèmes CDMA à étalement par séquence directe :

## 

## Figure 8 : Spectre du signal CDMA

## La figure 8 nous donne un aperçu des systèmes CDMA à étalement par séquence directe. La procédure d'étalement se compose de deux étapes :

## 1. Codage différentiel

## 2. Filtre de mise en forme des impulsions

## La procédure de désétalement se compose de deux procédures principalement :

## 1. Acquisition du code (désétalement et intégration)

## 2. Décodage différentiel

## Le suivi de code est facultatif mais améliore considérablement la procédure de correction des erreurs grâce à un système de rétroaction.

## Outil utilisé :

## GNU Radio :

## *GNU Radio* est un logiciel libre et gratuit offrant des blocs de traitement de signal dans le but de développer de radios logicielles.

## Figure 9 : GNU Radio

## Il peut être utilisé avec du matériel RF externe peu coûteux pour créer des radios définies par logiciel ou sans matériel dans un environnement de type simulation. Il est largement utilisé dans les environnements amateurs, universitaires et commerciaux pour soutenir à la fois la recherche sur les communications sans fil et les systèmes de radio réels.

## Le système GNU Radio est composé de Python pour le front-end et de C++

## pour le back-end. Les implémentations du traitement du signal sont réalisées en C++.

## Les blocs de traitement du signal sont organisés et connectés par Python.

## Les attributs des blocs comprennent le nombre de ports d'entrée, le nombre de ports de sortie et le type de données qui circulent dans chaque bloc. Les types de données les plus couramment utilisés sont complexes, flottants, courts.

## .

## Figure 10 : GNU Radio avec matériel RF

## Cette figure 10 représente la mise en œuvre de GNU Radio en utilisant du matériel RF.

## Sur la figure 10, nous pouvons voir une antenne, RF, un ADC et du code mis en œuvre par GNU Radio. L'ADC (Analog to Digital Converter) fait le pont entre le signal continu en temps réel et le signal discret qui est manipulé dans le logiciel. Deux caractéristiques des ADC sont le taux d'échantillonnage et la gamme dynamique.

## La fréquence d'échantillonnage est la mesure du nombre de fois où le signal analogique est mesuré. La plage dynamique est la mesure de la différence entre le plus grand et le plus petit signal. La plage dynamique est fonction du nombre de bits dans la sortie du CAN (Convertisseur Analogique Numérique). Par exemple, si un CAN (Convertisseur Analogique Numérique) de 12 bits est utilisé, il représente 4096 niveaux, un CAN de 14 bits représente 16384 niveaux de signal.

## Travail réalisé :

## Implémentation du CDMA dans GNU Radio :

## Pour l'implémentation du CDMA dans GNU Radio, comme mentionné ci-dessus, la création

## de blocs est nécessaire. Ces blocs ainsi que d'autres blocs intégrés sont connectés d'une

## manière particulière en fonction des applications.

## Les données circulent à travers ces blocs de traitement de signal comme un flux de données,

## où la longueur du flux varie de temps en temps en fonction du planificateur, il sera donc très

## important de le gérer correctement.

## En général, les blocs de GNU Radio sont classe abstraite C++ et sont principalement de trois

## types, blocs généraux, blocs synchrones, blocs interpolateurs et blocs décimateurs.

## Etapes de création de certains blocs utilisés :

## Dans ce projet, les blocs de traitement du signal d'étalement, de désétalement

## et de code tracking sont créés à l'aide du script de gr-model.

## Les 8 étapes pour les créer sont les suivantes :

## Étape 1 : Télécharger gr-modtool.

## Étape 2 : Dans la variable d'environnement UNIX, ajoutez le chemin

## d'accès au dossier gr-modtool (home/Documents/gnuradio/grmodtool-master).

## Etape 3 : Allez dans le dossier GNU Radio pour créer des modules comme gr-spreading, gr-despreading, gr-codetracking. Tapez la commande : gr\_modtool new mod spreading.

## Cette étape va créer un nouveau module nommé gr\_spreading.

## Étape 4 : Allez dans le dossier gr-spreading.

## Étape 5 : Pour créer le bloc de traitement du signal (*zcz\_spreading),*

## tapez la commande gr\_modtool add -t interpolator zcz\_spreading, ceci afin de créer le bloc d'étalement hérité du bloc interpolateur.

## Etape 6 : Allez dans le dossier lib. Entrez dans le fichier zcz\_spreading\_impl.cc. Modifiez la fonction general\_work selon nos exigences.

## Étape 7 : Pour implémenter GRC pour le module, allez dans le dossier GRC et modifiez le fichier .xml en conséquence.

## Étape 8 : Pour exécuter le module, allez dans le dossier gr-spreading

## et donnez la commande cmake -i, sudo make. Corrigez les erreurs provenant

## pendant la compilation. Puis donnez sudo make install et sudo ldconfig.

## De même, nous pouvons également créer le bloc de désétalement.

## Blocs utilisés :

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Numéro du block | Nom du block | Type de sortie | Type de l’entrée | Description |
| 1. | Random Source | Integer | N/A (Sink) | Générer des données sources aléatoires |
| 2. | Int to Float | Float | Integer | Convertir un nombre entier en un nombre flottant |
| 3. | Float to Char | Character | Float | Convertir un nombre flottant en caractère |
| 4. | Differential Encoder | Character | Character | Codage différentiel en utilisant la sortie précédente |
| 5. | Char to Float | Float | Character | Convertir un caractère en un nombre flottant |
| 6. | Float to Complex | Complex | Re : FloatIm : Float | Convertir un nombre flottant en un nombre complexe en utilisant les deux entrées (Ici la partie imaginaire est connectée au null sink) |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 7. | ZCZ\_Spreading | Complex | Complex | Effectuer l'étalement de code zcz pour les données |
| 8. | Root Raised Cosine Filter | Complex | Complex | Réduire l’ISI |
| 9. | ZCZ\_Despreading | Complex | Complex | Acquisition du code avec le désétalement |
| 10. | WX GUI Scope | Float | N/A (Sink) | Afficher les données reçues |

## Concernant les blocks créés en utilisant gr\_modtool, on trouve :

## *ZCZ\_Spreading* : qui prend deux entrées arg et user qui définissent la taille

## du code ZCZ et quel code user est utilisé.

## *ZCZ\_despreading* : qui prend deux entrées arg et utilisateur qui définissent

## la taille du code ZCZ et le code utilisateur utilisé.

## Il effectue également l'acquisition du code.

## Spreading block :

## Le bloc d'étalement (*Spreading Block*) est un bloc de traitement du signal distinct.

## Ce bloc multiplie chaque donnée entrante avec un code d’étalement ZCZ 64.

## Chaque entrée est étalée sur 64 longueurs de code. Le code est également modifié en fonction du type d'utilisateur.

## L'utilisateur 1 et l'utilisateur 2 auront des codes différents.

## Le bloc d'étalement est hérité du bloc interpolateur.

## L'interpolateur fournit la fonctionnalité de 1:N. Pour chaque entrée, il y aura N sorties.

## Pour le bloc d'étalement, pour chaque entrée, il y aura 64 sorties puisque la longueur du code est de 64. Le rapport entre l'entrée et la sortie sera de 1:64.

## Une image contenant texte, horloge Description générée automatiquement

## Figure 11 : Codage différentiel et étalement

## Désétalement de code sans code tracking

## (Code Despreading without code tracking) :

## Le bloc de désétalement est un bloc de traitement du signal distinct dans lequel la fonction d'acquisition du code est également présente.

## Sur l'émetteur, le sur-échantillonnage est effectué au niveau du filtre RRC.

## Comme la longueur du code est de 64 et que le rapport de suréchantillonnage est de 4,

## la longueur du symbole est de 64 4 256 u . La position de départ est le début du premier symbole entièrement disponible dans le buffer à partir duquel les positions de

## départ ultérieures des symboles à venir peuvent être atteintes, qui se trouve à 256 données

## de la position d'acquisition actuelle

## La taille du buffer est nu u 64 4 où n est le nombre de symboles.

## Les données multipliées par 64 codes ZCZ sont intégrées et envoyées au bloc de traitement du signal suivant.

## Le bloc de désétalement hérite du bloc décimateur, de sorte que pour chaque 256 échantillons d'entrée, il n'y a qu'une seule sortie. Le rapport entre l'entrée et la sortie sera de (N :1).

## Décodage avec code tracking (Decoding with code tracking) :

## La position de départ du code établi doit être vérifiée comme à chaque fois et s'il y a

## une erreur, elle doit être corrigée. Ceci est fait par le code tracking. Le code tracking joue

## un rôle important dans le déplacement de la position d'acquisition du code. Ceci est illustré à la

## Figure 12

## Figure 12 : Décodage différentiel et désétalement

# Schéma réalisé dans GNU Radio :

# Après la création du bloc, la connexion est établie en utilisant différents blocs existants, à savoir, source aléatoire (Random Sourec) , DBPSK, filtre en cosinus surélevé (Root Raised Cosine filter) et bien d'autres encore.

# Le signal d'entrée utilisé est un signal aléatoire.

# 

# Ce travail utilise le code ZCZ d'une longueur de code de 64. Les codes doivent être mentionnés dans la partie codage C++ des blocs.

# Le code utilisé ici est [1, 1, -1, -1, -1, -1, -1, -1, -1, 1, -1, 1, 1, 1, -1 , -1, 1, 0, 0, 0, 0, 0,

# 0, 0, 0, 0, 0, 0, 0,0, 0, 0, 0, 0, 1, 1,-1, - 1, 1, 1, 1, 1, -1, 1,-1, 1, -1, 1, 1, -1, 0, 0, 0, 0, 0,

# 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0] .

# Les codes peuvent être n'importe quel code ternaire satisfaisant les conditions

# d'auto-corrélation et de corrélation croisée.

# Les blocs prennent deux entrées, à savoir arg : nombre de bits/symbole et utilisateur :

# nombre de bits/symbole.

# Le schéma de modulation utilisé est le DBPSK qui présente l'avantage d'une meilleure

# performance à des niveaux de bruit élevés.

# Le filtre à racine de cosinus surélevé est utilisé pour la mise en forme de l'impulsion

# du signal étalé et enfin le bloc scope sink est utilisé pour afficher le signal.

# La figure suivante montre la connexion expérimentale finale dans GNU Radio

# et les blocs mis en évidence sont ceux créés dans le cadre de ce travail.

# 

# Figure 13 : Schéma réalisé dans GNU Radio

# Conclusion :

# La mise en œuvre à faible coût du CDMA utilisant le code ZCZ pour une mise en

# œuvre en temps réel est réalisée à l'aide de GNU Radio.

# En intégrant l'USRP à GNU Radio, il est possible de visualiser les résultats en

# temps réel. En utilisant cette expérience, il a été possible d'analyser les données de

# manière interactive. Dans le prolongement de ce travail, il est possible d'essayer

# codes différents qui peuvent être essayés avec des signaux audio et vidéo en entrée.

# Une image contenant texte Description générée automatiquement

# Références :

# <https://depot-e.uqtr.ca/id/eprint/1886/1/030096577.pdf>

# <https://medium.com/@ayushbharadwaj74/software-defined-radio-350e9f791fee>

# <https://fr.theastrologypage.com/code-division-multiple-access>

# <https://cyberleninka.org/article/n/438935/viewer>