

الجمهورية الجزائرية الديمقراطية الشعبية  
**REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE**  
وزارة التعليم العالي والبحث العلمي  
**Ministère de l'Enseignement Supérieur et de la Recherche Scientifique**  
جامعة أبي بكر بلقايد - تلمسان  
Université Aboubakr Belkaïd- Tlemcen –  
Faculté de TECHNOLOGIE



**MEMOIRE**

Présenté pour l'obtention du **diplôme** de **MASTER**

**En** : Télécommunications

**Spécialité** : Réseaux et Télécommunications

**Par** : ARBOUZ Fedwa et ALLALI Khayra

**Sujet** :

*Etude et modélisation d'une chaîne de transmission DVB*

Soutenu publiquement, le **28 /06/ 2018** , devant le jury composé de :

Mr F.KARIM	MCA	Univ. Tlemcen	Président
Mr M.BOUSAHLA	MCB	Univ. Tlemcen	Encadreur
Mr F.DERRAZ	MCB	Univ. Tlemcen	Encadreur
Mr M.HADJILA	MCB	Univ. Tlemcen	Examineur

## **Remerciements**

*Nous tenons avant tout à remercier Dieu tout puissant de nous avoir donné la force et la volonté pour ce modeste travail.*

*Nous tenons tout d'abord à remercier nos deux encadreurs de mémoire, Mr M. BOUSAHLA et Mr F. DERRAZ qui ont suivi attentivement et activement la progression de notre travail.*

*Nos profonds respects et remerciements vont également au membre de jury Mr F. Karim et Mr M. Hadjila pour l'honneur qu'ils nous ont fait en acceptant de juger ce travail.*

*Nous remercions également tous nos collègues I Belmokrane, B Rahmi, M Gaad.*

## **Dédicace**

*Merci mes chers parents, ma mère et mon père, ma sœur Khadîdja, mes frères Fouzi, Fouad et Abdellahi, merci mes parents pour leur soutien et leurs encouragements pendant mes années d'études.*

*Je vous aime et son oublier Hocine et Abderrahmane.*

*Je remercierai ma famille qui m'a soutenu tout au long mes années d'étude.*

*Dédicace pour mes collègues de bureau de Mr Darrez : M benyagoub, M benyoub, K Allali, I Benmokrane, B Rahmi, I Ramoun, M Gaad, M Chbourou , et tous mes collègues de classe Master 2*

*RT.*

*Je tiens à remercier toutes les personnes qui m'ont soutenu, encouragé et ont contribué au bon déroulement de cette mémoire.*

*Arbouz Fedwa.*

## **Dédicace**

*Nous tenons à remercier le Dieu tout puissant et tous avoir donné la santé et la volonté*

*Je dédie ce mémoire.*

*A mes chers parents et ma mère et mon père.*

*Pour leur patience, leur amour, leur soutien, et leur encouragement.*

*A mon marie et ma belle mère et surtout mon fils Islem Mohamed.*

*A mes frère Mohamed et Rabah, et mes sœur sans oublier Rayan, Sirine, Inase, Nourhane,  
Amine, Riham, Nouria.*

*Dédicace pour mes collègues de bureau de Mr Darrez, et tous mes collègues de classe Master 2*

**RT.**

*Je tiens à remercier toutes les personnes qui m'ont soutenu, encouragé et ont contribué au bon déroulement de cette mémoire.*

*Allali Khayra.*

# TABLE DES MATIERES

<i>Remerciements</i> .....	2
<i>Dédicace</i> .....	3
<i>Dédicace</i> .....	4
<i>LISTE DES FIGURES</i> .....	8
<i>LISTE DES TABLEAUX</i> .....	10
<i>ABREVIATION</i> .....	1
Introduction Générale.....	1
Chapitre I : Etude d'une chaîne de transmission DVB.....	2
<b><i>I.1 INTRODUCTION</i></b> .....	<b>2</b>
<b><i>I.2 Présentation du standard DVB</i></b> .....	<b>2</b>
I.2.1 Codage source .....	2
I.2.2 Codage Canal.....	3
<b><i>I.3 Les sous standards de diffusion numérique</i></b> .....	<b>3</b>
I.3.1 Transmission par satellite DVB-S.....	4
I.3.1.1 Le Standard de DVB-S.....	4
I.3.1.1.1 Principaux paramètres d'une chaîne d'émission/réception DVB-S .....	5
I.3.1.1.1.1 Le codeur Reed Salomon.....	6
<b><i>I.3.1.1.1.2 L'entrelaceur</i></b> .....	<b>7</b>
<b><i>I.3.1.1.1.3 Le codeur convolutif</i></b> .....	<b>7</b>
I.3.1.2 Le Standard DVB-S2.....	8
I.3.1.2.1 Caractéristiques du DVB-S2 .....	9
I.3.1.2.1.1 Codage avancé.....	9
I.3.1.2.1.2 Modulation et codage Variable .....	9
<b><i>I.3.1.2.1.3 Modulation et codage adaptatif</i></b> .....	<b>10</b>
I.3.1.2.1.4 Chaîne de transmission DVB-S2.....	10
I.3.1.2.1.5 Bloc d'adaptation .....	11
I.3.1.2.1.6 Codeur FE.....	12
I.3.1.2.1.7 Organisation suivant le schéma de constellation.....	12
I.3.1.2.1.8 Bloc de formation des trames de la couche physique.....	13
I.3.1.2.1.9 La modulation .....	13
I.3.1.3 Etude comparative de DVB-S et DVB-S2.....	13
I.3.2 Le standard de diffusion numérique DVB-T .....	14
I.3.2.1 Le Standard DVB-T.....	14
I.3.2.1.1 Schéma bloc du standard DVB-T .....	15
I.3.2.1.3 Paramètres techniques du standard DVB-T .....	16
I.3.2.1.4 Évolution de la DVB-T à DVB-T2 .....	17
I.3.3 Le standard de diffusion numérique DVB-C .....	18
I.3.3.1 Le standard DVB-C.....	18
I.3.3.2 Les composants du système DVB-C.....	18
I.3.3.2.1 Codage de source et multiplexage MPEG-2 (MUX).....	18
I.3.3.2.2 Adaptation MUX et dispersion d'énergie .....	19
I.3.3.2.3 Encodeur externe .....	19
I.3.3.2.3 Entrelaceur externe .....	19
I.3.3.2.4 Conversion byte/m-tulpe .....	19
I.3.3.2.5 Codage différentiel: .....	19
I.3.3.2.6 Carte QAM.....	19
I.3.3.2.7 Mise en forme de la bande de base.....	19
I.3.3.2.8 DAC et front-end.....	20
I.3.3.3 Débits binaires disponibles pour un système DVB-C (Mbit/s).....	20
I.3.3.4 Évolution de la DVB-C à DVB-C2 .....	20

I.3.4	Le standard de diffusion numérique DVB-H.....	21
I.3.4.1	Le standard de DVB-H.....	21
I.3.4.2	Les composants du système DVB-H.....	22
I.3.4.2.1	La source de service .....	22
I.3.4.2.2	Encodeur.....	23
I.3.4.2.3	Serveur ESG et FLUTE.....	23
I.3.4.2.4	PROCURATION .....	23
I.3.4.2.5	ENCAPSULATEUR IP .....	23
I.3.4.2.6	Modulateur DVB.....	23
I.3.4.2.7	AMPLIFICATEUR.....	23
I.3.4.2.8	ANTENNE DE TRANSMISSION.....	23
I.3.4.2.9	DESTINATAIRE .....	23
I.3.4.3	La diffusion par les technologies DVB-T et DVB-H.....	24
<b>I.4</b>	<b>Conclusion.....</b>	<b>24</b>
Chapitre II	: Introduction des techniques de modulation et codage .....	25
<b>II.1</b>	<b>Introduction .....</b>	<b>25</b>
<b>II.2</b>	<b>La modulation OFDM.....</b>	<b>25</b>
II.2.1	Principe de l'OFDM .....	25
II.2.2	L'Orthogonalité .....	25
II.2.3	Intervalle de garde.....	26
<b>II.3</b>	<b>Codage par Reed-Salomon .....</b>	<b>27</b>
II.3.1	Définition.....	27
II.3.2	champ de Galois.....	27
II.3.2.1	Propriétés d'un corps fini .....	27
II.3.2.2	Construction d'un corps fini .....	28
II.3.2.2.1	Représentation des éléments .....	28
II.3.3	Technique de codage .....	29
II.3.4	Technique de décodage.....	30
<b>II.4</b>	<b>MPEG.....</b>	<b>31</b>
II.4.1	Définition de MPEG.....	31
II.4.2	Le développement de MPEG/ MPEG 2 .....	31
II.4.2.1	MPEG-1.....	31
II.4.2.3	MPEG-4.....	31
<b>II.5</b>	<b>Codage LDPC .....</b>	<b>32</b>
II.5.1	définition.....	32
II.5.2	Codes LDPC.....	32
II.5.3	Décodage des codes LDPC .....	36
<b>II.6</b>	<b>BCH.....</b>	<b>37</b>
II.6.1.	Caractéristiques générales de codes BCH.....	37
II.6.2	Définition.....	38
II.6.3	Construction.....	38
<b>II.7</b>	<b>conclusion.....</b>	<b>40</b>
CHAPITRE III	: Programmation de RASPBERRY Pi à l'aide de Simulink .....	41
<b>III.1</b>	<b>Introduction.....</b>	<b>41</b>
<b>III.2</b>	<b>Description de Raspberry pi.....</b>	<b>41</b>
<b>III.3</b>	<b>L'historique de Raspberry pi.....</b>	<b>42</b>
<b>III.4</b>	<b>Composants de base.....</b>	<b>47</b>
III.4.1	Ports USB.....	47
III.4.2	Port Ethernet.....	47
III.4.3	Connecteur audio .....	47
III.4.4	GPIO .....	48
III.4.5	Fente pour carte MicroSD.....	48

III.4.6	Port HDMI .....	48
III.4.7	Puissance.....	49
III.4.8	Le connecteur d'interface série de caméra (CSI).....	49
III.4.9	Port DSI display .....	49
<b>III.5</b>	<b>Programmation de Raspberry Pi à l'aide de Simulink.....</b>	<b>49</b>
III.5.1	Configuration de Package de support Simulink pour le Raspberry Pi.....	50
III.5.2	L'installation du Support Package .....	50
<b>III.6</b>	<b>Teste de fonctionnement de la carte Raspberry Pi + Simulink.....</b>	<b>53</b>
III.6.1	Vérification de l'installation du Support Package .....	53
III.6.2	Test de la connectivité (PC à Raspberry Pi) .....	54
<b>III.7</b>	<b>Conclusion.....</b>	<b>55</b>
Chapitre IV:	Simulations et Résultats .....	55
<b>IV.1</b>	<b>Introduction.....</b>	<b>55</b>
<b>IV.2</b>	<b>Modèle Simulink de la chaine de transmission DVB-S2.....</b>	<b>55</b>
IV.2.1	Générateur Bernoulli.....	56
IV.2.2	BBFRAME .....	56
IV.2.3	Décodeur/Encodeur BCH.....	58
IV.2.4	Encodeur/décodeur LDPC .....	59
IV.2.5	General bloc interleaver : .....	61
IV.2.6	Bloc modulateur.....	61
IV.2.6	Canal AWGN.....	63
<b>IV.3</b>	<b>Résultats de Simulation de la chaine DVB-S2 sous simulink : .....</b>	<b>64</b>
<b>IV.4</b>	<b>Simulation de la chaine de transmission DVB-S2 sous Matlab.....</b>	<b>66</b>
<b>IV.5</b>	<b>Conception et simulation de la chaine de transmission DVB-S2.....</b>	<b>67</b>
IV.5.1	Conception et simulation de la chaine d'émission DVB-S2 .....	67
IV.5.1.1	Conception et Simulation de la chaine d'émission DVB-S2 sous simulink	68
IV.5.1.1	Sur-échantillonnage (Upsampling) : .....	68
IV.5.1.2	Bande de base vers la fréquence intermédiaire (BB to IF) .....	70
IV.5.1.3	Contrôle Automatique de Gain (AGC) : .....	71
IV.5.1.2	Implémentation de la chaine d'émission DVB-S2 sur la carte Respberry	73
IV.5.2	Conception et simulation de la chaine de réception DVB-S2.....	73
IV.5.2.1	Conception et simulation de la chaine de réception DVB-S2 sous	73
IV.5.2.1	simulink.....	73
IV.5.1.3	Implémentation de la chaine de réception DVB-S2 sur la carte	74
IV.5.1.3	Respberry Pi.....	74
<b>IV.6</b>	<b>Conclusion :.....</b>	<b>74</b>
conclusion generale	.....	76
Références et documentation.....		77
Résumé .....		78

## LISTE DES FIGURES

Figure I.1 : Schéma symptomatique d'une chaîne d'émission TV numérique.....	2
Figure I.2 : LES SUPPORTS DE DIFFUSION NUMERIQUE .....	3
Figure I.3 : LA CHAINE DE TRANSMISSION DVB-S .....	5
Figure I.4 : Mot-code de Reed Salomon .....	7
Figure I.5 : Le principe général du codage convolutif .....	7
Figure I.6 : La modulation QPSK.....	8
Figure I.7 : Système de transmission DVB-S2 .....	10
Figure I.8 : Bloc d'adaptation .....	11
Figure I.9 : Trame BBFRAME.....	12
Figure I.10 : Schéma de constellation 16-APSK .....	13
Figure I.11 : Schéma à blocs fonctionnels du standard DVB-T. [4] .....	15
Figure I.12 : SYSTEME DE TRANSMISSION DVB-C.....	18
Figure I.13 : Système de transmission DVB-H.....	22
<i>Figure II.1 : EFFET DE CANAL SUR LA TECHNIQUE OFDM .....</i>	<i>25</i>
<i>Figure II.2 : Spectre en sortie du modulateur OFDM .....</i>	<i>26</i>
<i>Figure II.3 : Intervalle de garde .....</i>	<i>26</i>
<i>Figure II.4 : Graphe factoriel d'un code LDPC .....</i>	<i>33</i>
<i>Figure II.5 : Exemple de cycles de longueur 4, 6 et 8.....</i>	<i>35</i>
<i>Figure III.1 : Représentation de la fonction <math>f(x)</math>.....</i>	<i>41</i>
<i>Figure III.2 : Les portes de Raspberry Pi .....</i>	<i>44</i>
<i>Figure III.3 : LES MODELES DE RASPBERRY PI.....</i>	<i>45</i>
<i>Figure III.4 : Les information pour Raspberry Pi.....</i>	<i>54</i>
<i>Figure IV.1 : Modèle Simulink.....</i>	<i>55</i>
<i>Figure IV.2 : les composants du bloc de bbframe .....</i>	<i>56</i>
<i>Figure IV.3 : Coniguration du bloc de générateur bernolli .....</i>	<i>56</i>
<i>Figure IV.4 : Configuration des parametres de bloc buffer .....</i>	<i>57</i>
<i>Figure IV.5 : CONFIGURATION DES PARAMETRES DU BLOC DE ZERO PAD .....</i>	<i>57</i>
<i>Figure IV.6 : CONFIGURATION DES PARAMETRES DU BLOC DE CODEUR BCH ..</i>	<i>58</i>
<i>Figure IV.7 : Configuration des parametres du bloc de décodeur BCH.....</i>	<i>59</i>
<i>Figure IV.8 : Configuration des parametres du bloc de codeur LDPC .....</i>	<i>60</i>
<i>Figure IV.9 : Configuration des parametres du bloc de décodeur LDPC.....</i>	<i>61</i>
<i>Figure IV.10 : Configuration des parametres du bloc d'interleaver.....</i>	<i>61</i>
<i>Figure IV.11 : Composant de bloc modulateur .....</i>	<i>62</i>

<i>Figure IV.12 : c</i> Configuration des parametres du bloc modulation QPSK.....	63
<i>Figure IV.13 :</i> Configuration des parametres du bloc AWGN.....	63
<i>Figure IV.14 :</i> Constellation de la QPSK à l'émission.....	64
<i>Figure IV.15 :</i> Constellation de la QPSK à la réception.....	64
<i>Figure IV.16 :</i> Constellation de la QPSK à la réception.....	65
<i>Figure IV.17 :</i> Constellation de la 8-PSK à l'émission.....	65
<i>Figure IV.18 :</i> Constellation de la 8-PSK à la réception.....	66
<i>Figure IV.19 :</i> Constellation de la QPSK à l'émission.....	66
<i>Figure IV.20 :</i> Constellation de la QPSK à la réception.....	67
<i>Figure IV.21 :</i> Le spectre du signal DVB-S2.....	67
<i>Figure IV.22 :</i> Schéma bloc de emission de DVB-s2.....	68
<i>Figure IV.23:</i> Bloc Upsampling.....	69
<i>Figure IV.24 :</i> Filtrage après sur échantillonnage par un facteur 2 .....	69
<i>Figure IV.25 :</i> Réglage des filtres.....	70
<i>Figure IV.26 :</i> Bloc BB to IF .....	71
<i>Figure IV.27 :</i> Fenêtre de configuration pour le bloc cosine wave et le bloc sine wave.....	71
<i>Figure IV.28 :</i> Bloc AGC.....	72
<i>Figure IV.29 :</i> Blocs Max et Divide.....	72
<i>Figure IV.30 :</i> Schéma bloc de la chaine d'émission DVB-S2 .....	73
<i>Figure IV.31 :</i> resultat de simulation sous simulink .....	73
<i>Figure IV. 32 :</i> Schéma bloc de la chaine de réception DVB-S.....	74

## **LISTE DES TABLEAUX**

Tableau I.1 : Principaux paramètres d'une chaîne DVB-S .....	6
Tableau I.2 : Evolution du standard de communication par satellite. ....	9
Tableau I.3 : Modulation et taux de codage utilisés .....	10
Tableau I.4 : Type de données pour un système de transmission DVB-S2.....	11
Tableau I.5 : Les paramètres de comparaison du DVB-S et DVB-S2 .....	14
Tableau I.6 : Mode 2k de standard DVB-T .....	16
Tableau I.7 : Mode 8k de standard DVB-T. ....	17
Tableau I.8 : Débits binaires disponibles .....	20
Tableau I.9 : LA DIFFUSION DES TECHNOLOGIE DVB-T ET DVB-H.....	24
Tableau III.9 : les caractéristiques des modèles raspberry.....	46

## ABBREVIATION

- A/D Analog/ Digital
- ACM Adaptive Coding and Modulation.
- AWGN Additive White Gaussian Noise, bruit blanc additif gaussien
- BBFRAME Base Band Frame.
- BCH Bose ChaudhuriHocquenghem.
- COFD Coded Orthogonal Frequency Division Multiplex.
- CP Cyclic-Prefix (CP-OFDM).
- CRC-8 CyclicRedundancy Check.
- DF Data Field.
- DSNG Digital Satellite News Gathering.
- DVB Digital VideoBroadcasting.
- DVB-C Digital VideoBroadcasting Cable.
- DVB-C2 Digital Vidéo Broadcasting - Câble 2nd génération
- DVB-H Digital Video Broadcasting Handheld.
- DVB-NGH NextGenerationHandheld.
- DVB-S Digital Video Broadcasting satellite.
- DVB-S2 Digital Vidéo Broadcasting - Satellite 2nd génération.
- DVB-SH Satellite to Handhelds.
- DVB-T Digital VideoBroadcastingTerrestrial.
- DVB-T2 Digital Vidéo Broadcasting - Terrestrial 2nd génération
- ETSI European Télécommunications Standards Institute.
- FEC ForwardError Correction.
- LDPC LowDensityParity Check.
- MPEG Moving Picture Experts Group.
- OFDM Orthogonal frequency-division multiplexing
- QAM Quadrature Amplitude Modulation, modulation de phase et d'amplitude
- QPSK Quadrature Phase Shift Keying.
- RS Reed-SOLOMON.
- SOC Système On Chip.
- TEB Bit Error Rate, Taux d'Erreur Binaire .

- Tg Temps de guard.
- TNT Télévision Numérique Terrestre
- Ts Temps de symbole.
- UPL User PacketLength.
- USB Universal Serial Bus.
- ZP Zero-Padding (ZP-OFDM).

# Introduction général

## INTRODUCTION GENERALE

La télévision numérique fait partie de notre quotidien. Actuellement, il existe plusieurs possibilités pour recevoir cette télévision. Ces systèmes ont été développés par le groupe DVB.

Il existe plusieurs normes de la DVB, les plus importantes sont les normes DVB-T (terrestre) et DVB-C (câble), DVB-S (par satellite). Malgré cela, les besoins des utilisateurs évoluent constamment dans le domaine de l'éducation, de la formation, de communication et de divertissement. C'est pour cette raison qu'il existe de nouveaux défis que les normes ne sont pas en mesure de satisfaire. De plus, elle exige une révision constante des normes DVB.

Dans ce projet de master nous nous intéressons à la conception et à la simulation d'une chaîne de transmission DVB. Dans le premier chapitre, nous allons présenter la chaîne de transmission DVB ainsi que les standards existants de diffusion de la télévision numérique à savoir DVB-S, DVB-T, DVB-C et DVB-H.

Dans le deuxième chapitre nous allons présenter les notions importantes de la théorie des codes RS, LDPC, et le codage BCH. Nous avons dans un premier temps présenté les caractéristiques générales de ces codes, puis nous avons donné la façon de les construire.

Dans le troisième chapitre nous allons présenter un aperçu général de la carte Raspberry Pi, ainsi qu'un historique sur le développement de ces modèles de carte. Ensuite nous avons présenté les composants de base de la carte Raspberry Pi. Enfin nous avons présenté toute les étapes de l'installation du support package de simulink pour le Raspberry Pi et la méthode de vérification de la connectivité cette carte avec le PC.

Dans le quatrième chapitre, nous allons simuler la chaîne de transmission DVB-S2 sous simulink. Cette simulation est importante pour déterminer les performances de cette chaîne. Ensuite nous allons faire la conception de la chaîne d'émission seule sous simulink en tenant compte du bloc Raspberry. La simulation sera faite en mode normal et external. Une fois que le modèle de la chaîne d'émission sera vérifié, nous l'implémenterons sur la carte Raspberry Pi. Le même travail sera fait avec la chaîne de réception.

# Chapitre I

Etude d'une chaine de  
transmission DVB

## CHAPITRE I : ETUDE D'UNE CHAÎNE DE TRANSMISSION DVB

### I.1 INTRODUCTION

Dans ce chapitre, nous allons présenter la chaîne de transmission DVB ainsi que les standards existants de diffusion de la télévision numérique à savoir DVB-S, DVB-T, DVB-C et DVB-H.

### I.2 Présentation du standard DVB

Les différents blocs de la chaîne de transmission sont décrits dans la figure I.1

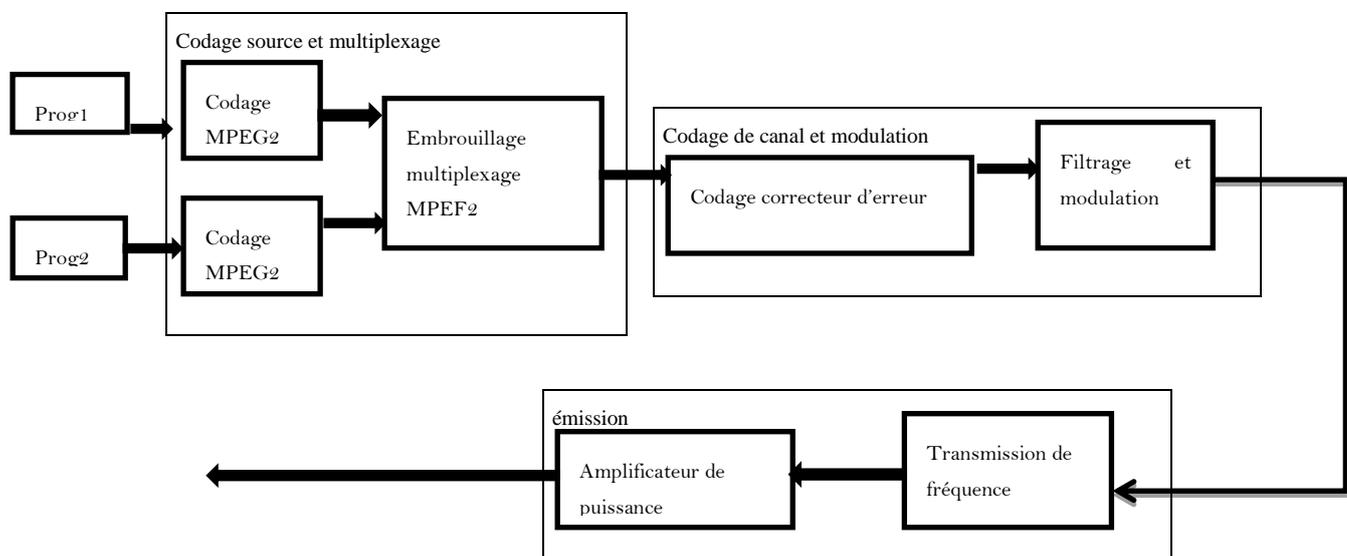


Figure I.1: SCHEMA SYNOPTIQUE D'UNE CHAÎNE D'ÉMISSION TV NUMÉRIQUE

#### I.2.1 Codage source

Le codage source a pour le but de réduire la quantité d'information transmise. Pour l'image, le codage source fait intervenir en matière de compression de signaux, le découpage de l'image en blocs, la prédiction temporelle et la compensation du mouvement, la quantification et le Codage à longueur variable. La norme prend en compte le balayage entrelacé propre au système de TV et permet un débit entre 2Mbps et 20 Mbps. Dans le domaine son, le codage peut s'effectuer en monophonie ou en stéréophonie selon le

# Chapitre I : Etude d'une chaîne de transmission de DVB

cas en utilisant plusieurs fréquences d'échantillonnage. Nous pouvons simultanément coder Plusieurs voies. [1]

## I.2.2 Codage Canal

Les opérations de codage de canal ont pour objet de préparer le signal avant l'émission. Elles comprennent essentiellement la dispersion d'énergie (brassage), le codage correcteur d'erreur et l'entrelacement. Les signaux codent des différents composants (sons, données et images) de plusieurs programmes, sont alors multiplexés. Le signal binaire résultant, transporte toutes les Informations de synchronisation des différentes composantes des programmes. L'entrelacement est une opération dont le but est de rendre le signal le plus aléatoire possible, l'intérêt est d'éviter les longues suites des « zéros » ou des « uns » qui créent une raie à forte énergie dans le spectre. Pour le codage correcteur d'erreur, plusieurs algorithmes sont disponibles suivant l'application. Parmi les codes correcteurs d'erreur on trouve le codage RS (Reed-SALOMON), le codage LDPC (Low Density Parity Check). Pour des raisons techniques ou de rentabilité, l'acheminement d'une information numérique, ne peut pas toujours se faire en bande de base. L'utilisation d'une fréquence porteuse est alors nécessaire. La modulation est l'opération qui fait correspondre à chaque niveau du signal numérique, un état d'amplitude, de fréquence ou de phase d'une onde porteuse. Le choix d'une modulation numérique dépend de : L'occupation spectrale, La résistance aux distorsions et aux diverses perturbations et La simplicité de réalisation des systèmes de modulation et de démodulation. [1]

## I.3 Les sous standards de diffusion numérique

Ces standards de diffusion numérique pour la télévision se déclinent en fonction des supports de diffusion. Plusieurs sous standards ont été définis : le DVB-S pour la diffusion par satellite, le DVB-C (DVB digital câble delivery system) pour la diffusion par câble, le DVB-T (DVB terres trial transmission standard) pour la diffusion terrestre. [2]

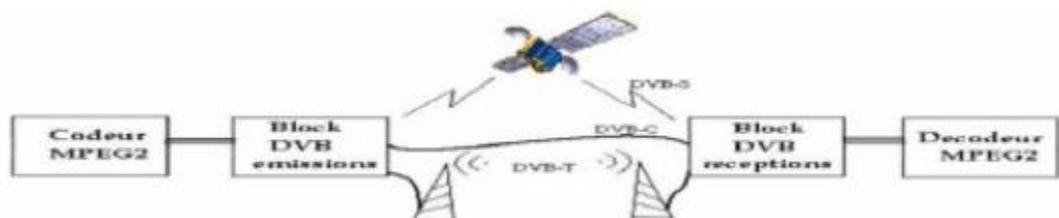


Figure I.1 : LES SUPPORTS DE DIFFUSION NUMERIQUE

### I.3.1 Transmission par satellite DVB-S

#### I.3.1.1 Le Standard de DVB-S

Le DVB-S est un standard défini pour la transmission par satellite. Ce standard a attiré l'intérêt des opérateurs et des chercheurs dans le domaine de communications en vue de sa bande large et des contraintes non strictes par rapports aux autres supports de transmissions. Le DVB-S a pris un succès depuis 1994 et les émissions ont débuté depuis 1996 pour les premiers opérateurs commerciaux. C'est un standard de diffusion relativement simple qui utilise la modulation QPSK (Quaternary Phase Shift Keying). Il utilise des canaux relativement larges (33 ou 36 MHz). Un code correcteur d'erreurs interne, dit de VITERBI est utilisé pour corriger les effets négatifs de la réception par satellite. Cette norme tient compte des caractéristiques d'une transmission satellitaire : La bande disponible est relativement large : de 26 à 36 MHz, Le canal est de type AWGN (Additive White Gaussian Noise), Le signal est fortement atténué et dominé par le bruit, La transmission est en ligne directe. Une deuxième version de DVB-S a été proposée en 1997, c'est le standard DVB-DSNG (DVB, Digital Satellite News Gathering). Il introduit en plus, la modulation 8PSK et 16QAM (16 Quadrature Amplitude Modulation) et les services payants. Mais avec la progression technologique et les nouvelles exigences, le DVB-S et le DVB-DSNG n'arrivent pas à satisfaire beaucoup les nouveaux besoins. D'où la nécessité d'un nouveau standard qui soit plus flexible et plus performant. Le DVB-S2 est le nouveau standard qui vient pour répondre à ces besoins actuels dans le domaine de communication. [1]

Le schéma de codage du canal du DVB-S est très particulier et il porte le nom de schéma de codage concaténé :

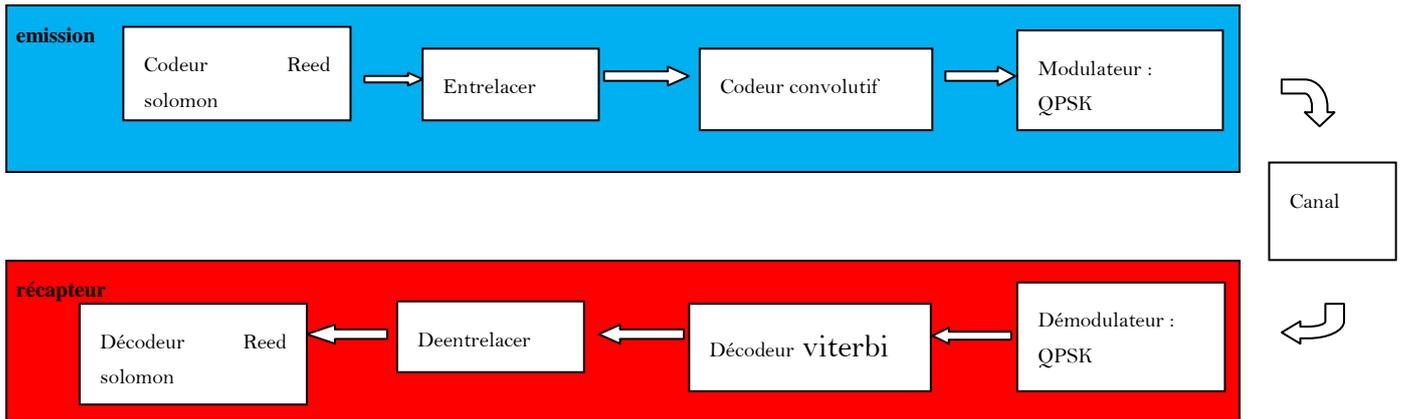


Figure I.2 : LA CHAINE DE TRANSMISSION DVB-S

### I.3.1.1.1 Principaux paramètres d'une chaîne d'émission/réception DVB-S

Le tableau 1 résume les principales caractéristiques d'une chaîne d'émission réception DVB-S comme c'est présenté par la figure 3. Les différents blocs composant la chaîne d'émission-réception doivent être configurés de sortes à obtenir une continuité des signaux durant la transmission et d'apporter à cette dernière une robustesse vis-à-vis des erreurs pouvant intervenir durant la communication dans le canal bruité.

<b>Paramètres d'une chaîne DVB-S</b>	
<b>Codage vidéo</b>	MPEG-2
<b>Codage audio</b>	MPEG-1
<b>Longueur des paquets transportés</b>	188 octets
<b>Polynôme de brouillage</b>	$1 + X^{14} + X^{15}$
<b>Codeur de Reed-Solomon</b>	204, 188, T = 8
<b>Entrelacement convolutif</b>	12 blocs
<b>Codeur convolutif</b>	171.133
<b>Modulation</b>	QPSK
<b>Roll-off</b>	0.35
<b>Largeur du canal</b>	de 26 à 36 MHz
<b>Débit</b>	23.7 à 41.5 Mb/s

Tableau I.1 : Principaux paramètres d'une chaîne DVB-S

### I.3.1.1.1.1 Le codeur Reed Salomon

Le code de Reed-Salomon est un code détecteur et correcteur. Ce code est basé sur les corps de Galois dont le principe est de construire un polynôme formel à partir des symboles à transmettre et de le sur-échantillonner. Le résultat est alors envoyé, au lieu des symboles originaux. La redondance du sur-échantillonnage permet au récepteur du message encodé de reconstruire le polynôme même s'il y a eu des erreurs pendant la transmission. Ces codes ont une propriété importante, ils sont linéaires et font partie des codes BCH. Le codeur prend  $k$  symboles de donnée (chaque symbole contenant  $s$  bits) et calcule les informations de contrôle pour construire  $n$  symboles, ce qui donne  $n-k$  symboles de contrôle. Le décodeur peut corriger au maximum  $t$  symboles, ou  $2t=n-k$ . [2]

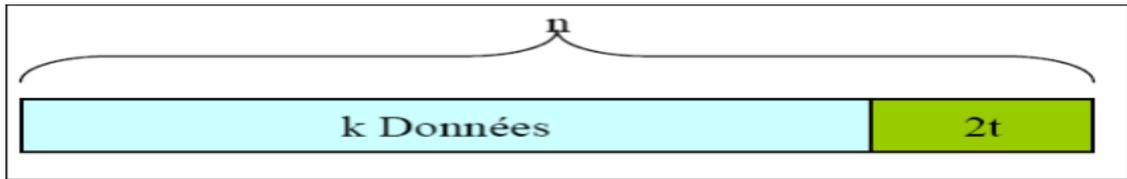


Figure I.3 : Mot-code de Reed Salomon

### ***1.3.1.1.2 L'entrelaceur***

A l'émission, les octets d'un paquet sont répartis dans d'autres paquets, ce qui permet d'éviter d'avoir à corriger une longue suite de bits (ou octets) faux consécutifs. L'entrelacement permet de répartir les erreurs sur plusieurs paquets, facilitant ainsi la détection et la correction d'erreurs du décodeur RS. Le principe de l'entrelaceur convolutif utilise plusieurs registres à décalages qui vont induire un retard. L'entrelaceur utilise deux paramètres : le nombre de branches  $K$  et la profondeur  $T$  (en octet) du registre à décalage de base.

### ***1.3.1.1.3 Le codeur convolutif***

Les codes convolutifs, introduits en 1955 par Elias, peuvent être considérés comme un cas particulier des codes en bloc linéaires, mais un point de vue plus large nous fera découvrir que la structure convolutive additionnelle munit le code linéaire de propriétés favorables qui facilitent à la fois son codage et améliorent ses performances. Les codes convolutifs forment une classe extrêmement souple et efficace de codes correcteurs d'erreurs. Ce sont les codes les plus utilisés dans les systèmes de télécommunications fixes et mobiles. Théoriquement, ils ont les mêmes caractéristiques que les codes en blocs sauf pour la valeur de leur dimension et leur longueur. Les codes convolutifs s'appliquent sur des séquences infinies de symboles d'information et génèrent des séquences infinies de symboles codés.

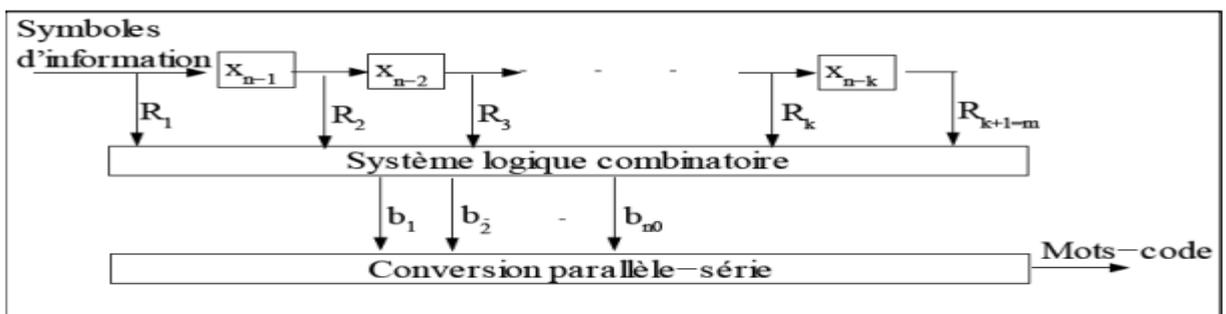


Figure I.4 : Le principe général du codage convolutif.

### I.3.1.1.4 Le modulateur QPSK

Pour transmettre des signaux MPEG-2 sur un transpondeur satellite, on utilise le QPSK (Quadrature Phase Shift Keying) pour moduler les informations numériques sur la fréquence porteuse. Plutôt que d'utiliser l'amplitude ou la fréquence de la porteuse pour transporter l'information, QPSK module la phase de la porteuse. En fonction des données à moduler, la porteuse est forcée dans une des quatre phases possibles, aussi appelée symbole. Le grand avantage de cette méthode est que chaque symbole code deux bits de données, donc double la quantité potentielle de données qui serait transmise avec une modulation d'amplitude ou de fréquence. Les dessins ci-dessous illustrent une implémentation typique de QPSK

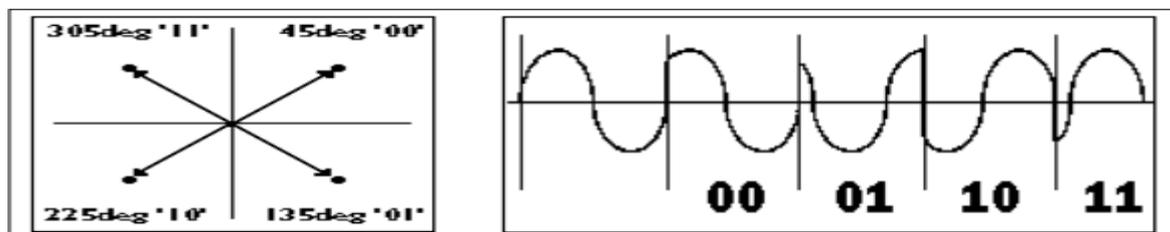


Figure I.5 : La modulation QPSK

La figure I.6 montre chaque paire possible de bits de données représentée par un angle de phase différent et un exemple de signal QPSK. A cause de l'utilisation du QPSK, le débit des données sont exprimés en débit symbole (symbol rate) plutôt qu'en débit binaire (bit rate). Dans le cas de la modulation QPSK, le débit binaire est deux fois le débit symbole. Par exemple un SR de 20MS/s (20 méga-symboles par seconde) équivaut à un débit binaire de 40Mb/s (40 mégabits par seconde).

### I.3.1.2 Le Standard DVB-S2

Le standard DVB-S2 représente une évolution de la diffusion numérique pour la télévision. Avec les nouveautés apportées, Ils ont gagné en efficacité spectrale d'une valeur de l'ordre 25% à 30% par rapport aux standards existants équivalents, en plus des nouvelles applications introduites. Cet apport par le standard DVB-S2 s'explique par les modifications introduites au niveau codage et modulation. Le tableau I.2 représente les principales différences du standard DVB S2 avec le DVB-S et le DVB-DSNG. [1]

Année	1994	2003
Standard	DVB-S	DVB-S
Mode de codage et de modulation	CCM	ACM
Codage	Viterbi, Reed Salomon	LDPC, BCH
Modulation	QPSK	QPSK, 8PQK, 16PSK, 32PSK
Format des données	MPEG2	MPEG2, MPEG4, donnée

Tableau I.2 : Evolution du standard de communication par satellite.

Ces modifications introduites pour le standard DVB-S2, ont permis d'attendre une performance remarquable que nous allons détailler dans la suite.

### **I.3.1.2.1 Caractéristiques du DVB-S2**

Les différentes caractéristiques du standard DVB-S2 sont présentées ci-dessus.

#### **I.3.1.2.1.1 Codage avancé**

Le codage canal adopté est une concaténation d'un code en bloc BCH (Bose-Chaudhuri-Hochquenghem code) et d'un code LDPC. Le code LDPC est un code linéaire caractérisé par sa grande capacité de détection d'erreur. Mais bien que ce code soit connu par sa complexité à cause de son besoin intense en mémoire, les problèmes posés par leur intégration matérielle commencent à être abordés.

#### **I.3.1.2.1.2 Modulation et codage Variable**

Afin de garantir la qualité de service requise par les différentes applications et d'exploiter les ressources spectrales d'une manière plus efficace, le DVB-S2 adopte à la fois un codage variable et une constellation variable. Les différentes possibilités de modulation et de taux de codage sont données dans le tableau I.3.

Modulation	Taux de codage
QPSK	1/4, 1/3, 2/5, 1/2, 3/5, 2/3, 3/4, 4/5, 5/6, 8/9 et 9/10
8PSK	3/5, 2/3, 3/4, 5/6, 8/9 et 9/10
16PSK	2/3, 3/4, 4/5, 5/6, 8/9 et 9/10
32PSK	3/4, 4/5, 5/6, 8/9 et 9/10

Tableau I.3 : Modulation et taux de codage utilisés

### I.3.1.2.1.3 Modulation et codage adaptatif

Le standard DVB-S2 permet de fournir des services interactifs en plus de la modulation et du codage variable : un canal de retour est utilisé pour réaliser la modulation et le codage adaptatif : ACM (Adaptive Coding and Modulation). Cette technique a permis une protection des canaux plus grande et une augmentation de la capacité de transmission, en plus de l'introduction de nouveaux services.

### I.3.1.2.1.4 Chaîne de transmission DVB-S2

Le système de transmission DVB-S2 est formé par des blocs pratiques qui assurent l'adaptation entre les flux d'entrées qui peuvent être des flux de transport MPEG ou des sources de données génériques et le signal RF à la sortie. Comme c'est montré dans la figure I.7, la chaîne de transmission est formée des séquences de bloc que nous allons les décrire ci-dessous.

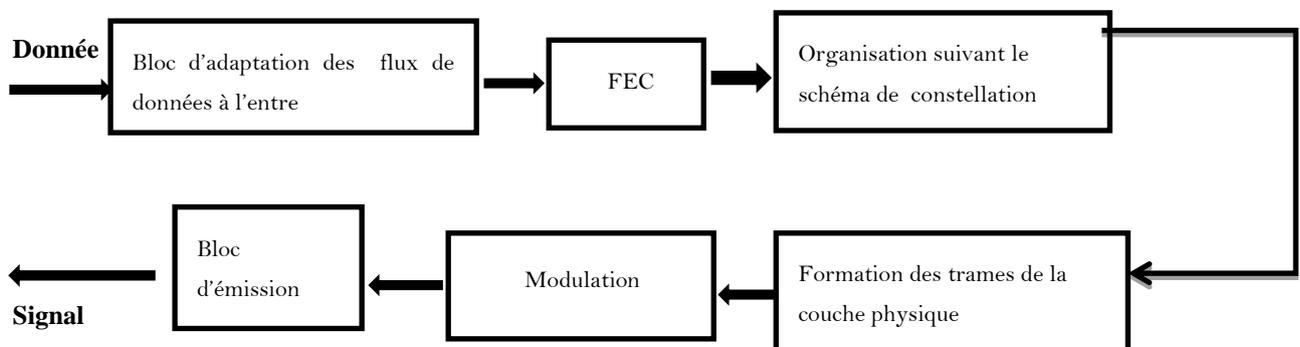


Figure I.6 : Système de transmission DVB-S2

## I.3.1.2.1.5 Bloc d'adaptation

C'est une interface d'entrée qui assure l'adaptation et la synchronisation des flux entrants. Il donne comme sortie des paquets de longueurs fixes  $UPL=188 \times 8$  bits (User Packet Length). Les types de données à l'entrée et à la sortie de ce bloc sont donnés dans le tableau I.4.

Interface	Interface Type	Connexion
Entrée	MPEG [1,4]	Multiplexeur MPEG
Entrée	Données génériques	Source de données
Entrée	Commande ACM	Unité de contrôle
Sortie	Trames de données	Bloc de réception

Tableau I.4 : Type de données pour un système de transmission DVB-S2

L'adaptation passe par plusieurs étapes dont la première étape est la synchronisation. Cet étage a pour but d'assurer un débit constant. La deuxième étape est la suppression des paquets nuls du flux MPEG après identification, ce qui permet de réduire le taux des données et d'augmenter le taux de protection. Les paquets nuls supprimés seront insérés dans leurs positions d'origine dans le récepteur. Ensuite, l'information passe par un codeur pour la détection d'erreur CRC-8 (Cyclic Redundancy Check). Les différentes étapes du bloc d'adaptation sont présentées par la figure I.8.

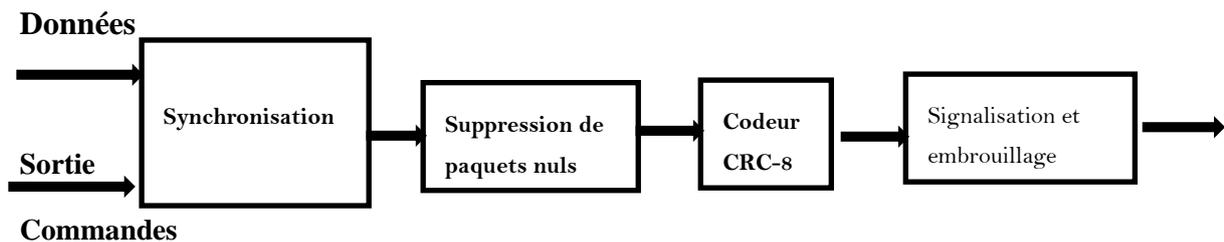


Figure I.7 : Bloc d'adaptation

Les données sont, ensuite, regroupées dans des champs plus grands : DF (Data Field), pour lesquelles nous ajoutons des informations de signalisation de longueur fixe (80

bits) pour obtenir la trame BBFRAME (Base Band Frame), comme c'est montré dans la figure I.9.

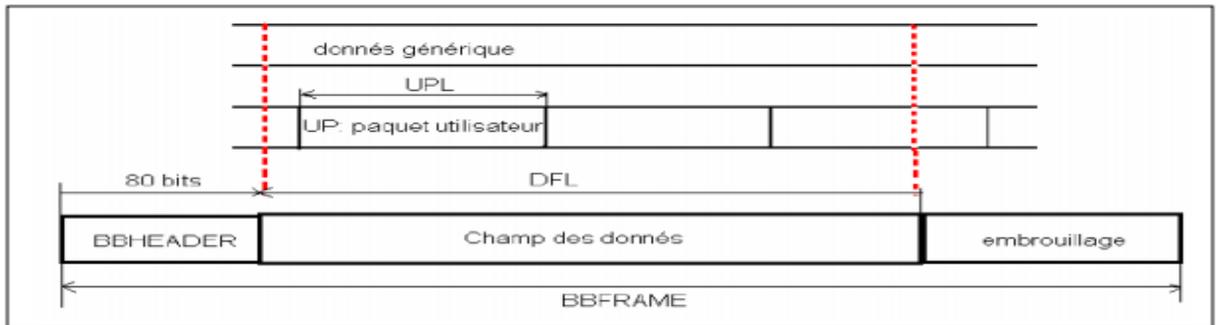


Figure I.8 : Trame BBFRAME

### I.3.1.2.1.6 Codeur FE

La correction d'erreur directe (FEC) est le sous-système fondamental pour obtenir d'excellentes performances par satellite, avec des niveaux de bruit et de brouillage particulièrement élevés. Dans ce bloc nous appliquons les codes retenus par le standard DVB-S2 qui sont les codes BCH et LDPC avec un taux de codage qui peut être  $1/4$ ,  $1/3$ ,  $2/5$ ,  $1/2$ ,  $3/5$ ,  $2/3$ ,  $3/4$ ,  $4/5$ ,  $5/6$ ,  $8/9$  ou  $9/10$ . Le nombre total des bits de la trame de sortie doit être constant, égale soit à 64800 ou à 16200.

Les codes BCH (Bose Chaudhuri Hocquenghem) externes concaténés permettent d'éviter les effets de seuil (d'erreur) dans le cas de faible Taux d'Erreur Binaire (TEB).

### I.3.1.2.1.7 Organisation suivant le schéma de constellation

Ce bloc a pour fonction la génération des symboles suivant le type de modulation adopté. Par exemple, et comme le montre le schéma de constellation 16APSK dans la figure I.10, les bits sont regroupés en des symboles de 4 bits.

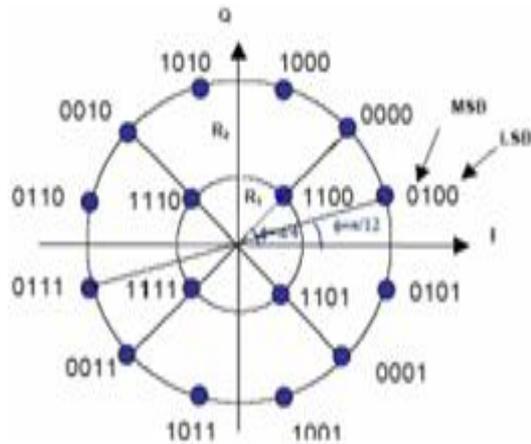


Figure I.9 : Schéma de constellation 16-APSK

### I.3.1.2.1.8 Bloc de formation des trames de la couche physique

Dans ce bloc le flux d'entrée sera coupé en slots de taille 90 symboles. Après chaque groupe de 16 slots, nous ajoutons un entête qui contient des informations des signalisations.

### I.3.1.2.1.9 La modulation

Dans ce bloc, le signal va être transformé de la bande de base vers la bande RF d'émission. Avant l'émission on filtre notre signal par un filtre en racine de Cosinus Surélevé avec un facteur de roll-off de 0.35, 0.25 ou 0.2. La chaîne de transmission du standard DVB-S2 ainsi défini, permet donc de générer un signal bien protégé et à débit plus important que celui du standard DVB-S.

### I.3.1.3 Etude comparative de DVB-S et DVB-S2

La comparaison entre les deux normes DVB-S et DVB-S2 se résume dans le tableau I.5. [1]

<b>Standard</b>	<b>DVB-S</b>	<b>DVB-S2</b>
<b>Codage</b>	Viterbi, Reed solo mon	LDPC, BCH
<b>Modulation</b>	QPSK	QPSK, 8PSK, 16APSK, 32APSK
<b>Mode de codage et de modulation</b>	CCM	ACM
<b>Formats des données</b>	MPEG-2	MPEG-2, MPEG-4, données
<b>Facteur de décroissance (Roll-off)</b>	0.35	0.35, 0.25, 0.2
<b>Débit binaire</b>	23.7 à 41.5 Mb/s	36 à 51 Mb/s

Tableau I.5 : Les paramètres de comparaison du DVB-S et DVB-S2

### **I.3.2 Le standard de diffusion numérique DVB-T**

#### **I.3.2.1 Le Standard DVB-T**

Le système de télévision numérique appelé DVB-T (Terrestrial Digital Video Broadcasting), spécifié dans la norme ETSI (Normes de l'Institut européen des télécommunications) EN 300 744. Ce système est conçu pour permettre utilisation du spectre de fréquences disponible avec une structure de données de diffusion suffisante pour accueillir nombreux services: multiplex de jusqu'à 8 programmes vidéo dans une bande passante de 8 MHz (où un seul programme diffusait), chaînes stéréo / sur round multilingues, etc. L'architecture du réseau DVB-T consiste en: codeur de programme, multiplexeur, adaptateur de réseau SFN, Modulateur COFDM (Coded Orthogonal Frequency Division Multiplex), convertisseur élévateur et transmetteur. Il spécifie tout le processus d'utilisation des canaux de transmission terrestres: codage et modulation des canaux. Le système de télévision numérique appelé DVB-T (Terrestrial Digital Video Broadcasting), spécifié dans la norme ETSI (Normes de l'Institut européen des télécommunications) EN 300 744. Ce système est conçu pour permettre utilisation du

spectre de fréquences disponible avec une structure de données de diffusion suffisante pour accueillir nombreux services [4] :

- Fréquence DVB-T: 8 MHz
- Debit DVB-T: 25 Mbits/s
- Reception mobile de qualité.
- Décrochages locaux
- 6 canaux en analogique= 30 à 36 en numerique

### I.3.2..1.1 Schéma bloc du standard DVB-T

Après un générateur de données, le bloc MPEG-2 est utilisé pour la compression de l'image et du son. Par la suite, il y a le bloc mapper qui effectue la modulation numérique en bande de base afin de placer les symboles numériques dans la constellation. Le bloc A/D est un convertisseur. Il effectue la conversion d'un nombre entier en un mot binaire. Nous passons ensuite au bloc OFDM qui applique la IFFT. Le bloc « Guard interval insertion » a pour but d'insérer un intervalle de garde entre les symboles. A la réception, les techniques inverses sont appliquées pour reconstruire le signal original.

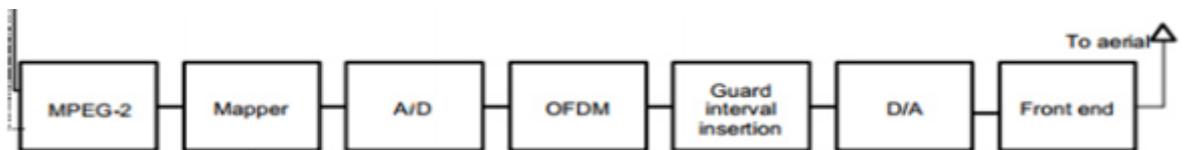


Figure I.10 : Schéma à blocs fonctionnels du standard DVB-T. [4]

En DVB-T est utilisé un codage canal utilisant un bloc de code et un code de convolution. En plus délais techniques de codage de canal sont également utilisées pour l'entrelacement des techniques afin d'éviter les erreurs dans les blocs de sortie au démodulateur.

Le système DVB-T permet une gamme d'options qui en font un système flexible:

- modes de transmission: 2k (1705 porteuses), 8k (porteuses 6817)
- schémas de modulation: QPSK, 16-QAM, 64-QAM
- Protection des erreurs de codage interne des relations: 1/2, 2/3, 3/4, 5/6, 7/8.
- Longueurs pour l'intervalle de garde: 1/4, 1/8, 1/16, 1/32.

- Modulation de canal non hiérarchique ou hiérarchique avec différentes valeurs de paramètre alpha.

### I.3.2.1.3 Paramètres techniques du standard DVB-T

Le choix des paramètres de transmission OFDM pour un système DVB-T dépend de l'application visée et des performances qu'on veut satisfaire. Les paramètres de transmission comprennent la largeur de bande du canal, le nombre de sous-porteuses, l'espacement entre les sous-porteuses et la durée de l'intervalle de garde.

La norme DVB-T prévoit l'utilisation de deux modes 2K ou 8K, les deux tableaux I.6 et I.7 représentent les différents paramètres utilisés.

Paramètre	Mode 2K			
Bande passante	8 MHz			
Nombre de porteuse	1704			
Espace entre sous porteuse	213 us			
Modulation	4-MAQ	16-MAQ	64-MAQ	
Intervalle de garde	1/4	1/8	1/16	1/32
Nombre de symboles	100			
Chaîne de transmission	AWGN (Additive white Gaussian Noise)			

Tableau I.6 : Mode 2k de standard DVB-T

Paramètre	Mode 8K			
Bande passante	8 MHz			
Nombre de porteuse	6816			
Espace entre sous porteuse	852 us			
Modulation	4-MAQ	16-MAQ	64-MAQ	
Intervalle de garde	1/4	1/8	1/16	1/32
Nombre de symboles	100			
Chaîne de transmission	AWGN (Additive white Gaussian Noise)			

Tableau I.7 : Mode 8k de standard DVB-T.

### I.3.2.1.4 Évolution de la DVB-T à DVB-T2

Le DVB-T2 est la deuxième génération de la norme DVB-T. Il a été publié à la fin de l'année 2008. Ce système offre une amélioration de l'efficacité dans l'utilisation du spectre autour de 30-50% que le DVB-T.

Les caractéristiques de ce système sont les suivantes :

- La priorité est d'être multicast sur HDTV, SDTV, et de permettre les services interactifs et autres dans systèmes numériques et appareils mobiles.
- DVB-T2 fonctionnera avec les systèmes d'antennes existants, en transmission et en réception.
- DVB-T2 fournira une vitesse de transmission plus élevée.
- A spécification DVB-T2 devrait offrir une robustesse améliorée contre les interférences d'autres émetteurs, par rapport à DVB-T, potentiellement améliorer la réutilisation des fréquences.
- La spécification DVB-T2 doit offrir un choix de différents niveaux de robustesse et de protection à s'applique également à toutes les données d'un flux de transport acheminé par un signal DVB-T2 dans un canal particulier.

- L'utilisation de ce système dans SFN permettra une plus grande distance entre les répéteurs existants que dans le système DVB-T.

### I.3.3 Le standard de diffusion numérique DVB-C

#### I.3.3.1 Le standard DVB-C

DVB-C "Digital Video Broadcasting - Cable" c'est la norme du consortium européen DVB pour la diffusion de la télévision numérique par câble. Ce système transmet un flux audio numérique / vidéo numérique MPEG-2 ou MPEG-4 en utilisant une modulation QAM avec codage de canal. Publiée pour la première fois par l'ETSI en 1994, cette norme est ensuite devenue le système de transmission de télévision numérique par câble le plus utilisé en Europe, en Asie et en Amérique du Sud. Il est déployé dans le monde entier dans des systèmes allant des plus grands réseaux de télévision par câble (CATV) aux plus petits systèmes de télévision par antenne maîtresse par satellite (SMATV).

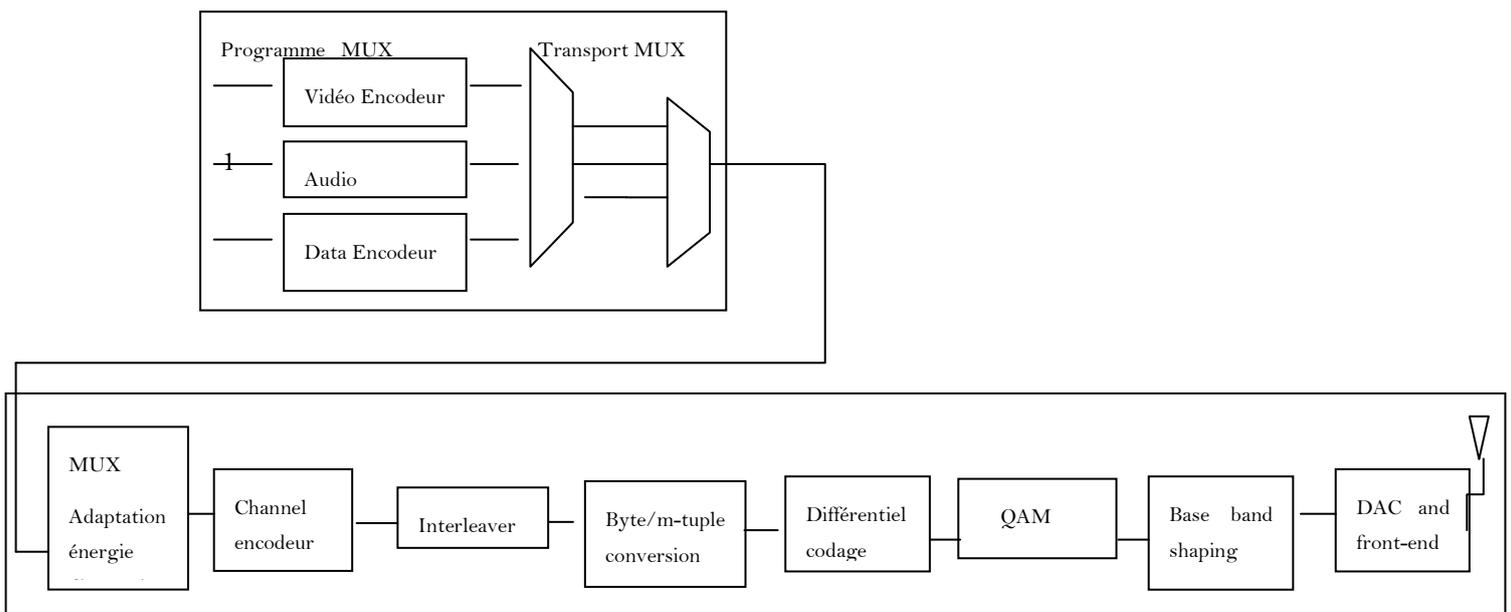


Figure I.11 : SYSTEME DE TRANSMISSION DVB-C

#### I.3.3.2 Les composants du système DVB-C

##### I.3.3.2.1 Codage de source et multiplexage MPEG-2 (MUX)

Les flux vidéo, audio et de données sont multiplexés en un flux de programme MPEG (MPEG-PS). Un ou plusieurs MPEG-PS sont réunis en un flux de transport MPEG

(MPEG-TS). Il s'agit du flux numérique de base qui est transmis et reçu par les décodeurs domestiques (STB) ou le module de décodeur intégrable correspondant (par exemple Canaux). Les débits binaires autorisés pour le MPEG-2 transporté dépendent d'un certain nombre de paramètres de modulation: ils peuvent varier d'environ 6 à environ 64 Mbit/s (voir la figure du bas pour une liste complète). [6].

### **I.3.3.2 Adaptation MUX et dispersion d'énergie**

Le MPEG-TS est identifié comme une séquence de paquets de données, de longueur fixe (188 octets). Avec une technique appelée dispersion d'énergie, la séquence d'octets est décarrelée.

#### **I.3.3.2.3 Encodeur externe**

Un premier niveau de protection est appliqué aux données transmises, à l'aide d'un code de bloc non binaire, un code Reed-Salomon RS (204,188), permettant la correction d'un maximum de 8 octets erronés pour chaque paquet de 188 octets.

#### **I.3.3.2.3 Entrelaceur externe**

L'entrelacement convolutionnel est utilisé pour réarranger la séquence de données transmises, de telle sorte qu'il devient plus robuste à de longues séquences d'erreurs.

#### **I.3.3.2.4 Conversion byte/m-tuple**

Les octets de données sont encodés en m-tuples bits ( $m = 4, 5, 6, 7$  ou  $8$ ).

#### **I.3.3.2.5 Codage différentiel:**

Pour obtenir une constellation à variante de rotation, cette unité doit appliquer un codage différentiel des deux Bits les Plus Significatifs (BPS) de chaque symbole.

#### **I.3.3.2.6 Carte QAM**

La séquence de bits est mappée dans une séquence numérique en bande de base de symboles complexes. Il y a 5 modes de modulation autorisés: 16-QAM, 32-QAM, 64-QAM, 128-QAM, 256-QAM.

#### **I.3.3.2.7 Mise en forme de la bande de base**

Le signal MAQ est filtré à l'aide d'un filtre en relief en forme de cosinus, afin d'éliminer les interférences mutuelles du côté de la réception.

### I.3.3.2.8 DAC et front-end

Le signal numérique est transformé en signal analogique, avec un convertisseur numérique-analogique (DAC), puis modulé en fréquence radio par le front-end RF.

### I.3.3.3. Débits binaires disponibles pour un système DVB-C (Mbit/s)

Modulation	Largeur de bande (MHz)				
	2	4	6	8	10
16 QAM	6.41	2.83	9.23	25.64	32.05
32 QAM	8.01	6.03	4.04	32.05	40.07
64 QAM	9.62	9.23	8.85	38.47	48.08
128 QAM	11.22	22.44	33.66	44.88	56.10
256 QAM	12.82	25.64	38.47	51.29	64.11

Tableau I.8 : Débits binaires disponibles

### I.3.3.4 Évolution de la DVB-C à DVB-C2

Le 18 février 2008, il a été annoncé qu'une nouvelle norme - DVB-C2 - serait élaborée au cours de l'année 2008 et un "Appel à Technologies" a été lancé. Des propositions comprenant des programmes de simulation et des renseignements sur les droits de brevet pourraient être soumises jusqu'au 16 juin 2008. "Les résultats de la mission d'étude DVB-C2 ont déjà fourni des indications claires sur l'existence de technologies permettant aux performances du système de transmission par câble DVB de deuxième génération de se rapprocher si près de la limite théorique de Shannon que toute amélioration ultérieure ne justifierait probablement pas l'introduction d'un système de transmission par câble perturbateur de troisième génération. Grâce à l'utilisation de techniques de codage et de modulation de pointe, le DVB-C2 devrait offrir une efficacité spectrale supérieure de plus de 30 % dans les mêmes conditions, et les gains de capacité du canal aval seront supérieurs à 60

% pour les réseaux HFC optimisés. La spécification DVB-C2 finale a été approuvée par le Comité directeur DVB en avril 2009. DVB-C2 permet des débits binaires jusqu'à 83,1 Mbit/s sur une bande passante de 8 MHz lorsque la modulation MAQ 4096 est utilisée; les extensions futures permettront jusqu'à 97 Mbit/s et 110,8 Mbit/s par canal avec une modulation MAQ-16384 et 65536-AQAM. [6].

### **1.3.4 Le standard de diffusion numérique DVB-H**

#### **1.3.4.1 Le standard de DVB-H**

DVB-H (Digital Video Broadcasting - Handheld) est l'un des trois formats de télévision mobile courants. C'est une spécification technique pour apporter des services de diffusion aux combinés mobiles. Le DVB-H a été formellement adopté en tant que norme ETSI EN 302 304 en novembre 2004. La spécification DVB-H peut être téléchargée sur le site officiel DVB-H. Depuis mars 2008, DVB-H est officiellement reconnu par l'Union européenne comme la "technologie préférée pour la diffusion mobile terrestre". Les principaux concurrents de cette technologie sont le système Media FLO de Qualcomm, la norme de télévision mobile MBMS basée sur le système cellulaire 3G, et le format ATSC-M/H du DVB-SH américain (Satellite to Handhelds) et du DVB-NGH (Next Generation Handheld). À l'avenir, il est possible d'améliorer le DVB-H, en fournissant une efficacité spectrale améliorée et une meilleure flexibilité de modulation. Le DVB-H a été un échec commercial, et le service n'est plus en ondes [douteux - discute]. La Finlande a été le dernier pays à couper ses signaux en mars. [7].

L'économie d'énergie réalisable dépend de la relation du temps de marche / arrêt. S'il y a environ dix services éclatés ou plus dans un flux DVB-H, le taux d'économie d'énergie pour le frontal peut atteindre 90%. DVB-H est un système technique qui a été soigneusement testé par le Groupe de travail sur la validation DVB-H au cours de l'année 2004. DVB-SH améliore les performances radio et peut être vu comme une évolution de la DVB-H.

DVB-H est conçu pour fonctionner dans les bandes suivantes:

- VHF-III (170-230 MHz, ou une partie de celui-ci)
- UHF-IV / V (470-862 MHz, ou une partie de celui-ci) L (1,452-1,492 GHz)
- DVB-SH maintenant et DVB-NGH dans un proche avenir devraient étendre les bandes supportées.

## I.3.4.2 Les composants du système DVB-H

Les composants fonctionnels de base d'un système DVB-H sont représentés sur la figure I.13 un composant fonctionnel peut être plusieurs équipements. D'autre part, plusieurs composants fonctionnels peuvent être combinés en un seul équipement. Les flèches en trait plein représentent le lien d'information obligatoire pour un système DVB-H, tandis que les flèches en pointillés représentent le lien facultatif pour un système DVB-H, le détail des composants concernés est décrit ci-dessous

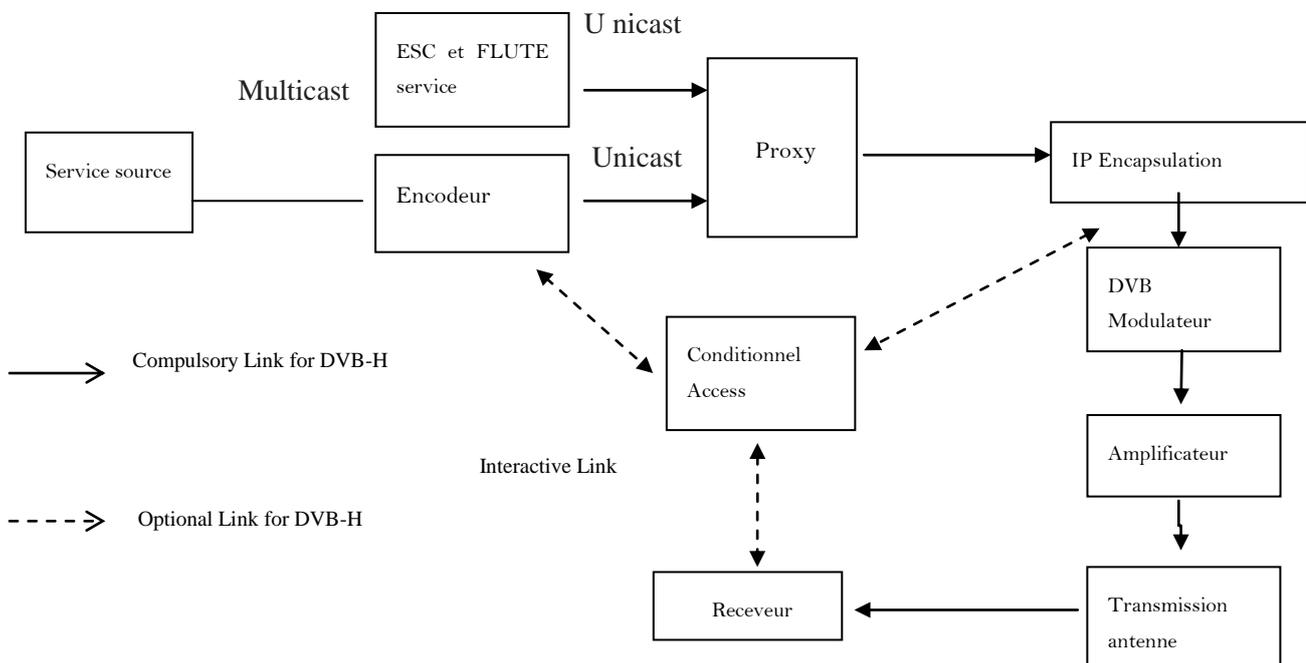


Figure I.12 : Système de transmission DVB-H

### I.3.4.2.1 La source de service

Est le composant situé du côté des fournisseurs d'applications. La source de service produit différents programmes audio / vidéo / données soit codés dans un certain format, soit non encodés.

### **I.3.4.2.2 Encodeur**

Le codeur est le composant qui code le contenu du nœud de source de service au format audio / vidéo / données approprié, qui peut être utilisé par les systèmes DVB-H.

### **I.3.4.2.3 Serveur ESG et FLUTE**

Les serveurs ESG et FLUTE sont utilisés pour produire l'ESG et le convertir au format correct pour la transmission dans le carrousel FLUTE. Le serveur FLUTE peut également être utilisé pour transmettre d'autres données normales (par exemple, des logiciels pour la mise à jour).

### **I.3.4.2.4 PROCURATION**

Le proxy est le nœud qui convertit le trafic unicast entrant vers le trafic de multidiffusion sortant.

### **I.3.4.2.5 ENCAPSULATEUR IP**

Est pour l'encapsulation des paquets IPv4 / IPv6 entrants aux paquets MPEG2 en cours.

### **I.3.4.2.6 Modulateur DVB**

Le modulateur DVB a pour fonction de moduler les paquets MPEG2 à rayonner dans l'air pour la transmission.

### **I.3.4.2.7 AMPLIFICATEUR**

L'amplificateur est en charge de l'amplification des signaux DVB modulés.

### **I.3.4.2.8 ANTENNE DE TRANSMISSION**

La taille de l'antenne d'émission varie également en fonction de la puissance de transmission prévue et de la zone de couverture ciblée.

### **I.3.4.2.9 DESTINATAIRE**

Le récepteur DVB-H est un appareil qui est un câble de réception de signaux DVB-H. Le téléphone mobile avec la capacité de réception DVB-H intégrée est un appareil DVB -H typique.

D'autres récepteurs mobiles portables et même les ordinateurs portables sont également des appareils DVB-H typiques.

### I.3.4.3 La diffusion par les technologies DVB-T et DVB-H

Dans ce tableau on donne la diffusion des technologies DVB-T et DVB-H :

	<b>Origine rédacteur et références</b>	<b>Références du standard ou de la norme ETSI</b>	<b>Code audio</b>	<b>Bandes de fréquences en MHz</b>
<b>DVB-T</b>	Spécification du consortium DVB	EN 300 744 (Juin 2004)	MPEG-1/2 audio layer ,Dolby AC-3  Enhanced AC-3,DTS,MPEG-4HE,AAC	Bande III : 173-223  Bande IV et V: 470-830
<b>DVB-H</b>	Spécification du consortium DVB	EN 302 304 (Novembre 2004)	MPEG-1/2 audio layer ,Dolby AC-3Enhanced AC-3,DTS,MPEG-4HE,AAC	Bande III : 173-223  Bande IV et V:470-830  Bande L: 1452-1492

Tableau I.9 : LA DIFFUSION DES TECHNOLOGIE DVB-T ET DVB-H.

## I.4 Conclusion

Dans ce chapitre, nous avons présenté la chaîne de transmission DVB ainsi que les standards existants de diffusion de la télévision numérique à savoir DVB-S, DVB-T, DVB-C et DVB-H.

# **Chapitre II**

**Introduction des techniques  
de modulation et de codage**

## CHAPITRE II : INTRODUCTION DES TECHNIQUES DE MODULATION ET CODAGE

### II.1 Introduction

Dans ce chapitre nous allons présenter les codes RS, OFDM, LDPC et le codage BCH. Nous allons commencer par présenter les caractéristiques générales de ces codes. Pour nous présenterons la façon de les construire.

### II.2 La modulation OFDM

#### II.2.1 Principe de l'OFDM

La modulation OFDM (multiplexage par répartition orthogonale de fréquence) est une technique qui divise la bande fréquentielle en un nombre de sous-canaux égale à  $N$  (ou sous-porteuses) orthogonaux et uniformes c'est-à-dire les symboles vont être répartie sur un grand nombre de porteuse a débit faible (figure 14), à l'opposé des systèmes conventionnels qui transmettent les symboles en série, chaque symbole occupe alors toute la bande passante disponible.[5]

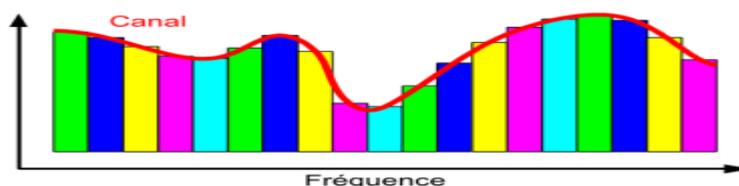


Figure II.1:EFFET DE CANAL SUR LA TECHNIQUE OFDM

#### II.2.2 L'Orthogonalité

La technique de modulation OFDM autorise un fort recouvrement spectral entre les sous porteuses. Cependant, les porteuses doivent respecter une contrainte d'orthogonalité dans les domaines temporel et fréquentiel en même temps pour un bon fonctionnement. L'orthogonalité se réalise si l'espace entre deux fréquence adjacentes  $f_n=1/T_n$

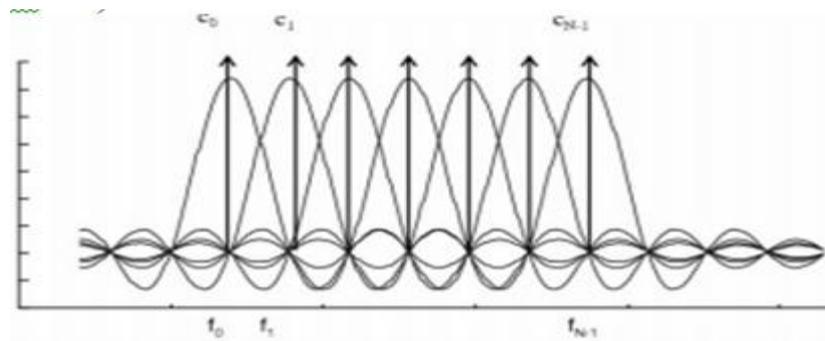


Figure II.2: Spectre en sortie du modulateur OFDM

Ainsi, il n'existe pas des interférences avec les autres sous porteuse, c'est ce qui permet recouvrement de spectre des différentes porteuses et une occupation optimale du spectre.

### I.2.3 Intervalle de garde

Les symboles subissent des échos et un symbole émis parvient au récepteur sous forme de plusieurs symboles atténués et retardés. Un symbole émis lors d'une période  $i \cdot T_S$  peut se superposer à un écho provenant du symbole émis à la période  $(i-1) \cdot T_S$ , il se produit alors des interférences. Pour éviter ces interférences, on ajoute un intervalle de garde de durée  $T_g$ . Chaque symbole est précédé par une extension périodique du signal lui-même, la durée du symbole totale transmis est alors  $T = T_S + T_g$ .

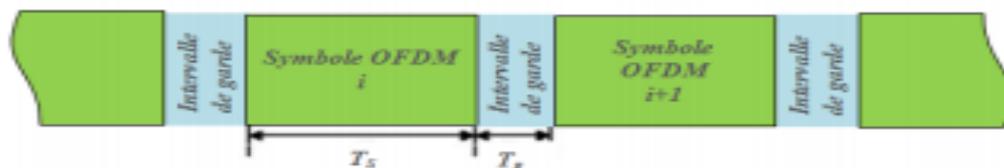


Figure II.3 : Intervalle de garde

Il existe trois d'intervalle de garde :

**Cycle-Prefix CP** : Est une technique qui consiste à insérer une copie d'un bloc d'information à transmettre en amont de la trame. Plus clairement, il s'agit de récupérer une partie des informations à transmettre d'insérer ces informations en début de trame.

**Zero-Padding ZP** : Insérer des zéros au lieu d'intervalle de garde et aucun signal est transmise pendant l'intervalle de garde.

**KSP** : Où l'intervalle de garde est constitué de symboles pilotes, cette technique d'intervalle de garde peut être utile pour résoudre l'ambiguïté de synchronisation temporelle qui se produit avec d'autres techniques d'intervalle de garde.

### II.3 Codage par Reed-Salomon

#### II.3.1 Définition

Les codes de Reed-Salomon RS  $(k, t)$  sont formés de  $n$  symboles, avec  $n = q - 1$  au maximum et  $q = 2k$ , chaque symbole appartenant à  $GF(q)$  qui est le champ de Galois (Galois Field) à  $q$  éléments,  $k$  représente donc le nombre de bits par symbole. Le nombre  $t$  représente le nombre de symboles d'erreurs que ce code sera capable de corriger.

Le nombre de symboles de contrôle est de  $2t$ . Par conséquent, le nombre de symboles d'information que l'on peut transmettre est de  $m = n - 2t$ . [9]

#### II.3.2 champ de Galois

La notation pour la suite, sera la suivante :

- $F$  représente un corps fini (Field).
- $F_p$  représente le corps fini à  $p$  éléments.
- $F^*$  représente le groupe multiplicatif du corps  $F$ . [9]

##### II.3.2.1 Propriétés d'un corps fini

On rappelle ici, quelques propriétés utiles des corps finis. Soit  $F$  un corps fini de caractéristique  $p$  ayant  $q$  éléments. Alors,

- $F$  est un  $F_p$ -espace vectoriel de dimension  $k$ , avec  $q = p^k$ .
- Le groupe additif de  $F$  est isomorphe au groupe  $(\mathbb{Z}/p\mathbb{Z}, +)$ .
- $F^*$  est cyclique d'ordre  $q - 1 = p^k - 1$ .
- Tout élément  $x$  de  $F^*$  vérifie  $x^{q-1} = 1$ .

Commentaire:

Nous affirme qu'il existe un élément, dit primitif, qui engendre le groupe multiplicatif  $F^*$ . Nous dit que  $F$  est un espace vectoriel de dimension  $k$  sur le corps  $F_p$ , le choix d'une base sera généralement les puissances entière de l'élément primitif, soit  $1, \alpha, \alpha^2, \alpha^3, \dots, \alpha^{k-1}$ . Un élément  $c$  de  $F$  s'écrira alors de la manière suivante :

$$c = c_{k-1} \cdot \alpha^{k-1} + \dots + c_1 \cdot \alpha^1 + c_0 \cdot \alpha^0 \quad (1)$$

Avec  $c_i$  appartenant à  $F_p$  pour  $i = 0, \dots, k-1$ . [9]

### II.3.2.2 Construction d'un corps fini

La construction d'un corps fini est basée sur le théorème suivant : [9]

**Théorème** : Soit  $\alpha$  un nombre algébrique sur  $F$ . Soit  $k$  le degré de son polynôme irréductible sur  $F$ . L'espace vectoriel engendré sur  $F$  par  $1, \alpha, \alpha^2, \dots, \alpha^{k-1}$  est alors un corps, et la dimension de cet espace vectoriel est  $k$ .

Ce théorème nous donne une méthode pour construire un corps avec une base donnée, mais ne nous donne pas de méthode pour trouver un élément primitif,  $\alpha$  n'est pas forcément un élément primitif.

#### II.3.2.2.1 Représentation des éléments

Il existe principalement deux méthodes pour représenter les éléments d'un corps fini, soit :

i)  $F_q$  est un  $F_p$ -espace vectoriel de dimension  $k$ , on choisit une certaine base  $\{b_1, \dots, b_k\}$  de cet espace et on repère chaque élément de  $F_q$  par ses composantes sur cette base.

ii) On prend un élément  $a$  de  $F^*$  qui engendre ce groupe et tout élément non nul de  $F_q$  s'écrit d'une manière, et d'une seule, sous la forme  $\alpha^m$ , avec  $m = 0, \dots, q-1$ .

Pour décrire un élément du corps, on utilisera exclusivement la première représentation, car elle permet d'associer un nombre à chaque élément du corps. Mais on prendra soin de tabuler ces deux représentations, la raison en est que la représentation i) se prête très bien pour l'addition et que la représentation ii) est bien adaptée pour la multiplication. Comme on a besoin de ces deux opérations, on travaillera avec ces deux représentations qui seront tabulées une fois pour toute.

### II.3.3 Technique de codage

Considérons un code de Reed-Salomon avec ses symboles dans  $GF(2^k)$ , où  $k$  est le nombre de bits par symbole. Soit

$$i(x) = c_{m-1} \cdot x^{m-1} + \dots + c_1 \cdot x^1 + c_0 \quad (2)$$

Le polynôme d'information, et soit

$$p(x) = c_{2t-1} \cdot x^{2t-1} + \dots + c_1 \cdot x^1 + c_0 \quad (3)$$

Le polynôme de contrôle, le code de Reed-Salomon sous sa forme polynomiale sera alors

$$c(x) = i(x) \cdot x^{2t} + p(x) = c_{n-1} \cdot x^{n-1} + \dots + c_1 \cdot x^1 + c_0 \quad (4)$$

avec  $c_i$  appartenant à  $GF(2^k)$  pour  $i = 0, \dots, n-1$ .

Un vecteur à  $n$  symboles,  $(c_{n-1}, \dots, c_1, c_0)$  est un code de Reed-Salomon si et seulement si son polynôme correspondant  $c(x)$  est un multiple du polynôme générateur  $g(x)$ . La méthode courante pour construire un tel polynôme, est de diviser  $i(x) \cdot x^{2t}$  par  $g(x)$  et d'ajouter le reste à  $c(x)$ . En effet,

$$i(x) \cdot x^{2t} = q(x) \cdot g(x) + r(x)$$

Où  $r(x)$  est le reste de la division de  $i(x) \cdot x^{2t}$  par  $g(x)$ ,

$$c(x) = i(x) \cdot x^{2t} + p(x) = q(x) \cdot g(x) + r(x) + p(x) = q(x) \cdot g(x) \quad (5)$$

Pour que  $c(x)$  soit un multiple de  $g(x)$ , soit  $c(x) = q(x) \cdot g(x)$ , il faut que  $p(x) = r(x)$ . Comme on travaille toujours sur des corps de caractéristique 2, l'opération de soustraction sera toujours égale à l'opération d'addition, soit de manière algébrique  $-1 = +1$ .

Cela nous donne une méthode pour construire le polynôme de contrôle, il suffit de prendre le reste de la division du polynôme  $i(x) \cdot x^{2t}$  par  $g(x)$ .

Il nous reste encore à construire le polynôme générateur  $g(x)$ . Il est défini de la manière suivante :

$$g(x) = (x + \alpha)(x + \alpha^2) \dots (x + \alpha^{2t}) = x^{2t} + g_{2t-1} \cdot x^{2t-1} + \dots + g_1 \cdot x^1 + g_0 \quad (6)$$

Où les coefficients  $g_i$  appartiennent à  $GF(2^k)$ , et est un élément primitif de  $GF(2^k)$ . [9]

### II.3.4 Technique de décodage

Considérons un code de Reed-Salomon  $c(x)$  correspondant au code transmis, et soit  $d(x)$  le code que l'on reçoit. Le code d'erreur est défini par :

$$e(x) = d(x) - c(x) = d(x) + c(x) \quad (7)$$

Car - et + sont équivalents dans  $GF(2^k)$ .

La première opération à faire, est de calculer les syndromes, qui sont définis de la manière suivante :

$$s_i = e(\alpha^i) = d(\alpha^i) + c(\alpha^i) = d(\alpha^i) \text{ avec } i = 1, \dots, 2t$$

car  $c(\alpha^i) = q(\alpha^i) \cdot g(\alpha^i) = 0$  puisque  $g(\alpha^i) = 0$  pour  $i = 1, \dots, 2t$  par définition de  $g(x)$ .

Le syndrome sous forme polynômiale sera :

$$s(x) = s_{2t} \cdot x^{2t-1} + \dots + s_3 \cdot x^2 + s_2 \cdot x + s_1 \quad (8)$$

Soit  $M$  l'ensemble des positions d'erreurs, on définit le polynôme de localisation (Locator polynomial) par :

$$l(x) = (1 - \alpha^{p_1} \cdot x)(1 - \alpha^{p_2} \cdot x) \dots (1 - \alpha^{p_j} \cdot x) \text{ avec } p_1, \dots, p_j \text{ dans } M$$

Si  $b = \alpha^i$  est une racine de  $l(x)$ , donc  $l(b) = 0$ , on aura alors un des monômes qui s'annule, soit  $(1 - \alpha^p \cdot b) = 0$ . Ce qui nous donne la relation suivante :

$b = \alpha^{-p} = \alpha^{q-1} \cdot \alpha^{-p} = \alpha^{q-p-1} = \alpha^i$  ( $\alpha^{q-1} = 1$ ), la position de l'erreur sera alors donné par

$p = q-1-i$  qui est positif et que l'on connaît, puisque l'on connaît  $b = \alpha^i$ , racine de  $l(x)$ . Il nous faudra donc chercher toutes les racines de  $l(x)$ . Il ne faut pas oublier que  $\alpha$  est un élément primitif, et que tout élément de  $GF(2^k)^*$  s'exprime comme une puissance de l'élément primitif, qui est un générateur du groupe  $GF(2^k)^*$ . [9]

### II.4 MPEG

#### II.4.1 Définition de MPEG

MPEG (Moving Picture Experts Group) est le groupe de travail SC 29/WG du comité technique mixte JTC de l'ISO et de la CEI pour les technologies de l'information. Ce groupe d'experts est chargé du développement de normes internationales pour la compression, la décompression, le traitement et le codage de la vidéo, de l'audio et de leur combinaison, de façon à satisfaire une large gamme d'applications.[3]

- Le codage des médias (audio, vidéo, graphique), c'est l'aspect le plus connu de MPEG qui est souvent confondu avec lui.
- Le codage des scènes composées.
- Le codage de la description des médias.
- Le support système.

#### II.4.2 Le développement de MPEG/ MPEG 2

MPEG a développé les normes suivantes : [3]

##### II.4.2.1 MPEG-1

Définie en 1988, il s'agit de la première norme audio et vidéo utilisée plus tard pour les Vidéo CD. Ce format offre une définition à l'écran de 352 x 240 pixels à 30 images par seconde ou de 352 x 288 à 25 images par seconde avec un débit d'environ 1,5 Mbit/s. Elle comprend le populaire format audio MPEG-1 partie 3 audio couche 3(MP3).

##### II.4.2.2 MPEG-2

Définie en 1994, cette norme est applicable au codage de l'audio et la vidéo, ainsi que leur transport pour la télévision numérique : télévision numérique par satellite, télévision numérique par câble, télévision numérique terrestre, et (avec quelques restrictions) pour les vidéodisques DVD ou SVCD. C'est notamment le format utilisé jusqu'à présent pour la TV sur ADSL. Les débits habituels sont de 2 à 6 Mbit/s pour la définition standard (SD), et de 15 à 20 Mbit/s pour la haute définition (HD).

##### II.4.2.3 MPEG-4

Définie à partir de 1999, cette norme est applicable aux bas débits (jusqu'à 2 Mbit/s), exclus de la matrice des décodeurs de MPEG-2. Permet, entre autres, de coder des objets vidéo/audio, le contenu 3D et inclut le DRM. La partie 2 de MPEG-4 (Visual) est compatible avec la partie Baseline de H.263 et a connu du succès grâce à la mise en application DivX ainsi que dans les téléphones mobiles.

- La partie 10 appelée MPEG-4 AVC ou H264, développée à partir de 2003, permet des gains d'un facteur 2 à 3 par rapport à MPEG-2 et a été retenue comme le successeur de celui-ci pour la TV haute définition, la TV sur ADSL et la TNT. L'extension de cette partie, appelée Scalable Video Coding (SVC) permet de proposer différents niveaux de qualité à partir d'un même flux codé.
- Le H.265/HEVC définie à partir de 2013, est la norme qui succède au H264, elle permet une diminution significative de la taille des vidéos par rapport à ce dernier.

### II.5 Codage LDPC

#### II.5.1 définition

La première fois normalise dans un contexte de diffusion par satellite : DVB-S2. Plus récemment, les codes LDPC ont été introduits dans les standards Wimax mobile et Wifi. Ces différents facteurs, ainsi que les nouveaux problèmes posent, ont contribué à accroître la popularité des codes LDPC dans le domaine industriel et scientifique. [10]

Un code LPDC (Low Density Parity Check) est un code dont la matrice de contrôle de parité H est de faible densité. Ainsi le nombre de "1" dans la matrice est faible devant le nombre de "0". On rappelle que la relation de contrôle de parité entre un mot de code et une matrice de codes LPDC. [10]

#### II.5.2 Codes LDPC

Le contrôle de parité est définie par :

$$Hx^t = 0^t \quad (9)$$

La matrice de contrôle de parité H est de dimension  $M \times N$  d' définissant donc un code en bloc ou le nombre de bits d'information est  $K = N - M$  (10)

## Chapitre II : Introduction des techniques de modulation et codage

□ Exemple: On considère la matrice de contrôle de parité suivante d'un code de rendement 1/2

$$\text{Et } H = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 1 & 0 & 0 \end{bmatrix} \quad (11)$$

Les équations de parité associée a cette matrice et à un mot de code  $x = [x_0, \dots, x_7]$  sont :

$$x_0 + x_4 = 0 \quad (12)$$

$$x_1 + x_4 + x_5 = 0 \quad (13)$$

Un code LDPC peut également être représenté, en plus de sa matrice de contrôle de parité, sous une forme graphique. Cette représentation est appelée graphe de Tanner, ou plus régalement graphe factoriel. Un graphe factoriel contient deux types de nœuds, les nœuds de données et les nœuds fonctionnels. Deux nœuds sont reliés par une branche. Dans le cas des codes LDPC, les nœuds de données représentent le mot de code et les nœuds fonctionnels correspondent aux contraintes de parité.

Nous appellerons donc les nœuds fonctionnels, nœud de contrôle de parité. Un nœud de données  $i$  est relié à un nœud de contrôle  $j$  par une branche, si: L'élément correspondant à la  $j^{\text{ème}}$  ligne et la  $i^{\text{ème}}$  colonne de la matrice de contrôle de Parité est non nulle. Par convention, les nœuds de données seront représentés par des Cercles et les nœuds de contrôle par des carrés. Un nœud de données correspondant à un bit du mot de code transmis sera représenté par un cercle blanc. Si un bit du mot de code n'est pas transmis (on parle de bit poinçonné), le nœud sera représenté par un cercle plein noir et appelé nœud poinçonné ou nœud caché.

Exemple: La figure II.4 représente le graphe de la matrice de contrôle de parité définie dans l'exemple précédent. Le mot de code transmis est le vecteur  $x$  ou le bit  $x_6$  a été poinçonné. [10]

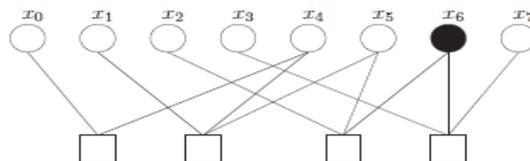


Figure II.4 : Graphe factoriel d'un code LDPC

Le graphe factoriel est une représentation graphique très simple du code. Ce graphe permet notamment d'illustrer les algorithmes de décodage associés aux codes LDPC qui seront présentes par la suite.

Une famille de codes LDPC peut se décrire en termes de taux de connexions des équations de parité. Quand le nombre de "1" par ligne et le nombre de "1" par colonne est constant, on parle de code LDPC régulier. Par conséquent chaque bit du mot de code participe à un même nombre d'équations de parité. De même, chacune des équations de parité utilise le même nombre de bits. Par extension, les codes LDPC irréguliers sont des codes définis par des matrices de contrôle de parité où le nombre de "1" par ligne et/ou par colonne n'est pas constant. Pour décrire ces codes, il est d'usage de spécifier l'irrégularité d'un code à travers deux polynômes  $\lambda(x)$  et  $\rho(x)$  :

$$\lambda(x) = \sum_{i \geq 1} \lambda_i x^{i-1} \quad (16)$$

$$\rho(x) = \sum_{i \geq 2} \rho_i x^{i-1} \quad (17)$$

Où  $\lambda_i$  (resp.  $\rho_i$ ) caractérise la proportion du nombre de branches connectées aux nœuds de données (resp. nœuds de contrôle) de degré  $i$  par rapport au nombre total de branches. Le degré est défini comme le nombre de branches connectées à un nœud.

On peut relier le profil d'irrégularité du code au rendement de codage de la façon suivante :

$$R \geq 1 - \frac{\sum_{i \geq 1} \frac{\lambda_i}{i}}{\sum_{i \geq 2} \frac{\rho_i}{i}} \quad (18)$$

Il y a égalité quand la matrice de contrôle de parité est de rang plein. On peut aussi caractériser l'irrégularité d'un code à travers deux autres polynômes,  $\tilde{\lambda}(x)$  et  $\tilde{\rho}(x)$  qui caractérise la proportion de nœuds de même degré (plutôt que la proportion de branches) de la manière suivante :

$$\tilde{\lambda}(x) = \sum_{i \geq 1} \tilde{\lambda}_i x^{i-1} \quad (19)$$

$$\tilde{\rho}(x) = \sum_{i \geq 2} \tilde{\rho}_i x^{i-1} \quad (20)$$

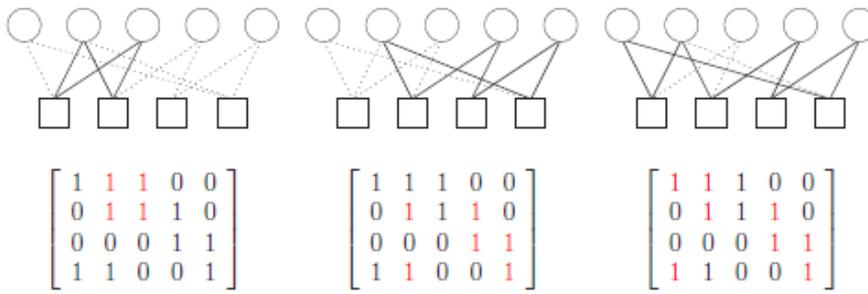


Figure II.5 : Exemple de cycles de longueur 4, 6 et 8.

Les coefficients  $\tilde{\lambda}_i$  et  $\tilde{\rho}_i$  de ces polynômes représentent la proportion de nœuds de données et de contrôle de degré  $i$ . Les deux représentations polynômiales sont reliées par :

$$\tilde{\lambda}_i = \frac{1}{i} \frac{\lambda_i}{\sum_{i \geq 1} \lambda_i / i} \quad (21)$$

$$\tilde{\rho}_i = \frac{1}{i} \frac{\rho_i}{\sum_{i \geq 2} \rho_i / i} \quad (22)$$

Exemple: le code représenté figure II.6 a une distribution des degrés égale a :

$$\tilde{\lambda}_i = \frac{5}{8} + \frac{3}{8}x$$

$$\tilde{\rho}_i = \frac{1}{4}x + \frac{3}{4}x^2$$

Qui peut aussi être exprimée de la façon suivante :

$$\lambda_i = \frac{5}{11} + \frac{6}{11}x$$

$$\rho_i = \frac{2}{11}x + \frac{9}{11}x^2$$

La représentation par graphe d'un code LDPC nous permet d'introduire la notion de cycle. Un cycle existe dans un graphe dès lors qu'il y a un chemin pour quitter et revenir à un nœud sans passer par les mêmes branches. Le nombre de branches traversées détermine la longueur du cycle. Un graphe sans cycle est appelé un arbre.

La figure II.6 représente des cycles de longueurs différentes. La position des éléments non nuls dans la matrice de contrôle de parité est également représentée. [10]

### II.5.3 Décodage des codes LDPC

L'algorithme de décodage itératif présenté initialement par Gallager, revue ensuite par MacKay dans le cadre de la théorie des graphes, est connu sous le nom d'algorithme de propagation de croyance (Belief Propagation (BP)). Cet algorithme peut être vu comme un algorithme d'échange d'information entre les nœuds du graphe à travers les branches. Ces messages transitant de nœuds en nœuds portent une information probabiliste sur l'état des nœuds.

Le principe de la propagation de croyance est l'application directe de la règle de Bayes sur chaque bit d'une équation de parité. La vérification de parité permet de calculer une estimation de chaque bit. Ces estimations, formant des messages se propageant sur les branches du graphe, sont alors échangées itérativement afin de calculer une information a posteriori sur chaque bit. Dans le cas d'une propagation de croyance sur un graphe sans cycle, les messages échangés sont indépendants, ce qui conduit au calcul simple et exact des probabilités a posteriori : l'algorithme est dans ce cas optimal. Dans le cas des codes LDPC, le graphe factoriel présente des cycles.

Dans ces conditions, l'hypothèse de messages indépendants n'est plus valide. Cependant, plus le graphe est creux (c'est à dire moins la matrice de contrôle de parité est dense), plus l'approximation d'un graphe sans cycle devient valide. C'est donc sous cette hypothèse que l'algorithme de décodage est décrit.

Si on considère une équation de parité  $c$  faisant intervenir un ensemble de bits  $V_c$ , la règle de Bayes appliquée au bit  $v$  permet d'exprimer les probabilités suivantes conditionnellement à la séquence reçue  $\{y\}$  :

$$\Pr(v = 0 \mid \{y\}) = \Pr(\sum_{v' \in V_c/v} v' = 0 \mid \{y\}) \quad (23)$$

$$\Pr(v = 1 \mid \{y\}) = \Pr(\sum_{v' \in V_c/v} v' = 1 \mid \{y\}) \quad (24)$$

Où la somme est réalisée modulo 2. Gallager a démontré que ces deux probabilités sont égales à :

$$\Pr(v = 0 \mid \{y\}) = \frac{1 + \prod_{v' \in V_c} (1 - 2 \Pr(v' = 1 \mid \{y\}))}{2} \quad (25)$$

$$\Pr(v = 1 \mid \{y\}) = \frac{1 - \prod_{v' \in V_c} (1 - 2 \Pr(v' = 1 \mid \{y\}))}{2} \quad (26)$$

En utilisant la relation :

$$\tanh\left(\frac{1}{2} \ln \frac{1 - \Pr(v'=1 | \{y\})}{\Pr(v'=1 | \{y\})}\right) = 1 - 2 \Pr(v' = 1 | \{y\}) \quad (27)$$

On peut calculer le rapport de vraisemblance suivant :

$$\frac{\Pr(v=0 | \{y\})}{\Pr(v=1 | \{y\})} = \frac{1 + \prod_{v' \in V_{c/v}} \tanh\left(\frac{1}{2} \ln \frac{\Pr(v'=0 | \{y\})}{\Pr(v'=1 | \{y\})}\right)}{1 - \prod_{v' \in V_{c/v}} \tanh\left(\frac{1}{2} \ln \frac{\Pr(v'=0 | \{y\})}{\Pr(v'=1 | \{y\})}\right)} \quad (28)$$

Qui peut être simplifié par :

$$\tanh\left(\frac{1}{2} \ln \frac{\Pr(v=0 | \{y\})}{\Pr(v=1 | \{y\})}\right) = \prod_{v' \in V_{c/v}} \tanh\left(\frac{1}{2} \ln \frac{\Pr(v'=0 | \{y\})}{\Pr(v'=1 | \{y\})}\right) \quad (29)$$

La fonction  $\tanh$  étant une fonction monotone impaire, on peut décomposer la relation précédente de la manière suivante :

$$\text{sign}\left(\ln \frac{\Pr(v=0 | \{y\})}{\Pr(v=1 | \{y\})}\right) = \prod_{v' \in V_{c/v}} \text{sign}\left(\ln \frac{\Pr(v'=0 | \{y\})}{\Pr(v'=1 | \{y\})}\right) \quad (30)$$

$$\tanh \left| \frac{1}{2} \ln \frac{\Pr(v=0 | \{y\})}{\Pr(v=1 | \{y\})} \right| = \prod_{v' \in V_{c/v}} \tanh \left| \frac{1}{2} \ln \frac{\Pr(v'=0 | \{y\})}{\Pr(v'=1 | \{y\})} \right| \quad (31)$$

En utilisant le fait que :

$$f(x) = -\ln \tanh\left(\frac{x}{2}\right) = \ln \frac{\exp x + 1}{\exp x - 1} = f^{-1}(x) \quad (32)$$

On peut exprimer la valeur absolue du rapport de vraisemblance dans l'espace logarithmique par :

$$\left| \ln \frac{\Pr(v=0 | \{y\})}{\Pr(v=1 | \{y\})} \right| = f\left(\sum_{v' \in V_{c/v}} f\left(\left| \ln \frac{\Pr(v'=0 | \{y\})}{\Pr(v'=1 | \{y\})} \right|\right)\right) \quad (33)$$

## II.6 BCH

### II.6.1. Caractéristiques générales de codes BCH

Soit  $m \geq 3$ . Les codes BCH binaires sont des codes possédant les caractéristiques suivantes :

- Longueur du code :  $n = 2^m - 1$ . (34)
- Dimension du code :  $k$

- Nombre de bits de vérification :  $n - k \leq mt$  avec  $t < 2^{m-1}$
- Distance minimum :  $d_{\min} \geq 2t + 1$  (35)

On pourra donc détecter un maximum de  $2t$  erreurs et corriger un maximum de  $t$  erreurs.

### II.6.2 Définition

Un code cyclique de longueur défini sur  $F_q$  est appelé code BCH de distance prescrite (designed distance) si son polynôme générateur  $g(x)$  est plus petit multiple commun (PPCM) des polynômes minimaux des racines  $\omega^j, \omega^{j+1}, \dots, \omega^{j+\delta-2}$  pour tout entier  $j$ , et où  $\omega$  est une  $n^{\text{ème}}$  racine primitive de l'unité.

### II.6.3 Construction

Nous allons construire un code binaire de longueur  $n$  impaire.

De la même façon que pour les codes cycliques, on commence par factoriser le polynôme  $x^n - 1$  sur  $F_2$  en produit de facteurs irréductibles.

On obtient donc :

$$x^n - 1 = f_1(x) \cdot f_2(x) \cdot \dots \cdot f_t(x) \quad (36)$$

Soit  $\omega$  une  $n^{\text{ième}}$  racine de l'unité, c'est-à-dire que  $\omega$  est une racine de  $x^n - 1$ .

Chaque polynôme facteur de  $x^n - 1$ ,  $f_j(x)$ ,  $1 \leq j \leq t$ , admet une racine  $\omega^{i_j}$ . On obtient donc un ensemble de racines  $\{\omega^{i_1}, \omega^{i_2}, \dots, \omega^{i_t}\}$ .

En choisissant une combinaison de plusieurs facteurs  $f_j(x)$ ,  $1 \leq j \leq t$ , on obtient un ensemble de racines. On cherche à choisir  $s$  facteurs  $f_1(x), f_2(x), \dots, f_s(x)$ , de sorte d'obtenir une suite de racines consécutives de la forme  $\{\omega^j, \omega^{j+1}, \dots, \omega^{j+d-2}\}$ .

Etant donné qu'en général on choisit  $j = 1$ , on obtiendra l'ensemble de racines  $\{\omega, \omega^2, \dots, \omega^{d-1}\}$ .

On obtient donc un code qui a une distance minimum au moins égale à  $d$ , et où le polynôme générateur sera défini comme étant le produit des  $f_j(x)$ ,  $1 \leq j \leq s$ , choisis :

$$G(x) = f_1(x) \cdot f_2(x) \cdot \dots \cdot f_s(x) \quad (37)$$

## Chapitre II : *Introduction des techniques de modulation et codage*

---

Exemple :

Nous chercherons à construire un code de longueur  $n = 15$ . Pour cela, on factorise  $x^{15} - 1$  :

$$x^{15} - 1 = (x + 1) (x^2 + x + 1) (x^4 + x + 1) (x^4 + x^3 + 1) (x^4 + x^3 + x^2 + x + 1).$$

En combinant ces polynômes, on obtient des codes de distance et de dimensions différentes:

Polynôme	Racines
$x + 1$	1
$x^2 + x + 1$	$\omega, \omega^2$
$x^4 + x + 1$	$\omega, \omega^2, \omega^4, \omega^8$
$x^4 + x^3 + 1$	$\omega, \omega^2, \omega^4, \omega^8$
$x^4 + x^3 + x^2 + x + 1$	$\omega, \omega^3, \omega^6, \omega^9, \omega^{12}$

Exemple :

$$\begin{aligned} \text{Soit : } g(x) &= (x^4 + x + 1) (x^4 + x^3 + x^2 + x + 1) \\ &= x^8 + x^7 + x^6 + x^4 + 1 \end{aligned}$$

On pose  $\omega$  une racine de  $x^4 + x + 1$ .

$G(x)$  admet donc les racines  $\omega, \omega^2, \omega^3, \omega^4, \omega^6, \omega^8, \omega^9, \omega^{12}$ , et on a donc :

$$G(\omega) = g(\omega^2) = g(\omega^3) = g(\omega^4) = 0 \tag{38}$$

On a donc construit un code de dimension  $k = n - \deg(g) = 15 - 8 = 7$ , et de distance minimum au moins égale à  $d = 5$ .

On pose  $\omega$  une racine de  $x^4 + x + 1$ .

$G(x)$  admet donc les racines  $\omega, \omega^2, \omega^3, \omega^4, \omega^6, \omega^8, \omega^9, \omega^{12}$ , et on a donc :

$$G(\omega) = g(\omega^2) = g(\omega^3) = g(\omega^4) = 0 \quad (39)$$

On a donc construit un code de dimension  $k = n - \deg(g) = 15 - 8 = 7$ , et de distance minimum au moins égale à  $d = 5$ .

### **II.7 conclusion**

Dans ce chapitre nous avons présenté les codes RS, LDPC, et le codage BCH. Nous avons dans un premier temps présenté les caractéristiques générales de ces codes, puis nous avons donné la façon de les construire.

# Chapitre III

Programmation de  
Raspberry Pi à l'aide Simulink.

## CHAPITRE III : PROGRAMMATION DE RASPBERRY PI A L'AIDE DE SIMULINK

### III.1 Introduction

Dans ce chapitre, nous introduisons une petite description sur le Raspberry pi ainsi qu'un historique sur le développement de ces modèles de cartes ensuite présenté. Puis nous exposons les composants de Raspberry pi. Enfin, nous introduisons les étapes de l'installation de supporte package de Simulink pour le Raspberry pi.

### III.2 Description de Raspberry pi



Figure III.1 : Modale de Raspberry Pi

RASPBERRY Pi est un ordinateur intégré (figure 19) , fabriqué à partir d'une seule puce électronique contenant les composants traditionnels de l'ordinateur, un processeur à un seul cœur de 700MHZ et un GPU double cœur de 250MHZ capable de lire des films haute définition, avec RAM aller jusqu'à 512MHZ.

EN plus des sorties de contrôle numériques qui peuvent contrôler diverses composants électriques et électroniques tels que les microcontrôleurs, toutes ces capacités

## Chapitre III : Programmation de RASPBERRY Pi à l'aide de Simulink

---

sont sur une micro puce, connue sous le nom de SOC (Système On Chip). Ce petit ordinateur fonctionne sur des systèmes Linux open source.

Il a une taille de 8.6 cm x 4.5 cm et pèse environ 45 grammes, ce qui en fait l'un des panneaux d'ordinateur les plus légers dans la planète.

On peut utiliser RASPBERRY Pi comme n'importe quel ordinateur traditionnel pour surfer sur internet et envoyer des e-mails et même éditer des fichiers via le bureau libre office... etc. On peut par exemple faire :

- La conception de système de contrôle de maison intelligente « Smart Home Auto motion ».
- La fabrication de robots de sous-marines et de drones « ROV and UAV Robots ».
- Des applications de surveillance telles que des caméras de travail pour diffuser des vidéos et des images à distances « Caméra Streamers ».
- Surveillance de l'environnement tel que le système de surveillance de la température et de l'humidité à distance « Remote Monitor ».
- Smart TV.
- Différents serveurs Linux tel que Linux : serveur http, FTP, MSQL, SSH, etc...
- Supercalculateurs « Super computers ».

L'utilisation de RASPBERRY Pi donne de meilleures opportunités que de voir seulement les résultats de simulation. De plus, il encourage la pensée créative et l'expérimentation de nouvelles applications. Cependant, le dispositif d'application comme RASPBERRY Pi a des capacités limitées et donc le logiciel doit être développé en gardant à l'esprit les limitations.

### III.3 L'historique de Raspberry pi

RASPBERRY Pi, tel qu'illustré à la figure 2.1, est un ordinateur de bord unique ayant un format aussi petite qu'une carte de crédit de la fondation RASPBERRY Pi. Une idée de production RASPBERRY Pi a commencé en 2006 avec la prise de conscience de l'absence de la jeune génération des connaissances sur le fonctionnement de l'ordinateur. Un

## Chapitre III : Programmation de RASPBERRY Pi à l'aide de Simulink

---

groupe d'universitaires et d'ingénieurs de l'Université d'Ottawa de L'Université de Cambridge a donc décidé de mettre au point un très petit ordinateur qui permet d'effectuer les tâches suivantes que tout le monde peut se permettre d'acheter pour créer un environnement d'apprentissage dans la programmation. Le RASPBERRY Pi projet est devenu prometteur avec l'apparition de bon marché et puissant des processeurs mobiles avec de nombreuses fonctionnalités avancées permettant un développement possible de RASPBERRY Pi qui s'est poursuivi sous la fondation RASPBERRY Pi spécialement créée à cet effet avec le premier produit lancé en 2012 [pomyen].

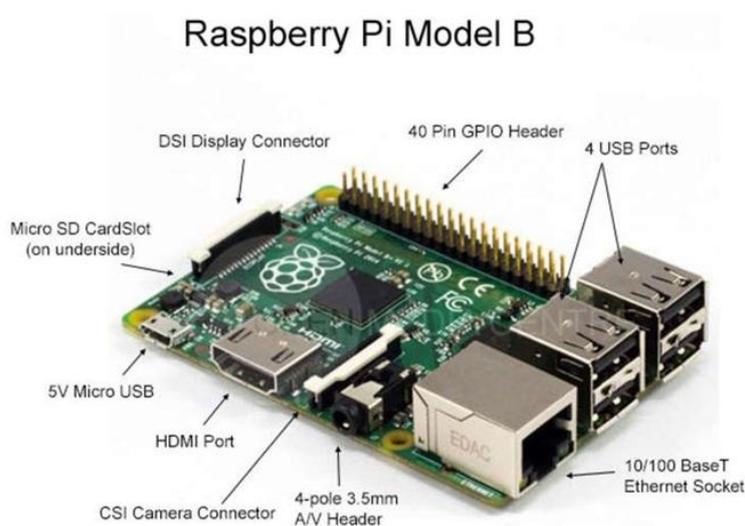


Figure III.2 : Les portes de Raspberry Pi

Le développement de l'ordinateur RASPBERRY Pi est un processus continu. Jusqu'au printemps 2015, la fondation RASPBERRY Pi a publié cinq modèles en deux générations d'ordinateurs. La première génération se compose de quatre modèles, comme le montre la figure III.2. D'abord le modèle B suivi respectivement du modèle A, du modèle B+ et du modèle A+. Ces deux derniers modèles sont des versions mises à jour de leurs versions précédentes pour rendre l'ordinateur plus efficace et plus pratique pour les utilisateurs, en particulier en ayant une consommation d'énergie plus faible et plus de ports USB (Universal Serial Bus). La deuxième génération se compose d'un seul modèle appelé Raspberry Pi 2

## Chapitre III : Programmation de RASPBERRY Pi à l'aide de Simulink

---

modèle B dont la spécification est basée sur Raspberry Pi modèle B+ mais avec un CPU plus rapide et plus de mémoire [pomyen].

RASPBERRY Pi modèle B, comme le montre la figure III.3 (b), a été publié au début de 2012 avec la spécification de 256 Mo de RAM, deux ports USB et un port Ethernet. Plus tard dans la même année, dans une nouvelle version, la quantité de RAM a été augmentée à 512 Mo, suivie de la sortie de la carte à spécifications inférieures, modèle A, comme le montre la Figure III.3 (a). Le modèle A est sorti avec la même quantité de RAM que l'ancien modèle B à 256 Mo, mais avec un port USB et pas de port Ethernet. Ces ordinateurs de première génération utilisent Broadcom SoC, BCM2835, qui intègre un processeur ARM1176JZF-S à cœur unique de 700 MHz, un GPU VideoCore IV et une variété de périphériques. Alors que le modèle B peut être utilisé dans n'importe quelle application, le modèle A, moins cher, est utile dans des applications spécifiques qui nécessitent un poids léger et une faible consommation d'énergie, comme la robotique ou tout service de médias portables.

## Chapitre III : Programmation de RASPBERRY Pi à l'aide de Simulink

---

En 2014, une version améliorée des deux modèles précédents a été lancée, sous les noms B+ et A+, comme le montrent les figures III.3(c) et III.3 (d). Bien que les deux nouveaux modèles aient le même processeur, GPU et RAM que leurs versions précédentes, plusieurs mises à jour et de nouvelles fonctionnalités ont été ajoutées, telles qu'une prise mémoire micro SD remplaçant une fente pour carte SD, une version améliorée de GPIO de 26 à 40 broches, un son à faible bruit et une consommation d'énergie inférieure de 0,5 à 1 W en ayant un régulateur à découpage au lieu d'un régulateur linéaire. De plus, le modèle A+ est d'environ 2 cm plus court que son prédécesseur, tandis que le modèle B+ fournit quatre ports USB au lieu de deux dans le modèle B [pomyen].

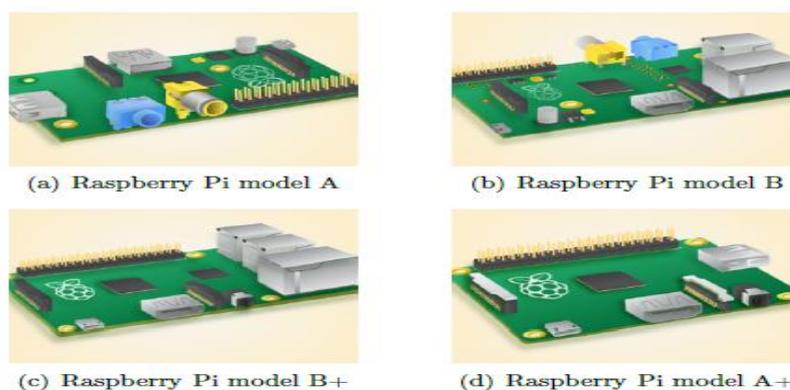


Figure III.3 : LES MODELES DE RASPBERRY PI

## Chapitre III : Programmation de RASPBERRY Pi à l'aide de Simulink

---

Le tableau III.1 représente les principales caractéristiques des modèles RASPBERRY Pi.

Raspberry Pi	Génération 1				Génération 2
Spécification	Model A	Model B	Model A+	Model B+	Model B
Power	300mA	700mA	200mA	600mA	900mA
Ethernet Port	No	Yes	No	Yes	Yes
USB Port	1	2	1	4	4
GPIO	26	26	40	40	40
SD Carte Slot	SD	SD	MicroSD	MicroSD	MicroSD
SoC	BCM2835	BCM2835	BCM2835	BCM2835	BCM2836
CPU	700MHz ARM11 Single-core	700MHz ARM11 Single-core	700MHz ARM11 Single-core	700MHz ARM11 Single-core	900MHz ARM Cortex-A7 Quad-core
RAM	256MB	512MB	256MB	512MB	1GB

Tableau III.1 : les caractéristiques des modèles raspberry

## Chapitre III : Programmation de RASPBERRY Pi à l'aide de Simulink

---

Bien qu'un nouveau modèle RASPBERRY Pi offre plus de caractéristiques, il est préférable de savoir pour quelles applications ou projets la carte sera utilisée afin de choisir la plus efficace en termes de puissance et de coût.

### III.4 Composants de base

#### III.4.1 Ports USB

Le RASPBERRY Pi possède des ports USB permettent de connecter des claviers, des souris, des dongles WiFi et des clés USB. Comme les ports ne fournissent pas beaucoup d'énergie, si on veut ajouter un hub USB au Raspberry Pi, on doit trouver un qui vient avec une alimentation externe.



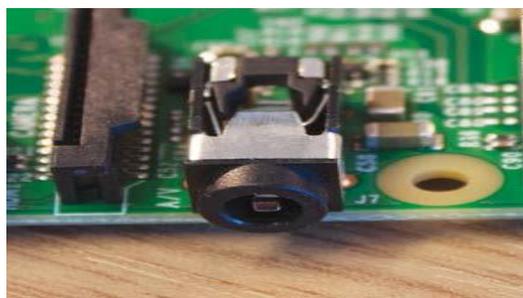
#### III.4.2 Port Ethernet

Ce port sert à connecter le Raspberry pi à l'Internet via un câble Ethernet, il est semblable à celui qu'on trouve à l'arrière d'un routeur. Cette méthode est plus facile à mettre en place que le WiFi et peut fournir un accès Internet plus rapide, mais on est limité par la longueur du câble.



#### III.4.3 Connecteur audio

Est une sortie audio via une prise casque de 3,5 mm qui prend également en charge une sortie vidéo analogique pour les modèles A+, B+ et RASPBERRY Pi modèle B. Pour les modèles précédents, A et B, le connecteur n'est que pour une sortie audio.

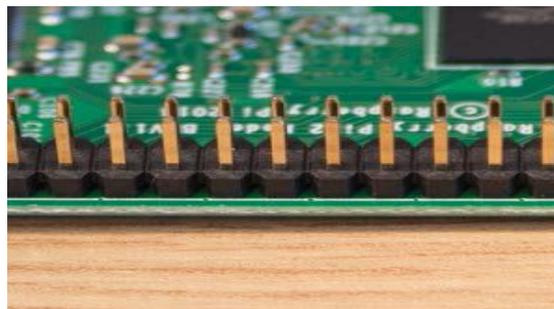


## Chapitre III : Programmation de RASPBERRY Pi à l'aide de Simulink

---

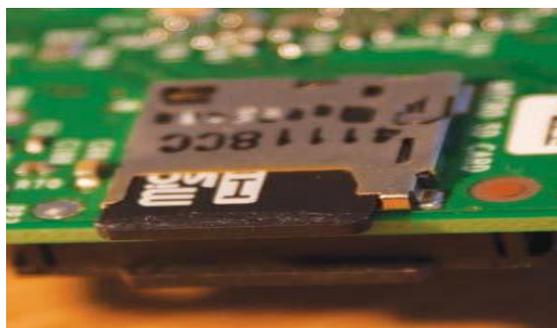
### III.4.4 GPIO

Est un ensemble de broches de connecteurs d'entrée et de sortie universelles ont diverses fonctions, elles permettent de connecter au RASPBERRY Pi avec un circuit électronique. Vous pouvez alors programmer le Pi pour contrôler le circuit et faire des choses importantes. Il y a 26 épines dans les modèles A et B et 40 épines dans les modèles A+, B+ et Raspberry Pi 2 modèle B.



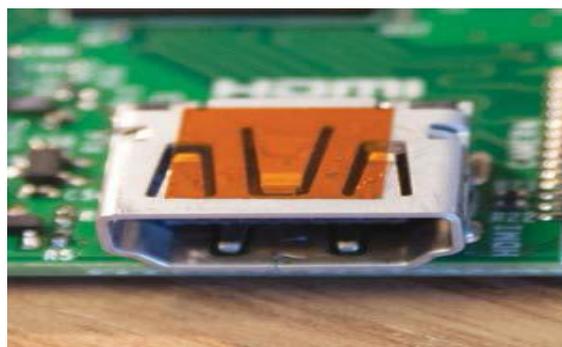
### III.4.5 Fente pour carte MicroSD

Cette fente est utilisée pour contenir une petite carte SD utilisée comme disque dur du RASPBERRY Pi dont laquelle le système d'exploitation sera installé.



### III.4.6 Port HDMI

C'est un port HDMI similaire à celui qui se trouve à l'arrière de la plupart des téléviseurs et moniteurs d'ordinateur modernes. On utilise un câble HDMI standard pour connecter le RASPBERRY Pi un écran, pour voir (et entendre) tout ce qu'il fait et pour configurer le Pi.

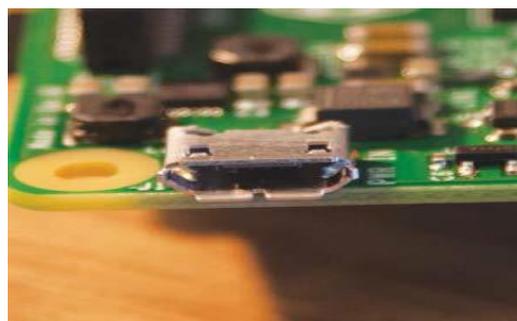


## Chapitre III : Programmation de RASPBERRY Pi à l'aide de Simulink

---

### III.4.7 Puissance

C'est un port de charge qu'on peut trouver dans un Smartphone. Ce port micro-USB signifie qu'on peut alimenter le Pi avec le bon type de chargeur de téléphone portable ou directement à partir d'un PC - cependant, il est préférable d'utiliser l'alimentation officielle RASPBERRY Pi pour s'assurer que le Pi reçoit suffisamment d'énergie.



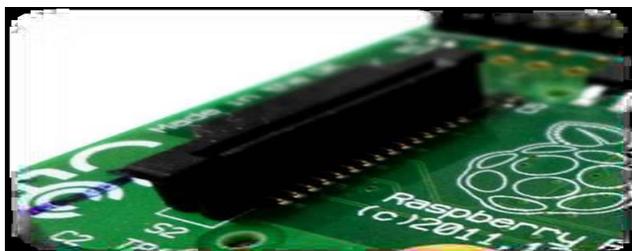
### III.4.8 Le connecteur d'interface série de caméra (CSI)

Est un connecteur pour un module de caméra, comme illustré à la figure (a). L'appareil caméra est spécialement conçu pour RASPBERRY Pi et peut être connecté avec un câble ruban fournissant un support à la fois pour la photographie et la vidéo HD.



### III.4.9 Port DSI display

Il est utilisé pour connecter des écrans tactiles tels que les smart phones et les tablettes.



## III.5 Programmation de Raspberry Pi à l'aide de Simulink

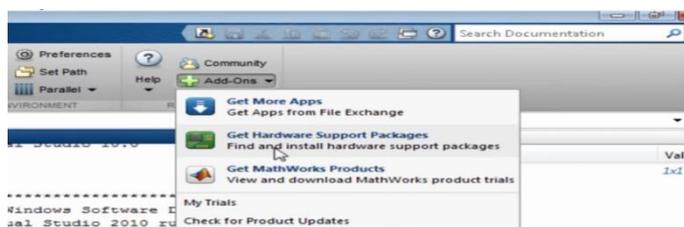
Il est possible d'utiliser MATLAB pour communiquer avec les cartes RASPBERRY Pi et ses périphériques en utilisant MATLAB Support Package for RASPBERRY Pi. Ainsi on peut programmer ces cartes pour exécuter nos algorithmes à l'aide de Simulink Support Package for RASPBERRY Pi Hardware. Le package de support génère du code à partir de notre modèle Simulink en un clic d'un bouton qui s'exécute ensuite sur la carte RASPBERRY Pi.

Et pour effectuer ses opérations on passe par plusieurs étapes :

## Chapitre III : Programmation de RASPBERRY Pi à l'aide de Simulink

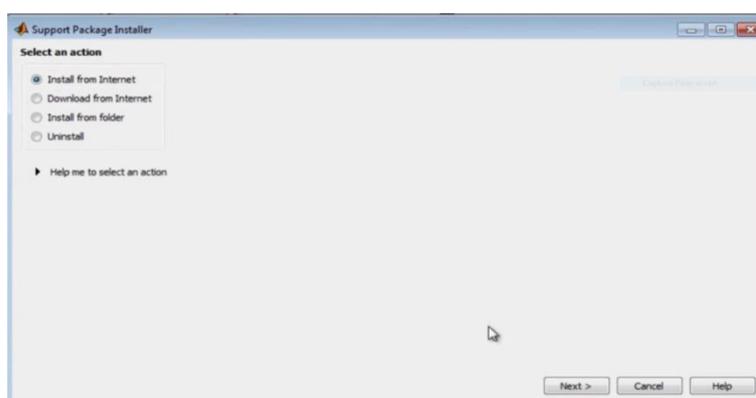
### III.5.1 Configuration de Package de support Simulink pour le Raspberry Pi

Pour obtenir le support package, premièrement on ouvre MATLAB et on clique sur le bouton Add-Ons qui situe en haut à droite et on choisit « Get hardware Support Packages ».

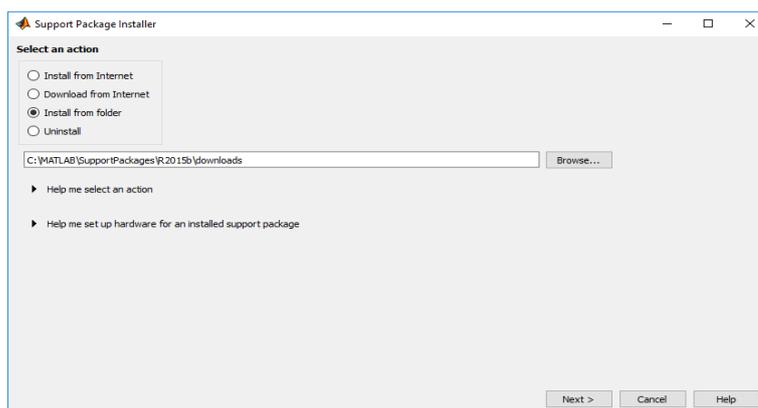


### III.5.2 L'installation du Support Package

Pour installer le support package de RASPBERRY Pi, soit on l'installe directement à partir d'internet en choisissant « Install From Internet » et il sera télécharger et installé, soit on l'installe à partir du dossier en choisissant « Install From

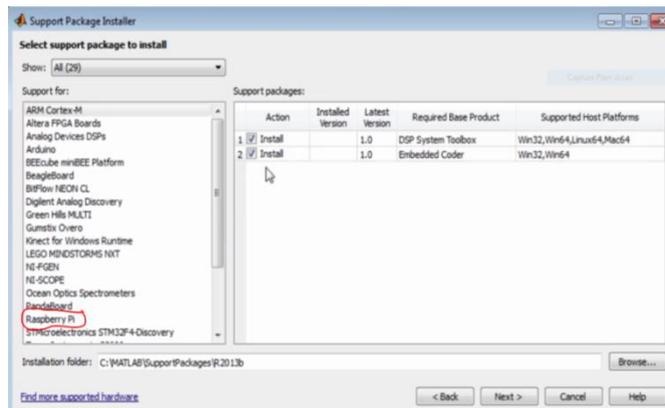


Folder » si on l'a déjà téléchargé. En choisissant par exemple l'installation directement à partir de l'internet alors on a choisi « Install From Internet ». En suivant les étapes suivantes :

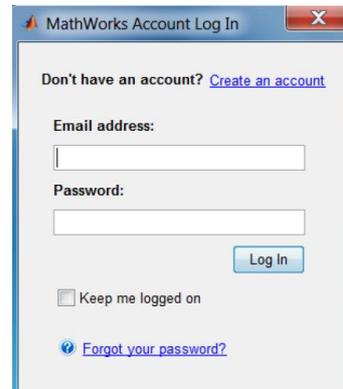


En cliquant sur next on obtient une liste de supports packages dans cette liste. On choisit RASPBERRY Pi pour installer les deux paquets en même temps.

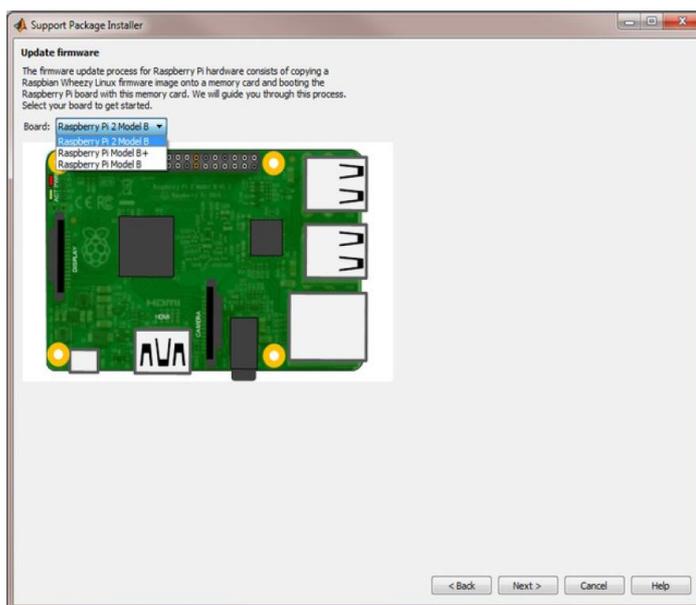
## Chapitre III : Programmation de RASPBERRY Pi à l'aide de Simulink



En cliquant sur « Next », une fenêtre s'affiche pour entrer l'adresse E-mail et le mot de passe du compte Matlab, si l'utilisateur n'a pas le compte, il a la possibilité de la créer pendant le processus d'installation, après avoir cliqué sur Login l'installation commence. Une fois l'installation de package terminé le programme commence à installer les pilotes ou les mises à jour « firmware ».



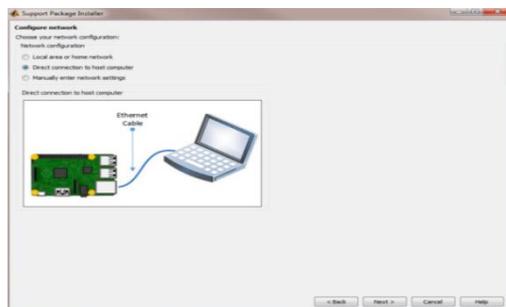
Le processus de mise à jour du firmware pour le matériel RASPBERRY Pi consiste à copier une image de firmware rasbian wheezy linux dans une carte mémoire et à démarrer le RASPBERRY Pi avec cette carte mémoire. L'étape obligatoire dans ce processus est de choisir le model de RASPBERRY Pi.



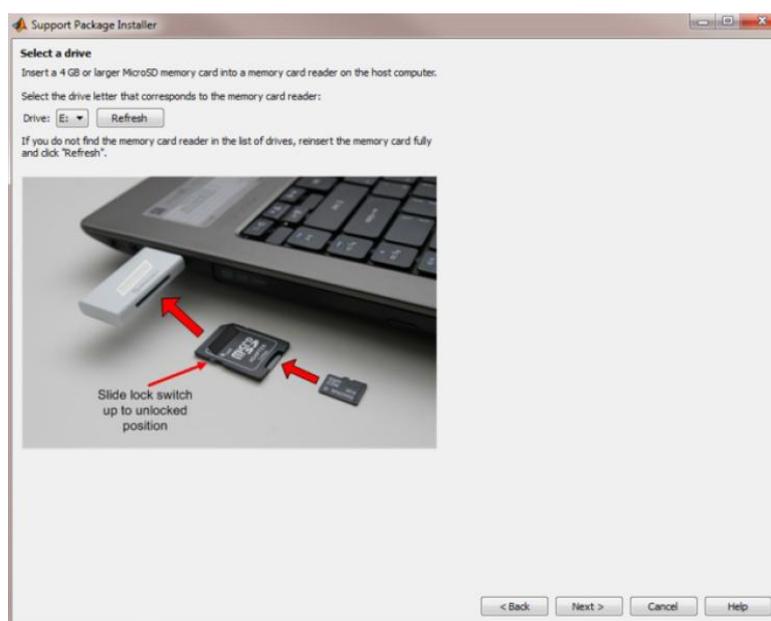
## Chapitre III : Programmation de RASPBERRY Pi à l'aide de Simulink

---

En cliquant sur «Next » une fenêtre s'affiche où on doit sélectionner « **Direct connection to host computer** ».

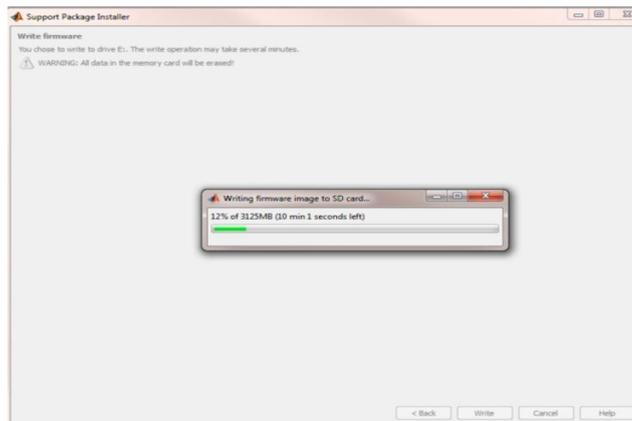


Une autre fenêtre s'affiche après avoir cliqué sur « Next » où on doit Sélectionner un lecteur de cartes MicroSD détectées par MATLAB et qui apparaîtront sous forme de liste. Si la carte mémoire MicroSD n'est pas détectée par MATLAB, mais est détectée par le système d'exploitation, il faut fermer MATLAB et le redémarrer en tant qu'administrateur. Pour continuer le processus, la commande « targetupdater » peut être utilisée dans MATLAB.



En cliquant sur « Next », une fenêtre s'affiche qui nous permet de sélectionner l'option d'écriture pour effacer les éléments existants sur la carte mémoire et faire flasher le dernier firmware requis par le Support Package.

## Chapitre III : Programmation de RASPBERRY Pi à l'aide de Simulink

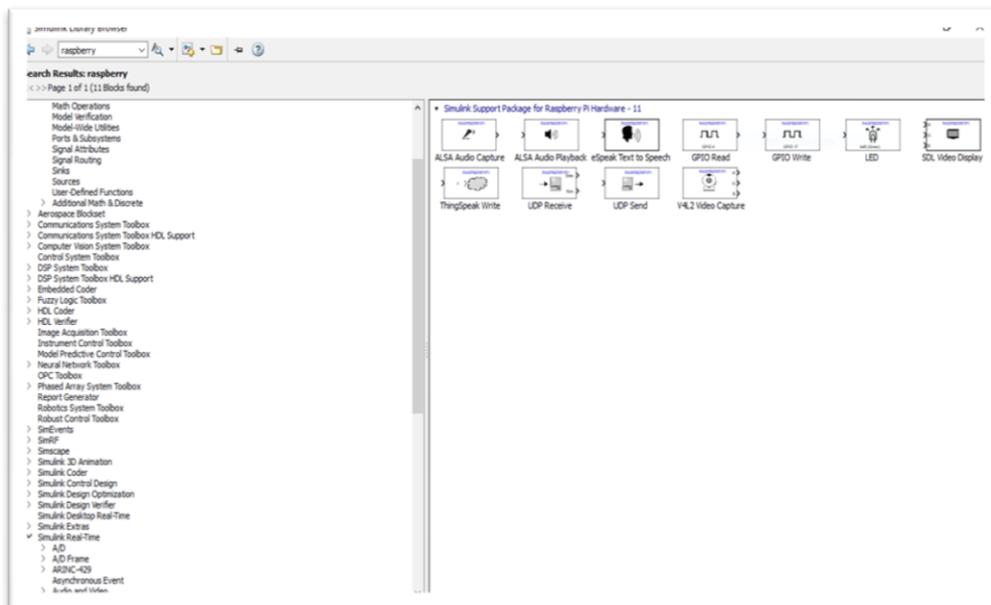


En fin, en cliquant sur « Next » le programme termine l'installation complet de support package pour Raspberry pi. Donc dans ce cas on peut implémenter un schéma Simulink dans la RASPBERRY.

### III.6 Teste de fonctionnement de la carte Raspberry Pi + Simulink

#### III.6.1 Vérification de l'installation du Support Package

Le support Package de Simulink pour le matériel Raspberry Pi doit se trouver dans la bibliothèque Simulink.



## Chapitre III : Programmation de RASPBERRY Pi à l'aide de Simulink

---

On peut aussi taper directement en ligne de commande MATLAB « raspberrypilib » pour vérifier la bibliothèque simulink. [[Technical Marketing Brian.McKay@mathworks.com](mailto:Technical_Marketing_Brian.McKay@mathworks.com)].

### III.6.2 Test de la connectivité (PC à Raspberry Pi)

Si on ne sait pas le HOSTNAME de Raspberry on tape sur la ligne commande "raspberrypi", on obtient les informations nécessaires de Raspberry pi.

La figure III.4 suivant illustre cette étape :



```
command window
New to MATLAB? See resources for Getting Started.
>> raspberrypi
ans =
LinuxServices with properties:
    HostName: '169.254.0.2'
    UserName: 'pi'
    Password: 'raspberry'
    BuildDir: '/home/pi/ofdm-pfe'
>> !ping 169.254.0.2
```

Figure III.4 : Les informations pour Raspberry Pi

Ensuite on tape en ligne de commande MATLAB "! ping + l'adresse IP" de raspberrypi, par exemple le cas où l'adresse IP de raspberrypi est 169.254.0.2 et voilà le résultat de test.

L'affichage montré sur la figure suivante montre que la carte est bien connectée.

## Chapitre III : Programmation de RASPBERRY Pi à l'aide de Simulink

---

```
Command Window
New to MATLAB? See resources for Getting Started.

>> !ping 169.254.0.2

Envoi d'une requête 'Ping' 169.254.0.2 avec 32 octets de données :
Réponse de 169.254.0.2 : octets=32 temps=3 ms TTL=64
Réponse de 169.254.0.2 : octets=32 temps=2 ms TTL=64
Réponse de 169.254.0.2 : octets=32 temps=2 ms TTL=64
Réponse de 169.254.0.2 : octets=32 temps=2 ms TTL=64

Statistiques Ping pour 169.254.0.2:
Paquets : envoyés = 4, reçus = 4, perdus = 0 (perte 0%),
Durée approximative des boucles en millisecondes :
Minimum = 2ms, Maximum = 3ms, Moyenne = 2ms
fx \>> |
```

### III.7 Conclusion

Dans ce chapitre nous avons présenté un aperçu général de la carte Raspberry Pi, ainsi qu'un historique sur le développement de ces modèles de carte. Ensuite nous avons présenté les composants de base de la carte Raspberry Pi. Enfin nous avons présenté toute les étapes de l'installation du support package de simulink pour le Raspberry Pi et la méthode de vérification de la connectivité cette carte avec le PC.

# Chapitre IV

Simulations et résultats

## CHAPITRE IV: SIMULATIONS ET RESULTATS

### IV.1 Introduction

Dans ce chapitre on va présenter la chaîne de transmission DVB-S2 simulée sous simulink. Dans la première partie nous allons décrire les différents blocs de cette chaîne. Ensuite nous allons présenter les résultats de simulation sous le logiciel simulink. Dans une deuxième étape et après avoir vérifié par simulation le bon fonctionnement de notre chaîne de transmission DVB-S2 nous allons l'implémenter sous la carte Raspberry Pi.

### IV.2 Modèle Simulink de la chaîne de transmission DVB-S2

La réalisation d'un simulateur pour une norme spécifique à savoir DVB-S2, consiste à assembler des blocs de base d'après une synoptique envisagée. Malgré l'apparence simpliste de cette conception, ils demandent beaucoup de précision afin d'ajuster leur fonctionnement.

La conception du simulateur de cette chaîne est basée sur le diagramme bloc de la figure IV.1 qui représente l'architecture du système DVB-S2 .

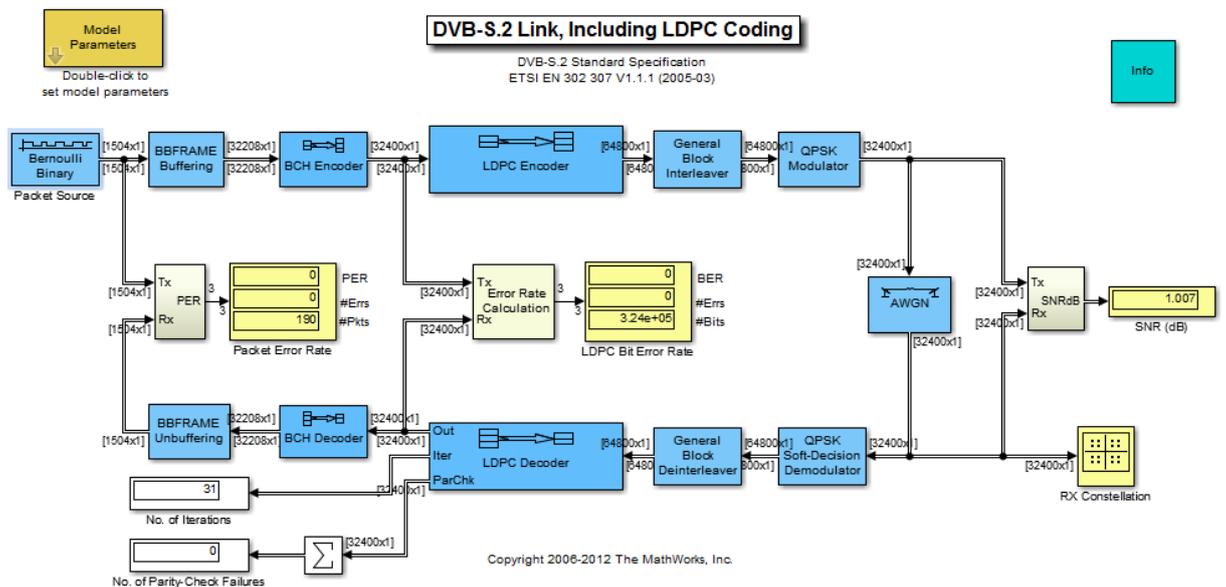


Figure IV.1 : Modèle Simulink de la chaîne de transmission DVB-S2

Les différents blocs de ce modèle sont :

### IV.2.1 Générateur Bernoulli

Le générateur de Bernoulli permet de générer les bits binaires aléatoires

La figure suivante représente les paramètres de réglage de ce bloc.

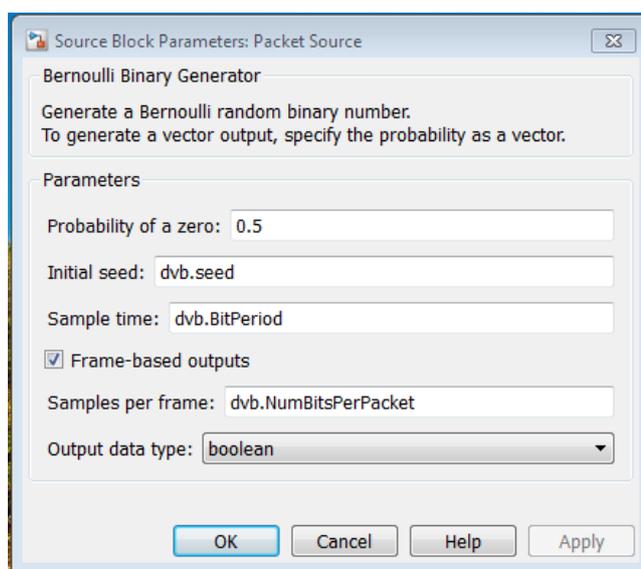


Figure IV.2 : Configuration du bloc générateur de Bernoulli

### IV.2.2 BBFRAME

Ce bloc est composé de deux sous bloc : Buffer et Zero Pad

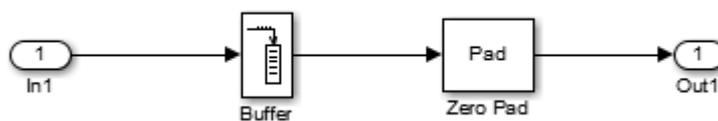


Figure 13 : les composants du bloc BBframe

## CHAPITRE IV: SIMULATIONS ET RESULTATS

- **Buffer** : il permet de convertir des échantillons scalaires en une sortie de trame à une fréquence plus faible. Nous pouvons également convertir les trames en une taille plus petite ou plus grande avec chevauchement optionnel.

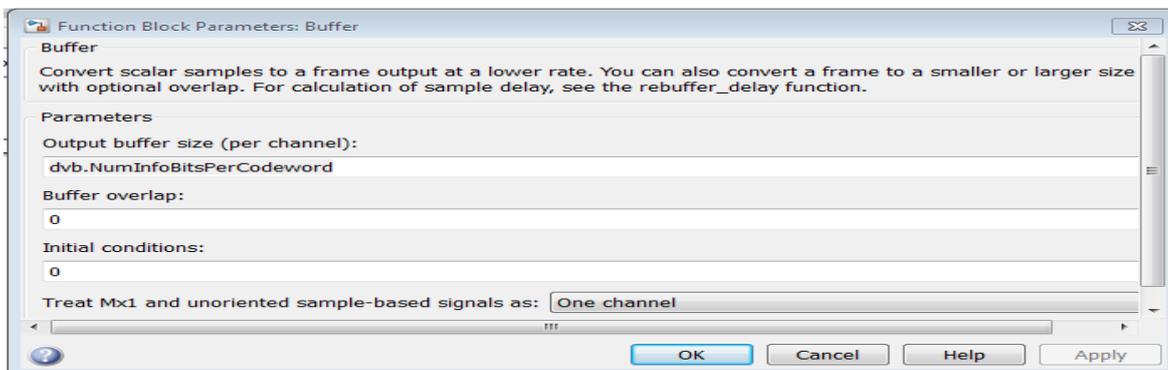


Figure IV.4 : configuration des parametre du bloc buffer

- **Zero Pad** : Ce bloc permet d'ajouter ou de préfixer une valeur constante à l'entrée selon les dimensions spécifiées. La troncature se produit lorsque les dimensions de sortie spécifiées sont plus courtes que les dimensions d'entrée correspondantes.

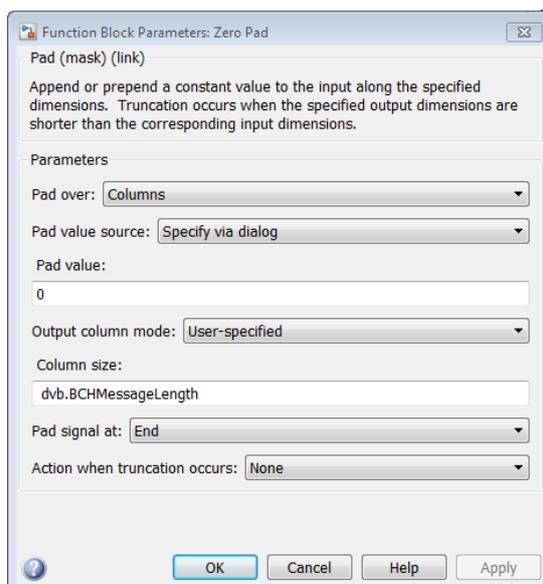


Figure IV.5 : CONFIGURATION des PARAMETRE DE BLOC Zero PAD

### IV.2.3 Décodeur/Encodeur BCH

Ce bloc encodeur permet d'encoder le message dans le vecteur d'entrée à l'aide d'un codeur  $(N, K)$  BCH avec le polynôme générateur de sens étroit. Ce bloc accepte un signal d'entrée vectoriel colonne avec un multiple entier de  $K$  éléments. Chaque groupe d'éléments d'entrée  $K$  représente un mot de message à coder. Les valeurs de  $N$  et  $K$  doivent produire un code BCH à sens étroit valide.

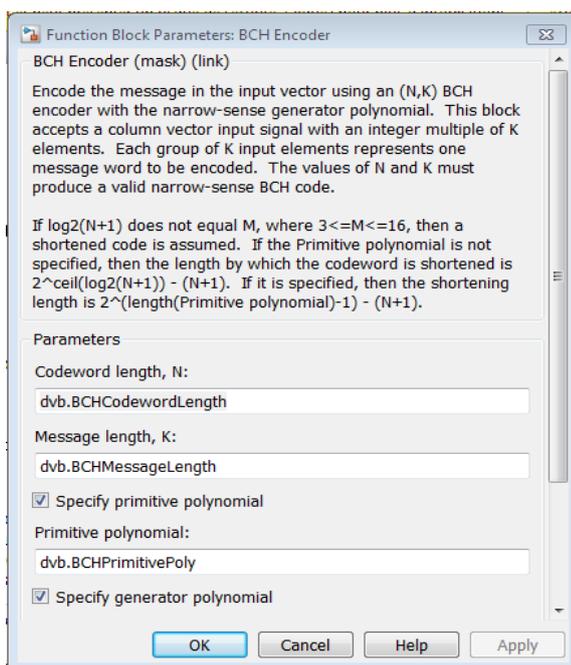


Figure IV.6 : CONFIGURATION DES PARAMETRES DE BLOC ENCODEUR BCH

Ce bloc décodeur permet de décodeur le message dans le vecteur d'entrée en utilisant un décodeur  $(N, K)$  BCH avec le polynôme générateur de sens étroit. Ce bloc accèpte un signal d'entrée vectoriel colonne avec un multiple entier de  $N$  éléments. Chaque groupe de  $N$  éléments d'entrée représente un mot de code à décodeur. Les valeurs de  $N$  et  $K$  doivent produire un code BCH à sens étroit valide.

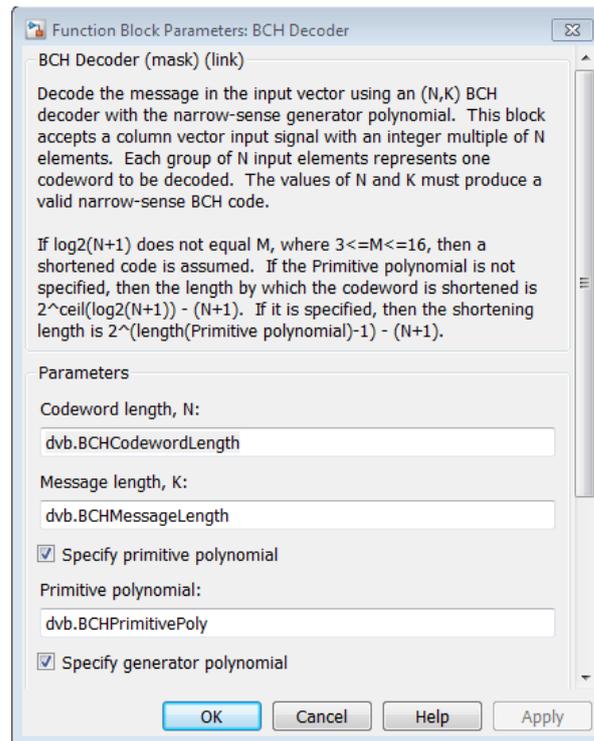


Figure IV.7 : configuration paramètre de bloc décodeur bch.

### IV.2.4 Encodeur/décodeur LDPC

- **Codeur LDPC :**

Ce codeur permet d'encoder un message à l'aide d'un code binaire de vérification de parité à faible densité. La matrice de contrôle de parité spécifie le code.

(N-K) et N sont le nombre de lignes et de colonnes dans la matrice de contrôle de parité. Ce bloc accepte un signal d'entrée vectoriel colonne avec des éléments K. La sortie est un vecteur colonne binaire avec N éléments.

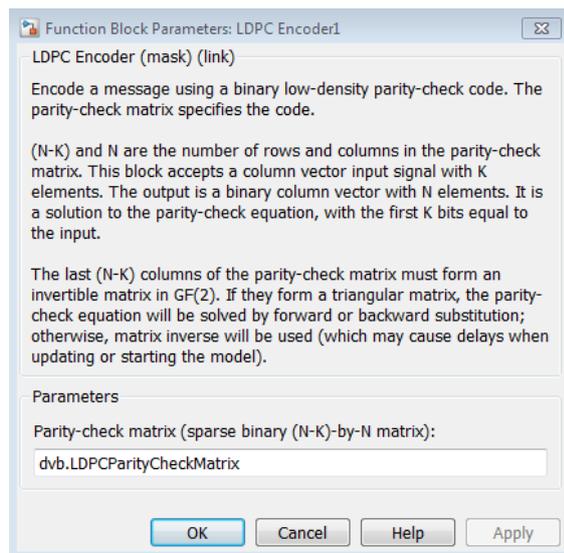


Figure IV.8 : Configuration des paramètres du bloc codeur LDPC

Les dernières colonnes  $(N-K)$  de la matrice de contrôle de parité doivent former une matrice inversable dans  $GF(2)$ . S'ils forment une matrice triangulaire, l'équation de contrôle de parité sera résolue par substitution vers l'avant ou vers l'arrière ; sinon, l'inverse de la matrice sera utilisé (ce qui peut causer des retards lors de la mise à jour ou du démarrage du modèle).

Décodage LDPC :

Il permet de décoder le code LDPC. La figure suivante représente les paramètres de configuration de ce bloc.

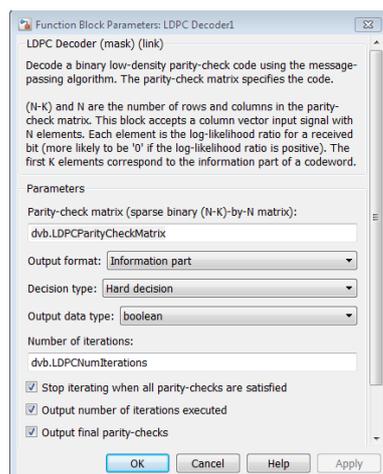


Figure IV.9 : configuration des paramètres de bloc décodeur LPDC.

### IV.2.5 General bloc interleaver :

Ce bloc permet réorganiser les éléments du vecteur d'entrée.  $y = u(k)$ . La longueur du vecteur de permutation doit correspondre à la largeur du signal d'entrée.

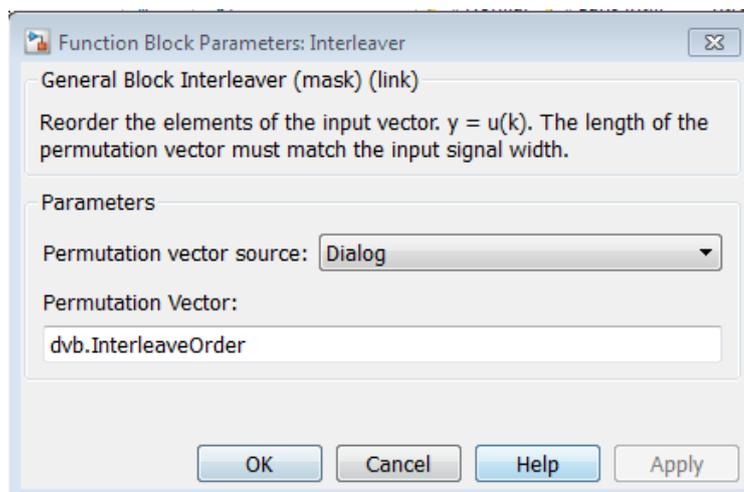


Figure IV.10 : Configuration des paramètres du bloc interleaver.

### IV.2.6 Bloc modulateur

La figure suivante représente le bloc modulateur.

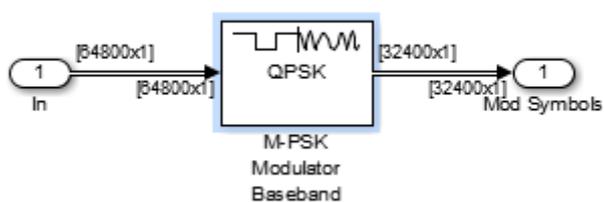


Figure IV.11 : composants du bloc modulateur

In et Out :

Permet de fournir un port d'entrée et de sortie pour un sous-système ou un modèle.

M-PSK Modulator:

Ce bloc permet de moduler le signal d'entrée à l'aide de la méthode de modulation par déphasage. Ce bloc accepte un signal d'entrée scalaire ou vectoriel. Le signal d'entrée peut être soit des bits, soit des entiers. Si le paramètre type d'entrée est réglé sur 'Bit', la largeur d'entrée doit être un multiple entier du nombre de bits par symbole.

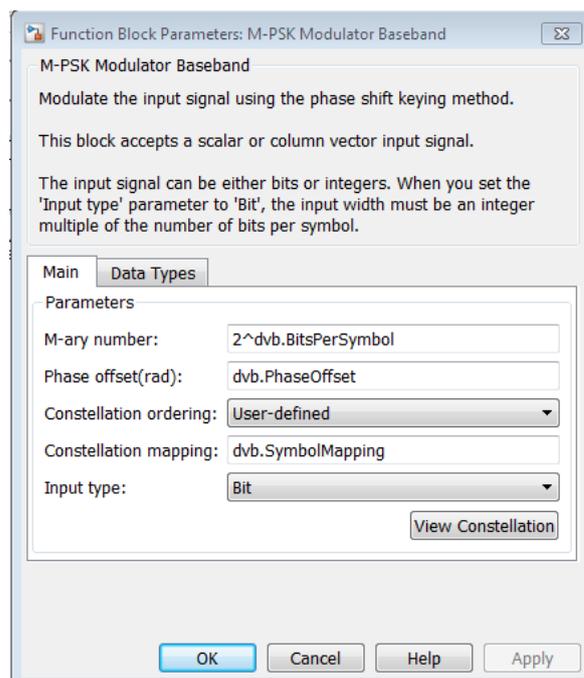


Figure IV.12 : configuration paramètre de bloc modulation QPSK

### IV.2.6 Canal AWGN

Il permet de générer un bruit blanc Gaussien additif (AWGN) qui va s'ajouter au signal envoyé.

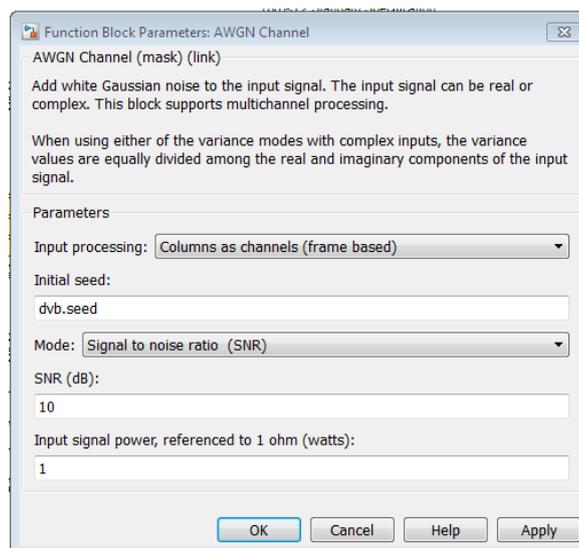


Figure IV.13 : configuration des paramètres du bloc AWGN

### IV.3 Résultats de Simulation de la chaîne DVB-S2 sous simulink :

Une représentation de la constellation QPSK à 4 états avant l'ajout du bruit produit par le canal avec une probabilité de zéros=0.5 et  $S/N=10$  est montrée sur la figure suivante.

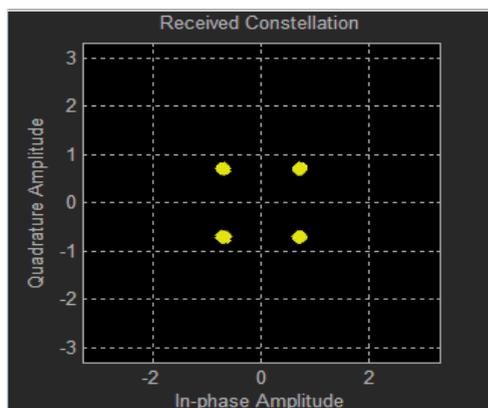


Figure IV.14 : Constellation de la QPSK à l'émission.

La constellation de la modulation numérique QPSK est prise à la réception avant la démodulation numérique avec la probabilité de zéros=0.5, et  $S/N=10$

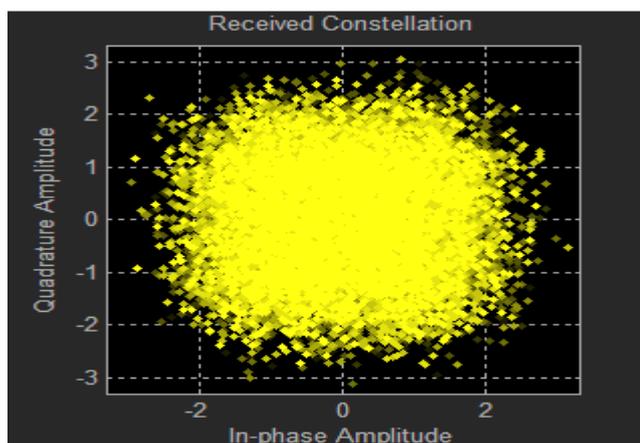


Figure IV.15 : Constellation de la QPSK à la réception.

## CHAPITRE IV: SIMULATIONS ET RESULTATS

---

Le résultat suivant est obtenu avec la probabilité de zéros=0.25.

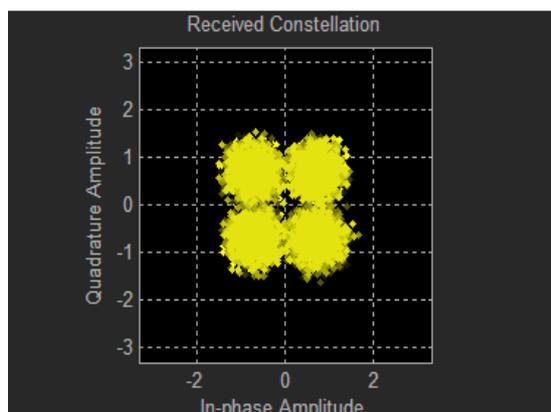


Figure IV.16 : Constellation de la QPSK à la réception avec  $p(0)=0.25$ .

Ce résultat montre l'effet de la probabilité sur la constellation.

L'effet du bruit blanc apparaît clairement sur la transmission.

La figure suivante donne une représentation de la constellation PSK à 8 états avant l'ajout du bruit produit par le canal avec la probabilité de zéros=0.25 et SNR=30 .

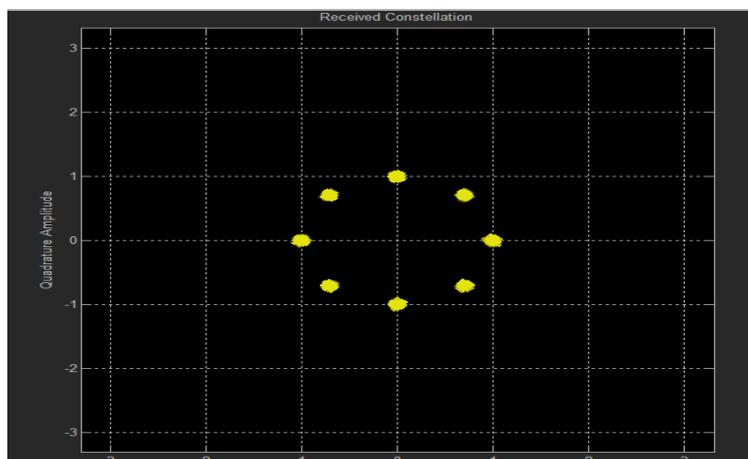


Figure IV.17 : Constellation de la 8- PSK à l'émission.

La figure suivante représente la constellation de la modulation numérique PSK à la réception avant la démodulation numérique avec la probabilité de zéros=0.25 et SNR=10.

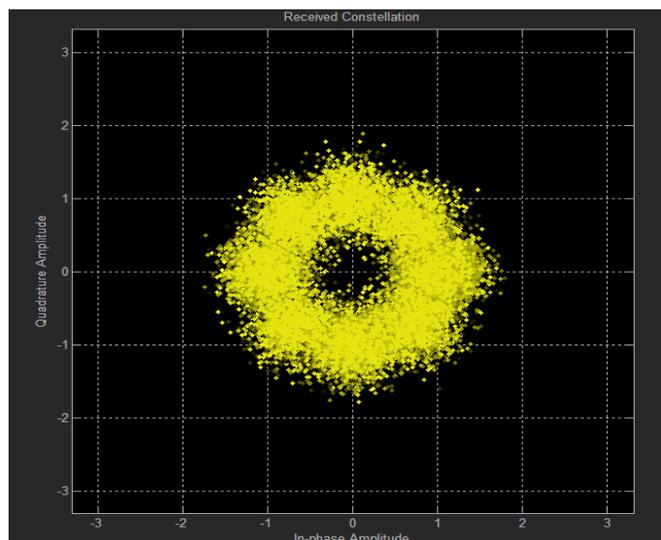


Figure IV.18 : Constellation de la 8- PSK à la réception.

### IV.4 Simulation de la chaine de transmission DVB-S2 sous Matlab

La figure suivante représente la constellation de la modulation numérique QPSK à l'émission.

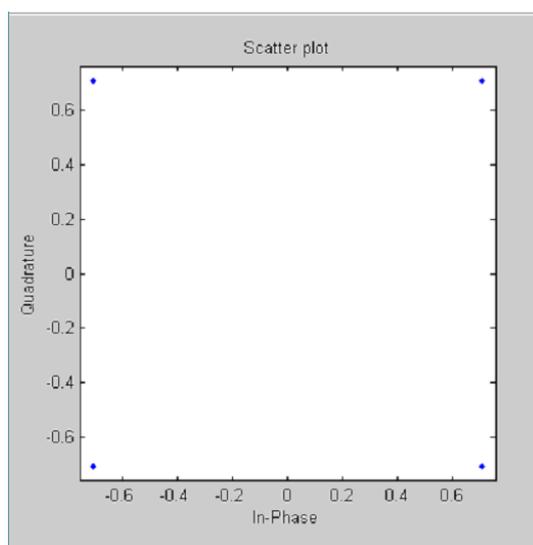


Figure IV.19 : Constellation de la QPSK à l'émission.

## CHAPITRE IV: SIMULATIONS ET RESULTATS

La figure suivante représente la constellation de la modulation numérique QPSK à la réception avec un SNR=  $[0.1 : 0.5 : 5]$  dB du canal AWGN.

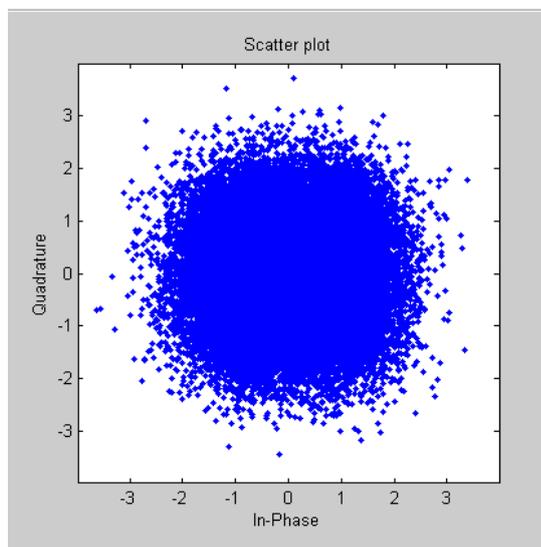


Figure IV. 20 : Constellation de la QPSK à la réception.

La figure suivante représente la BER.

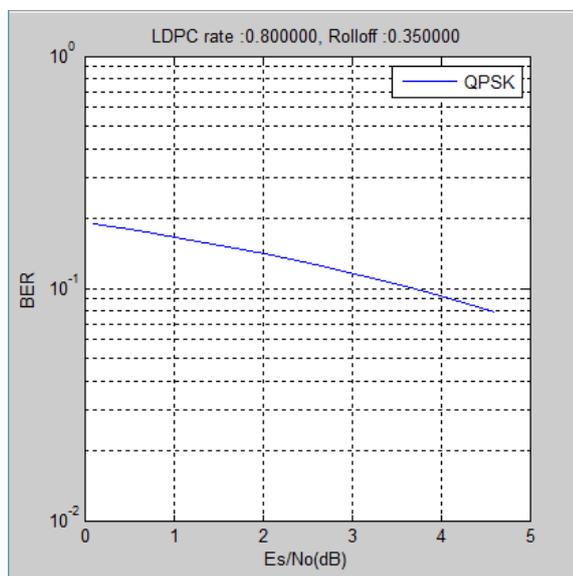


Figure IV.21: BER du signal DVB-S2.

### IV.5 Conception et simulation de la chaîne de transmission DVB-S2

#### IV.5.1 Conception et simulation de la chaîne d'émission DVB-S2

# CHAPITRE IV: SIMULATIONS ET RESULTATS

## IV.5.1.1 Conception et Simulation de la chaine d'émission DVB-S2 sous simulink

La figure suivante représente le schéma bloc de la chaine d'émission de DVB-s2 sous simulink.

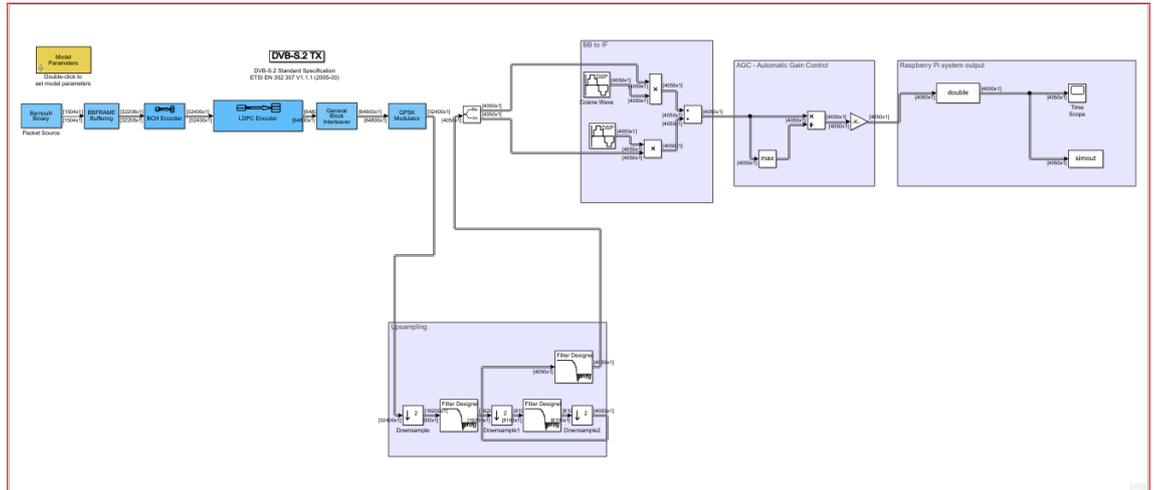


Figure IV.22 : Schéma bloc de la chaine d'émission DVB-s2

Nous allons décrire les blocs que nous avons ajoutés.

### IV.5.1.1 Sur-échantillonnage (Upsampling) :

Le signal en bande de base généré par le macro-bloc OFDM a une fréquence d'échantillonnage de  $F_s = \frac{48000}{20} = 2400 \frac{\text{samples}}{s}$ .

Le taux d'échantillonnage  $\frac{1}{\text{sample time}} = \frac{48000}{20} * \frac{48}{64} = 1800 \frac{\text{samples}}{s}$  choisis dans le bloc Générateur de Bernoulli a été, en fait, augmenté d'un facteur  $\frac{64}{48}$  par le bloc IFFT, qui produit une trame de 64 éléments pour chaque trame de 48 éléments reçus en entrée.

## CHAPITRE IV: SIMULATIONS ET RESULTATS

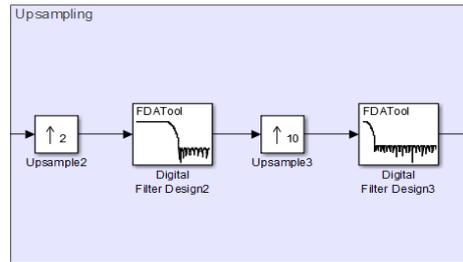


Figure IV.23 : Bloc Upsampling.

Pour moduler le signal OFDM en bande de base, la fréquence d'échantillonnage doit être augmentée, il convient donc, de l'augmenter d'un facteur de 20, en l'amenant à la valeur de  $48000 \frac{\text{samples}}{s}$  requis par le CAD de Raspberry Pi.

L'opération de sur échantillonnage est réalisée par la séquence de blocs inclus dans le macro-bloc Upsampling, qui effectue un sur échantillonnage par un facteur de 2 d'abord et ensuite par un facteur de 10. La procédure en deux étapes est pratique, comparée à un sur échantillonnage en une seule étape par un facteur de 20, parce qu'elle réduit la charge de calcul globale.

Le premier étage du macro-bloc Upsampling (Figure IV.23) introduit un 0 entre un échantillon et l'autre du signal d'entrée. Cette opération augmente la fréquence d'échantillonnage d'un facteur 2, la faisant passer de  $2400 \frac{\text{samples}}{s}$  à  $4800 \frac{\text{samples}}{s}$ , sans modifier le spectre du signal. Le filtre Digital « Filter Design 2 » suivant a donc pour tâche de supprimer une répétition périodique sur deux dans le spectre du signal, comme le montre la figure IV.24.

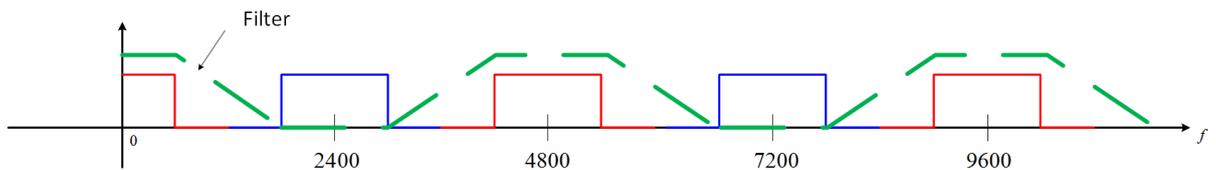


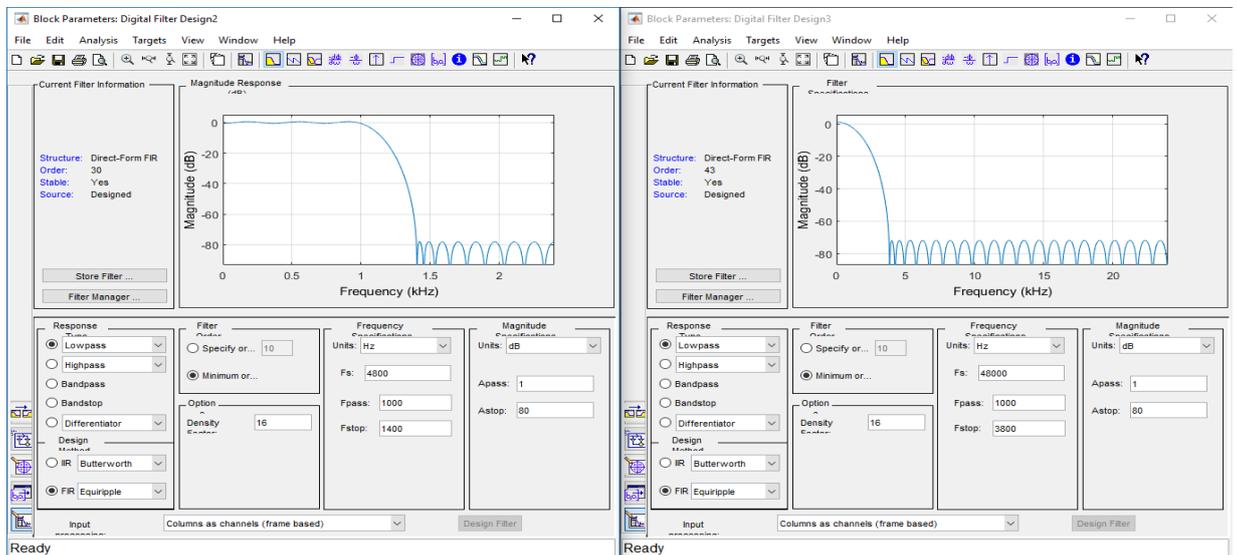
Figure IV.24 : Filtrage après sur échantillonnage par un facteur 2

Le spectre du signal à la sortie de filtre montre des répétitions périodiques avec une période égale à 4800 Hz, car il est approprié pour un signal avec une fréquence d'échantillonnage de  $4800 \frac{\text{samples}}{s}$ .

## CHAPITRE IV: SIMULATIONS ET RESULTATS

L'étape suivante, où le bloc effectue le sur échantillonnage par un facteur 10 suivi d'un filtre, fonctionne selon le même principe : l'étage de sur échantillonnage insère neuf 0s après chaque échantillon de signal d'entrée, augmentant la fréquence d'échantillonnage d'un facteur 10, et le filtre supprime 9 répétitions spectrales périodiques sur 10, de sorte que la première répétition périodique du spectre est centrée sur  $48000 \frac{\text{samples}}{\text{s}}$ .

Afin de concevoir correctement les filtres, régler  $F_s$  à la nouvelle fréquence d'échantillonnage, assigner  $F_{\text{pass}}$  la limite supérieure de la bande à préserver et assigner  $F_{\text{stop}}$  la valeur de l'ancienne fréquence d'échantillonnage moins la bande à préserver. La figure IV.25 montre les réglages des deux filtres.



(a) Filtre avec  $F_s=4800\text{Hz}$

(b) Filtre avec  $F_s=48000$

Figure IV.25 : réglage des filtres

### IV.5.1.2 Bande de base vers la fréquence intermédiaire (BB to IF)

Le macro-bloc BBtoIF module le signal, le traduisant de la bande de base à la fréquence intermédiaire. Cette opération est réalisée au moyen d'un modulateur en quadrature avec des porteuses générées en interne à une fréquence  $f_{\text{IF}} = 15 \text{ kHz}$ .

Le macro-bloc BBtoIF est un modulateur classique en quadrature, avec deux chemins séparés pour les composantes réelles (en phase) et imaginaires (en quadrature) du signal. Chaque composant est upconverti par un mélangeur (représenté par le bloc Produit) piloté par un cosinus ou un support sinusoïdal, généré par les blocs Cosine Wave et Sine Wave.

## CHAPITRE IV: SIMULATIONS ET RESULTATS

Les deux signaux modulés sont ensuite additionnés et injecter à l'étage suivant qui sera détaillé par ce qui suit (Figure IV.26).

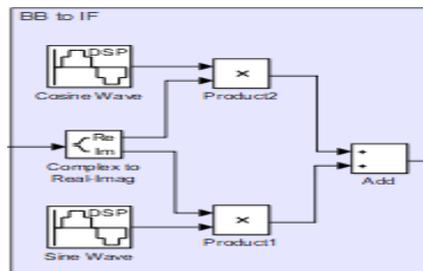
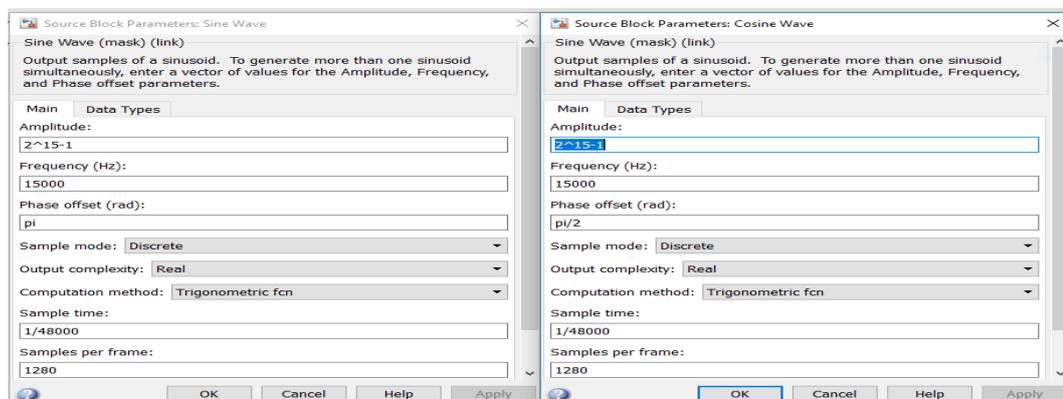


Figure IV.26 : Bloc BB to IF

Nous utilisons conventionnellement  $\cos(2\pi f_{IF} t)$  comme porteuse pour le signal en phase et  $-\sin(2\pi f_{IF} t)$  comme porteuse pour le signal en quadrature.

La figure IV.27 montre les réglages des blocs de production des porteuses. En particulier, le signal  $-\sin(2\pi f t)$  est généré en introduisant un décalage de phase  $\pi$  au signal  $\sin(2\pi f t)$  qui serait généré par défaut, tandis que le  $\cos(2\pi f t)$  est généré en introduisant un décalage de phase  $\pi/2$ .



(a) Réglage pour la génération  $-\sin(2\pi f t)$       (b) Réglage pour la génération  $\cos(2\pi f t)$

Figure IV.27 : Fenêtre de configuration pour le bloc cosine wave et le bloc sine wave

### IV.5.3 Contrôle Automatique de Gain (AGC) :

Le macro-bloc de contrôle automatique du gain adapte la plage dynamique du signal à l'intensité du signal. Les exigences du port de sortie du Raspberry Pi. La structure du macro-bloc, illustrée à la figure IV.28, est basée sur les blocs élémentaires Max, Diviser et Gain.

## CHAPITRE IV: SIMULATIONS ET RESULTATS

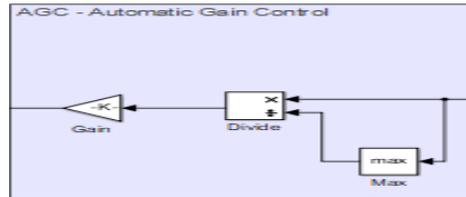
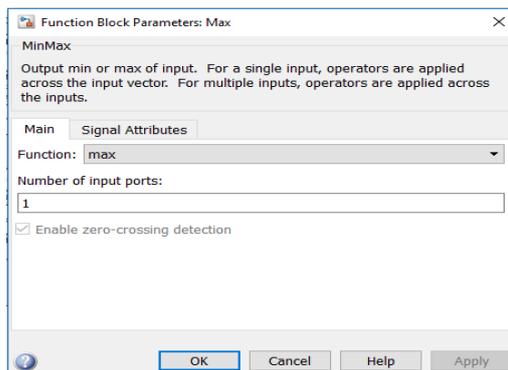


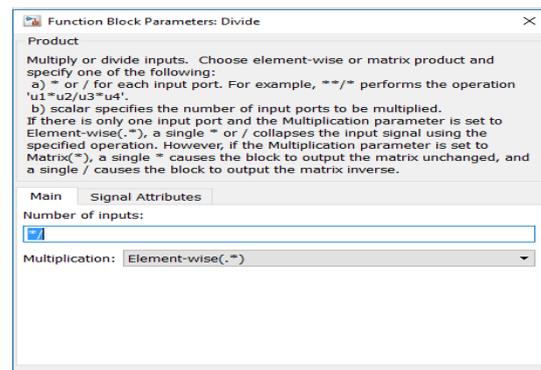
Figure IV.28 : Bloc AGC.

Pour chaque trame d'entrée, le macro-bloc de contrôle automatique de gain effectue, d'une normalisation, en divisant tous les échantillons de chaque trame par leur valeur maximale, suivie d'une multiplication par  $K = 2^{14} - 1$ . Les étapes de configuration sont montrées sur les figures: figure IV.29(a) et figure IV.29(b) et figure 30.

En conséquence de cette opération, la valeur la plus élevée dans chaque trame est égale à  $2^{14} - 1$ , correspondant à la valeur la plus élevée requise par le port de sortie du Raspberry Pi et égale à  $2^{15} - 1$ . Cette dernière valeur peut causer un dysfonctionnement dans certaine configuration. Ce problème est résolu en simple choix de la plage dynamique à  $2^{14} - 1$ .



(a) Max



(b) Division

Figure IV.29 : Blocs Max et Divide

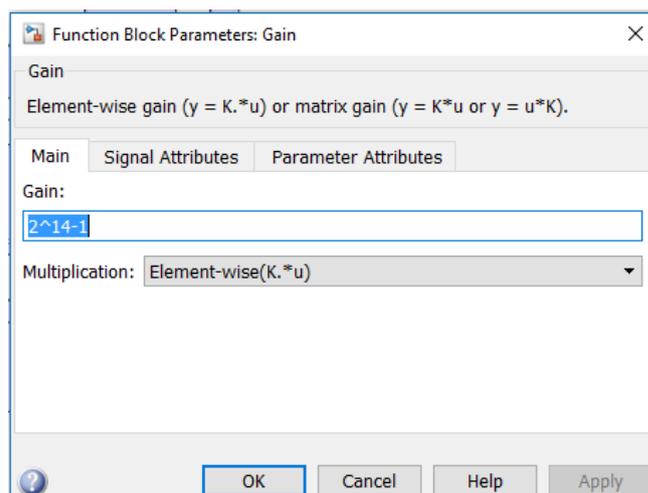


Figure IV.30 : BLOC GAIN

La figure suivante représenté la simulation de la chaine d'émission DVB-S2 sous simulink.

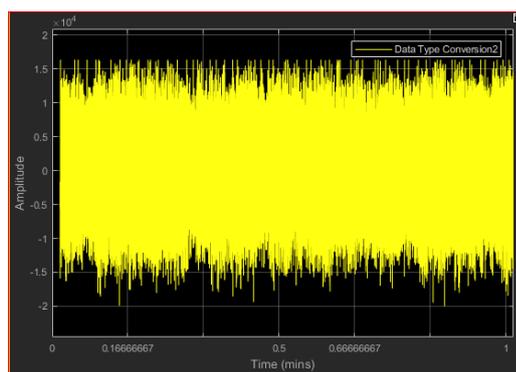


Figure IV.31 : Résultat de simulation sous simulink.

### IV.5.1.2 Implémentation de la chaine d'émission DVB-S2 sur la carte Raspberry Pi

Une fois que le fonctionnement de la chaine d'émission a été vérifié par simulation sous simulink en mode normal et external, nous pouvons l'implémentée sur la carte Raspberry Pi après avoir générer le code machine.

### IV.5.2 Conception et simulation de la chaine de réception DVB-S2

#### IV.5.2.1 Conception et simulation de la chaine de réception DVB-S2 sous simulink

## CHAPITRE IV: SIMULATIONS ET RESULTATS

La figure suivante représente le schéma bloc de la chaîne de réception DVB-S2 sous simulink. Elle est constituée des mêmes blocs que la chaîne d'émission.

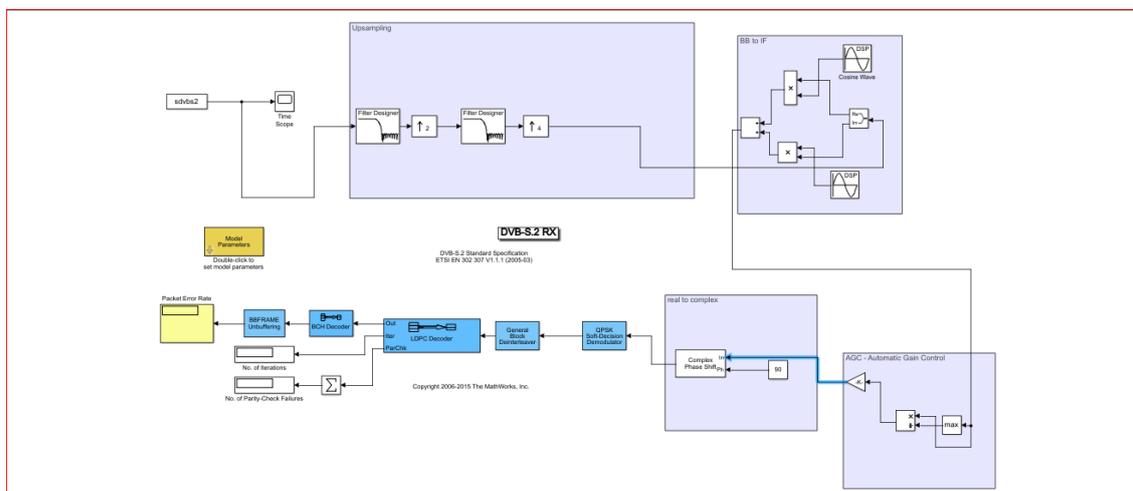


Figure IV. 32 : Schéma bloc de la chaîne de réception DVB-S2

### IV.5.1.3 Implémentation de la chaîne de réception DVB-S2 sur la carte Raspberry Pi

Une fois que le fonctionnement de la chaîne de réception a été vérifié par simulation sous simulink en mode normal et external, nous pouvons l'implémentée sur la carte Raspberry Pi après avoir générer le code machine.

## IV.6 Conclusion :

Dans ce chapitre, nous avons simulé la chaîne de transmission DVB-S2 sous simulink. Cette simulation est importante pour déterminer les performances de cette chaîne. Ensuite nous avons fait la conception de la chaîne d'émission seule sous simulink en tenant compte du bloc Raspberry. La simulation a été faite en mode normal et external. Une fois que le modèle de la chaîne d'émission est vérifié, nous pouvons l'implémenter sur la carte Raspberry Pi. Le même travail a été fait avec la chaîne de réception.

# Conclusion générale

## CONCLUSION GENERALE

Le travail que nous avons réalisé dans le cadre de ce mémoire nous a permis d'approfondir nos connaissances et aussi il nous a permis d'aborder un thème d'actualité dans le domaine de technologie de transmission sans fil.

Dans le premier chapitre, nous avons présenté la chaîne de transmission DVB ainsi que les standards existants de diffusion de la télévision numérique à savoir DVB-S, DVB-T, DVB-C et DVB-H.

Dans le deuxième chapitre nous avons présenté les notions importantes de la théorie des codes RS, LDPC, et le codage BCH. Nous avons dans un premier temps présenté les caractéristiques générales de ces codes, puis nous avons donné la façon de les construire.

Dans le troisième chapitre nous avons présenté un aperçu général de la carte Raspberry Pi, ainsi qu'un historique sur le développement de ces modèles de carte. Ensuite nous avons présenté les composants de base de la carte Raspberry Pi. Enfin nous avons présenté toute les étapes de l'installation du support package de simulink pour le Raspberry Pi et la méthode de vérification de la connectivité cette carte avec le PC.

Dans le quatrième chapitre, nous avons simulé la chaîne de transmission DVB-S2 sous simulink. Cette simulation est importante pour déterminer les performances de cette chaîne. Ensuite nous avons fait la conception de la chaîne d'émission seule sous simulink en tenant compte du bloc Raspberry. La simulation a été faite en mode normal et external. Une fois que le modèle de la chaîne d'émission est vérifié, nous pouvons l'implémenter sur la carte Raspberry Pi. Le même travail a été fait avec la chaîne de réception.

## REFERENCES ET DOCUMENTATION

- [1] : Yasser Bouslah –Salim, Gantassi –KaisSlimeni, étude et modélisation sous-Simulink d'une chaîne de transmission DVB-S. le 31 janvier 2015.
- [2] : Etude comparative entre DVB-S et DVB-S2.
- [3] : Lilian Bossuet Guillaume Ferre, étude et modélisation sous-Simulink. Le 29 août 2008.
- [4] : [Http//.wikipeda.org/wiki.com](http://.wikipeda.org/wiki.com).
- [5] : Hajjami Jaouad , étude technique de la chaîne DVB-T. Le 14 Juin 2013
- [6] : TejedorAlonso\_Maros, Ari Rantala Digital TV: fixed and mobile standards.
- [7] :G. A. Franco, G. Lachs, an orthogonal coding technique for communication ; IRE Inten. Conv. Rec ; Vol. 9 ; pp. 126-13,1961.
- [8] : Mr. Jyrki T, J. Penttinen, The DVB-H Handbook. Le may 2005.
- [9] : Oliver Pretzel. Clarendon Press, Oxford, Error-Correcting Codes and Finite Fields.
- [10] : Jean-Baptiste Doré, Optimisation conjointe de codes LDPC et de leurs architectures de décodage et mise en œuvre sur FPGA. Le 12 May 2008
- [11] : Nelly Burrin. Master en systèmes de communication Projet de semestre, été 2006.
- [12] : Raspberry Pi Programming using Simulink.

## RESUME

La télévision numérique fait partie de notre quotidien. Actuellement, il existe plusieurs possibilités pour recevoir cette télévision. Ces systèmes ont été développés par le groupe DVB.

Il existe plusieurs normes de la DVB, les plus importantes sont les normes DVB-T (terrestre) et DVB-C (câble), DVB-S (par satellite). Malgré cela, les besoins des utilisateurs évoluent constamment dans le domaine de l'éducation, de la formation, de communication et de divertissement. C'est pour cette raison qu'il existe de nouveaux défis que les normes ne sont pas en mesure de satisfaire. De plus, elle exige une révision constante des normes DVB.

Dans ce projet de master nous nous sommes intéressés à la conception et à la simulation d'une chaîne de transmission DVB. Nous avons simulé la chaîne de transmission DVB-S2 sous simulink. Cette simulation est importante pour déterminer les performances de cette chaîne. Ensuite nous avons fait la conception de la chaîne d'émission seule sous simulink en tenant compte du bloc Raspberry. La simulation a été faite en mode normal et external. Une fois que le modèle de la chaîne d'émission est vérifié, nous pouvons l'implémenter sur la carte Raspberry Pi. Le même travail a été fait avec la chaîne de réception.

**Mot clé :** Télévision numérique, DVB, normes DVB, Raspberry Pi, émetteur, récepteur.

## Summary

Digital television is part of our daily lives. Currently, there are several possibilities to receive this television. These systems were developed by the DVB Group.

There are several DVB standards, the most important being DVB-T (terrestrial) and DVB-C (cable), DVB-S (satellite). Despite this, users' needs are constantly evolving in the fields of education, training, communication and entertainment. For this reason, there are new challenges that standards are not able to meet. Furthermore, it requires a constant revision of DVB standards.

In this master project we were interested in the design and simulation of a DVB transmission chain. We simulated the DVB-S2 transmission chain under simulink. This simulation is important to determine the performance of this chain. Then we did the design of the emission chain alone under simulink taking into account the Raspberry block. The simulation was performed in normal and external mode. Once the transmission chain model is verified, we can implement it on the Raspberry Pi card. The same work was done with the receiving chain.

**Keyword:** Digital TV, DVB, DVB standards, Raspberry Pi, transmitter, receiver.