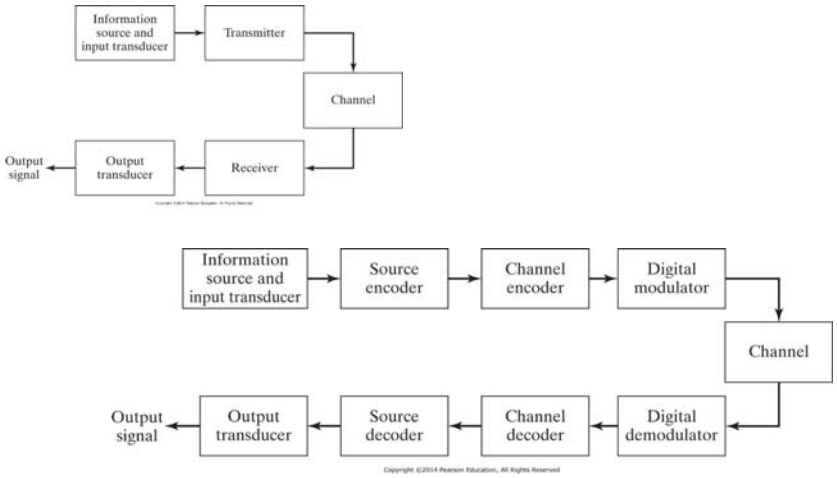
Chapitre 1.
Conversion analogique/numérique

Contenu

- Échantillonnage
- Quantification
- Encodage
- Codage en signaux
  - Modulation impulsionnelle codée
  - Modulation delta
- Techniques d'analyse-synthèse
- Transmission numérique de l'audio

De l'analogique au numérique

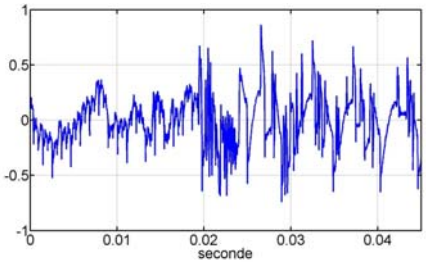


Procédure de conversion A-N

- 3 étapes:
  - Échantillonnage
  - Quantification
  - Encodage
- Avantages obtenus par rapport à l'analogique:
  - Meilleure tolérance au bruit
  - Facilité d'implémentation sur circuits intégrés
  - Possibilité de cryptographie/anti-brouillage

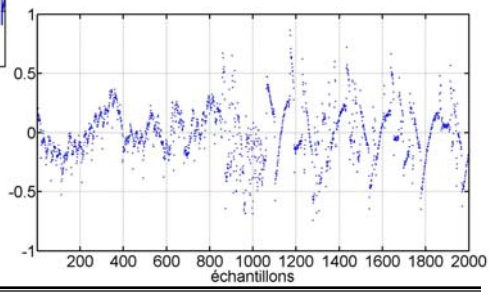
Échantillonnage

- Saisit périodiquement l'amplitude d'un signal analogique pour le remplacer par la suite de points obtenue:



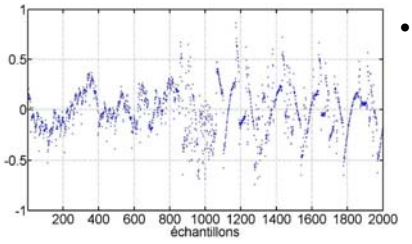
Échantillon	Valeur
....	....
1079	0.3029
1080	0.2901
1081	0.2773
1082	0.2654
....	....

- L'axe des temps est remplacé par des indices pour une représentation implicite $(t = nT_s)$



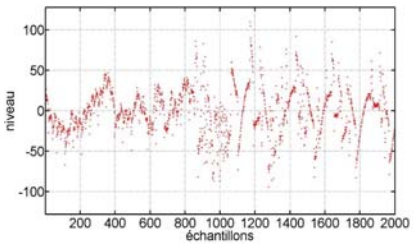
Quantification

- Divise l'axe vertical en 2^n paliers pour représenter l'amplitude d'un signal analogique



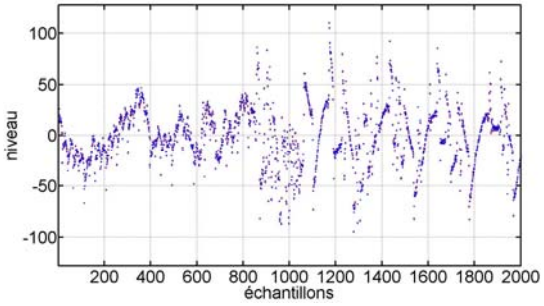
Échantillon	Valeur
....	....
1079	00100110
1080	00100101
1081	00100011
1082	00100001
....	....

- *n* est le nombre de bits des valeurs binaires équivalentes
  - ex.: $2^8 = 256$ niveaux espacés uniformément



Quantification

- Plus le nombre de pas augmente et plus le signal original ressemble au signal original



Théorème d'échantillonnage

- Définit la fréquence d'échantillonnage minimum pour pouvoir reconstituer le signal d'origine

  Pour un signal analogique à bande passante finie, f_s doit ***être au moins le double*** *de la plus haute fréquence du signal*

  (critère de Nyquist ~1928; théorème de Shannon~1948)

Signal analogique d'origine

$$x_\delta(t) = x(t) \sum_{n=-\infty}^{\infty} \delta(t - n\,T_s)$$

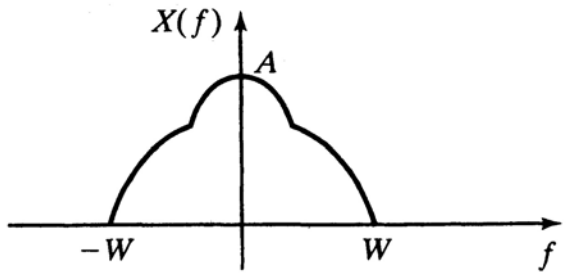
Signal échantillonné

Période d'échantillonnage $(1/f_s)$

Analyse en fréquence

- Signal analogique:

$$x(t) \overset{\mathcal{F}}{\leftrightarrow} X(f)$$



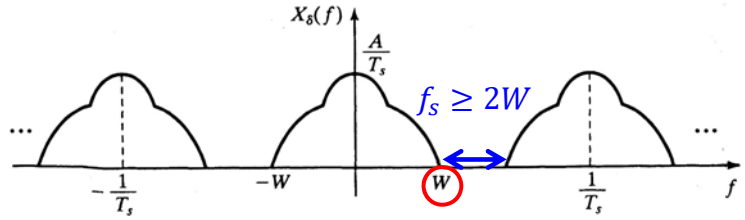
Analyse en fréquence

- Signal échantillonné:

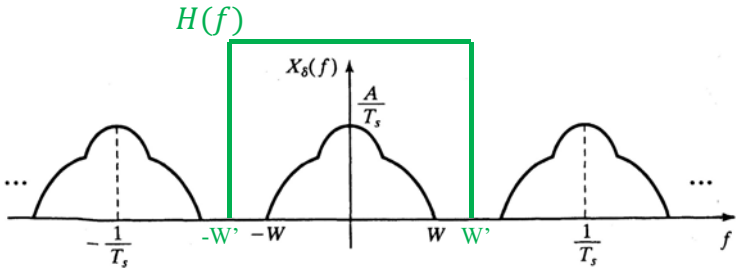
$$x_\delta(t) \overset{\mathcal{F}}{\leftrightarrow} X_\delta(f) = X(f) * \mathcal{F}\left[\sum_{n=-\infty}^{\infty} \delta(t - nT_s)\right]$$

$$= X(f) * \frac{1}{T_s}\sum_{n=-\infty}^{\infty} \delta\left(f - \frac{n}{T_s}\right) = \frac{1}{T_s}\sum_{n=-\infty}^{\infty} X\left(f - \frac{n}{T_s}\right)$$

Le spectre du signal d'origine devient périodique après échantillonnage!



Reconstruction



$$X(f) = X_\delta(f) \cdot H(f) = X_\delta(f) \cdot T_s \prod\left(\frac{f}{2W'}\right)$$

$$x(t) = x_\delta(t) * 2W'T_s\, sinc(2W't)$$

$$= \sum_{n=-\infty}^{\infty} 2W'T_s x(nT_s) sinc(2W'(t - nT_s))$$

Erreur de quantification

- Durant la procédure de quantification, on associe chaque valeur d'un ensemble infini (signal analogique) à une valeur d'un ensemble fini.
  - Perte possible d'information due à l'erreur de quantification
- On peut mesurer la distorsion introduite par un calcul d'erreur

Fonction de quantification

$$\hat{x} = Q(x)$$

$$d(x, \hat{x}) = (x - Q(x))^2 = \tilde{x}^2$$

Erreur de quantification

- La valeur *x* du signal au moment de l'échantillonnage est inconnue a priori et on peut la considérer comme une variable aléatoire *X*.
  - Une analyse statistique de l'erreur de distorsion devient possible
- Erreur quadratique moyenne:

$$D = E\left[d\left(X, \hat{X}\right)\right] = E(X - Q(X))^2$$

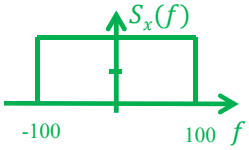
- On évalue son importance en calculant le rapport signal-sur-bruit-de-quantification

$$SQNR = \frac{E(X^2)}{E\left(X - Q(X)\right)^2}$$

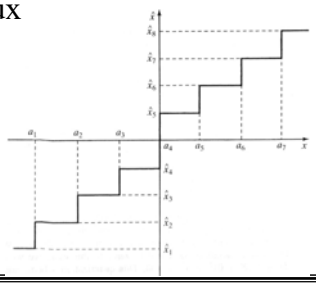
Exemple 1

- Source gaussienne avec une moyenne de zéro est une densité spectrale de puissance:

$$S_x(f) = \begin{cases} 2 & |f| < 100 \text{ Hz} \\ 0 & autrement \end{cases}$$



- Fréquence d'échantillonnage au taux de Nyquist, quantification à 8 niveaux
- Quels sont le taux de transfert, le niveau de distorsion et le SQNR?



Indice	a	$\hat{x}$
1	-60	-70
2	-40	-50
3	-20	-30
4	0	-10
5	20	10
6	40	30
7	60	50
8	n/a	70

Exemple 1

- Fréquence d'échantillonnage: 2x100=200 Hz
- Quantification avec 8 niveaux ⇒
  - $\log_2 8 = 3$ bits/échantillons
  - Taux de transmission de 3x200 = 600 bits/seconde
- Distorsion:

$$D = E\left(X - \hat{X}\right)^2 = \int_{-\infty}^{\infty} \left(x - Q(x)\right)^2 f_x(x)dx$$

- Densité de probabilité: $f_x(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{x^2}{2\sigma^2}}$

Exemple 1

$$\sigma^2 = E(X^2) = \int_{-\infty}^{\infty} S_x(f)df = \int_{-100}^{100} 2df = 400$$

$$D = \int_{-\infty}^{a_1} (x - \hat{x}_1)^2 f_x(x)dx + \sum_{i=2}^{7} \int_{a_i - 1}^{a_i} (x - \hat{x}_i)^2 f_x(x)dx + \int_{a_7}^{\infty} (x - \hat{x}_8)^2 f_x(x)dx$$

Demande l'usage de tables d'intégrales pour la solution

où $f_x(x) = \frac{1}{\sqrt{800\pi}} e^{-\frac{x^2}{800}}$

Exemple 1

Avec $\varphi(x) = \frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2}}$ et $\Phi(x) = 0.5\left(1 + erf\left(\frac{x}{\sqrt{2}}\right)\right)$:

1) $\int \varphi(x)dx = \Phi(x) + C$

2) $\int x\, \varphi(x)dx = -\varphi(x) + C$

3) $\int x^2 \varphi(x)dx = \Phi(x) - x\varphi(x) + C$

Owen, D. *Communications in Statistics: Simulation and Computation* **B9**. pp. 389 – 419 (1980)

Exemple 1

$$D = \sum_{i=1}^{8} 0.5(1 + {x_i}^2)\left(1 + erf\left(\frac{x}{\sqrt{2\sigma^2}}\right)\right) - \sigma^2(x + 2x_1)f_x(x)\Big]_{c_i}^{b_i}$$

où $b = [a, \infty]$ et $c = [-\infty, a]$

Finalement, $D = 33.38$

Exemple 1

- Le ratio signal sur bruit de quantification:

$$SQNR = \frac{E(X^2)}{E(X - Q(X))^2} = \frac{P_X}{P_{\tilde{X}}} = \frac{400}{33.38} = 11.98$$

ou 10.78 dB.

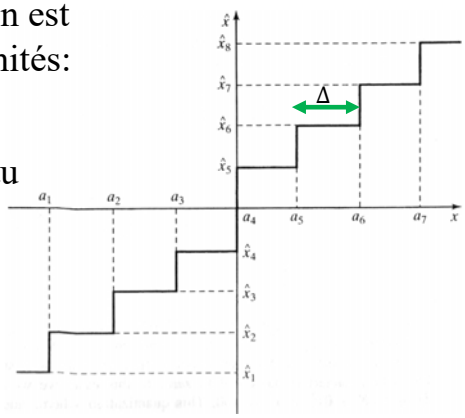
Quantification uniforme

- Dans l'exemple, la distance entre 2 niveaux de quantification est constante, sauf aux extrémités:

$$a_i - a_{i-1} = \Delta$$

- Si les niveaux sont fixés au milieu des frontières:

$$erreur\ max = \frac{\Delta}{2}$$



Quantification uniforme

- On peut minimiser l'erreur en optimisant:

$$D = \int_{-\infty}^{a_1} \left(x - \left(a_1 - \Delta/2\right)\right)^2 f_x(x)dx + \sum_{i=1}^{N-2} \int_{a_1+(i-1)\Delta}^{a_1+i\Delta} \left(x - \left(a_1 + i\Delta - \Delta/2\right)\right)^2 f_x(x)dx + \int_{a_1+(N-2)\Delta}^{\infty} \left(x - \left(a_1 + (N-2)\Delta - \Delta/2\right)\right)^2 f_x(x)dx$$

- D dépend d'intégrales de type $\int x^2 f_x(x)$ et demande souvent l'usage de méthodes numériques pour trouver Δ optimal

Quantificateur uniforme optimal

- Pour un signal Gaussien avec $\mu = \sigma = 0$. Il y a toujours une frontière à $x = 0$.

Nombre Niveau	Δ	Erreur moyenne carrée (D)
2	1.596	0.3634
4	0.9957	0.1188
8	0.5860	0.03744
16	0.3352	0.01154
32	0.1881	0.003490

Quantificateur non uniforme

- Diminue l'erreur en utilisant des intervalles de quantification inégaux
- Avec N niveaux de quantification, on a

$$D = \int_{-\infty}^{a_1} (x - \hat{x}_1)^2 f_x(x)dx + \sum_{i=1}^{N-2} \int_{a_i}^{a_{i+1}} (x - \hat{x}_{i+1})^2 f_x(x)dx + \int_{a_{N-1}}^{\infty} (x - \hat{x}_N)^2 f_x(x)dx$$

$\Rightarrow$ D dépend de $2N$-1 variables (N-1 a_i et N $\hat{x}_i$)

– En posant $\frac{\partial D}{\partial a_i}$ =0, on obtient $a_i = \frac{1}{2}(\hat{x}_i + \hat{x_{i+1}})$ (point milieu)

– En posant $\frac{\partial D}{\partial \hat{x}_i}$ =0, on on obtient $\hat{x}_i = \frac{\int_{a_{i-1}}^{a_i} x f_x(x)dx}{\int_{a_{i-1}}^{a_i} f_x(x)dx}$ (barycentre)

– Conditions de Lloyd-Max; équations aussi difficiles à résoudre

TABLE 7.2 OPTIMAL NONUNIFORM QUANTIZER FOR A GAUSSIAN SOURCE

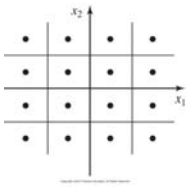
N	$\pm a_i$	$\pm \hat{x}_i$	D	$H(\hat{X})$
1	—	0	1	0
2	0	0.7980	0.3634	1
3	0.6120	0, 1.224	0.1902	1.536
4	0, 0.9816	0.4528, 1.510	0.1175	1.911
5	0.3823, 1.244	0, 0.7646, 1.724	0.07994	2.203
6	0, 0.6589, 1.447	0.3177, 1.000, 1.894	0.05798	2.443
7	0.2803, 0.8744, 1.611	0, 0.5606, 1.188, 2.033	0.04400	2.647
8	0, 0.5006, 1.050, 1.748	0.2451, 0.7560, 1.344, 2.152	0.03454	2.825
9	0.2218, 0.6812, 1.198, 1.866	0, 0.4436, 0.9188, 1.476, 2.255	0.02785	2.983
10	0, 0.4047, 0.8339, 1.325, 1.968	0.1996, 0.6099, 1.058, 1.591, 2.345	0.02293	3.125
11	0.1837, 0.5599, 0.9656, 1.436, 2.059	0, 0.3675, 0.7524, 1.179, 1.693, 2.426	0.01922	3.253
12	0, 0.3401, 0.6943, 1.081, 1.534, 2.141	0.1684, 0.5119, 0.8768, 1.286, 1.783, 2.499	0.01634	3.372
13	0.1569, 0.4760, 0.8126, 1.184, 1.623, 2.215	0, 0.3138, 0.6383, 0.9870, 1.381, 1.865, 2.565	0.01406	3.481
14	0, 0.2935, 0.5959, 0.9181, 1.277, 1.703, 2.282	0.1457, 0.4413, 0.7505, 1.086, 1.468, 1.939, 2.625	0.01223	3.582
15	0.1369, 0.4143, 0.7030, 1.013, 1.361, 1.776, 2.344	0, 0.2739, 0.5548, 0.8512, 1.175, 1.546, 2.007, 2.681	0.01073	3.677

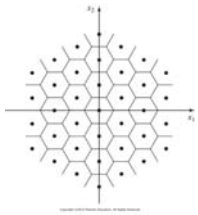
Copyright ©2014 Pearson Education, All Rights Reserved

Quantification vectorielle

- En quantification scalaire, chaque valeur quantifiée de n bits provient de la même source
- On peut aussi quantifier N sources à la fois et traiter le résultat comme point dans un espace euclidien de dimension nN.
  - On obtient $M = 2^{nN}$ points (symboles) répartis dans les centres de régions contiguës; le débit de chaque source demeure inchangé ($R = nf_s$ bits/s)
  - On peut optimiser la distorsion d'échantillonnage en jouant sur la forme des régions.

16 régions, chacune de 4 bits





k régions, chacune de de $\log_2 k$ bits

Quantification vectorielle

- En supposant K régions de quantification et une dimension d'espace n, le taux du code source est $R = \frac{\log K}{n}$ bits/valeur.
- La conception du quantificateur vectoriel optimal suit les critères suivants:
  - Chaque région de quantification R_i englobe l'ensemble des points les plus proches de la valeur de quantification $\widehat{x_i}$
  - $\widehat{x_i}$ est au barycentre de la région R_i
- La quantification vectorielle est fréquemment utilisée pour le codage de la parole et des images
  - Plus efficace que la quantification scalaire, car plus flexible

Codage

- Assigne une valeur binaire distincte à chaque valeur quantifiée
- L'assignation peut se faire de différentes façons
- Peut être utilisé avec l'une ou l'autre forme de quantification

TABLE 7.3 NBC AND GRAY CODES FOR A 16-LEVEL QUANTIZATION

Quantization Level	Level Order	NBC Code	Gray Code
$\hat{x}_1$	0	0000	0000
$\hat{x}_2$	1	0001	0010
$\hat{x}_3$	2	0010	0011
$\hat{x}_4$	3	0011	0001
$\hat{x}_5$	4	0100	0101
$\hat{x}_6$	5	0101	0100
$\hat{x}_7$	6	0110	0110
$\hat{x}_8$	7	0111	0111
$\hat{x}_9$	8	1000	1111
$\hat{x}_{10}$	9	1001	1110
$\hat{x}_{11}$	10	1010	1100
$\hat{x}_{12}$	11	1011	1101
$\hat{x}_{13}$	12	1100	1001
$\hat{x}_{14}$	13	1101	1000
$\hat{x}_{15}$	14	1110	1010
$\hat{x}_{16}$	15	1111	1011

Copyright ©2014 Pearson Education, All Rights Reserved

Codage de signaux

- L'objectif est de reproduire le signal d'une source, une fois rendu à la destination, avec le moins de distorsion possible.
- On veut aussi utiliser le moins de bits possibles
- Plusieurs approches
  - La modulation impulsionnelle codée (*Pulse Code Modulation* ou PCM)
  - La modulation delta (DM)

Modulation impulsionnelle codée

- Simple
- On suppose que
  - La bande du signal est limitée avec une fréquence maximale de W. Sinon, on introduit un filtre.
  - L'amplitude du signal est finie, $|x(t)| \leq x_{\max}$
  - Le nombre de niveaux de quantifications $N = 2^n$ est large
- PCM uniforme
- PCM non uniforme

Copyright ©2014 Pearson Education, All Rights Reserved

Erreur d'un PCM uniforme

- On peut estimer le SQNR d'un PCM uniforme si:
  - Le quantificateur est symétrique
  - L'amplitude maximale du signal analogique (x_{max}) ne cause pas de débordement
  - $\Delta = \frac{2x_{max}}{N}$ est petit.
- On assume que l'erreur à une distribution uniforme:

$$f_X(\hat{x}) = \begin{cases} \frac{1}{\Delta}, & -\frac{\Delta}{2} < \hat{x} \leq \frac{\Delta}{2} \\ 0, & autrement \end{cases}$$

Erreur d'un quantificateur uniforme

$$E(\hat{X}^2) = \int_{-\Delta/2}^{\Delta/2} x^2 f_X(\hat{x}) d\hat{x} = \frac{1}{\Delta} \int_{-\Delta/2}^{\Delta/2} \hat{x}^2 \, d\hat{x} = \frac{\Delta^2}{12} = \frac{x^2{}_{max}}{3 \times 4^v}$$

- Rapport signal-bruit de quantification:

$$SQNR = \frac{P_X}{E(\hat{X}^2)} = \frac{3P_X}{(x_{max})^2} \cdot 4^v \text{ (en dB ?)}$$

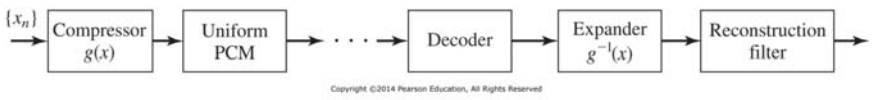
- Pour un **signal sinusoïdal**:

$$P_X = \frac{(x_{max})^2}{2} \rightarrow SQNR = \frac{3}{2} \cdot 4^v$$

- en dB: $SQNR_{dB} = 1.8 + 6R$
- et pour un signal uniformément distribué entre -1 et 1 ?

PCM non uniforme

- Le PCM fonctionne bien si l'amplitude du signal d'entrée est proche de la distribution uniforme.
- Pour des signaux comme la voix, les petites amplitudes sont plus probables et un PCM non uniforme améliorerait les performances
  - On peut revenir à un PCM uniforme en utilisant un élément non linéaire (compression) à l'entrée
  - La fonction inverse (expansion) est réalisée à la sortie.



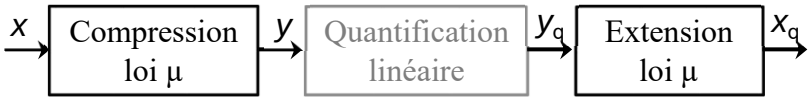
Copyright ©2014 Pearson Education, All Rights Reserved

Compresser comment ?

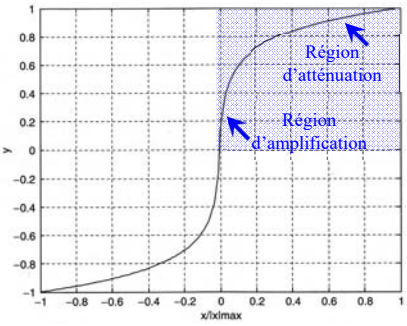
- On peut :
  - Réduire la gamme dynamique ou le nombre de valeurs du signal d'entrée par une transformation appropriée
  - Exploiter la redondance de données (ex. les flux audio et vidéo, incluant les passages silencieux, le texte, etc.)
  - Exploiter les particularités de la perception humaine
  - Exploiter la corrélation entre données voisines dans un objet spatial ou temporel

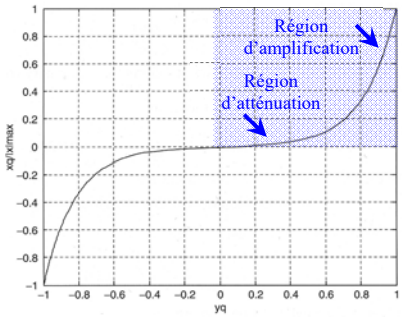
Compression-extension par loi μ

- Appelé aussi log-PCM (standard G.711)
- Maintient le SNR constant pour *N* donné, alors que le pas de quantification est constant et l'amplitude du signal variable
  - Amplifie les faibles amplitudes et attenue les grandes afin de maintenir un SNR acceptable
- Usage standard en téléphonie



Compression-extension par loi μ





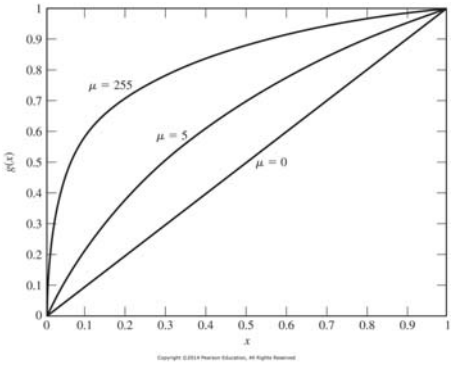
$$y = sign(x)\frac{\ln\left(1+\mu\frac{|x|}{|x_{max}|}\right)}{\ln(1+\mu)}$$

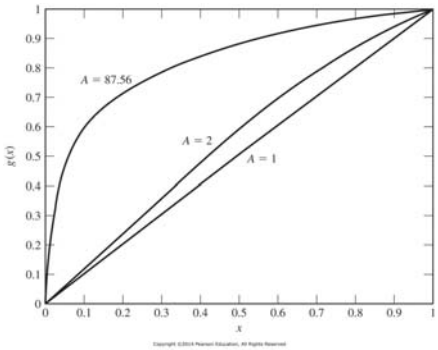
$$x_q = |x_{max}|sign(y_q)\frac{(1+\mu)^{|y_q|}-1}{\mu}$$

μ = 255 en téléphonie
pour SNR=48 dB

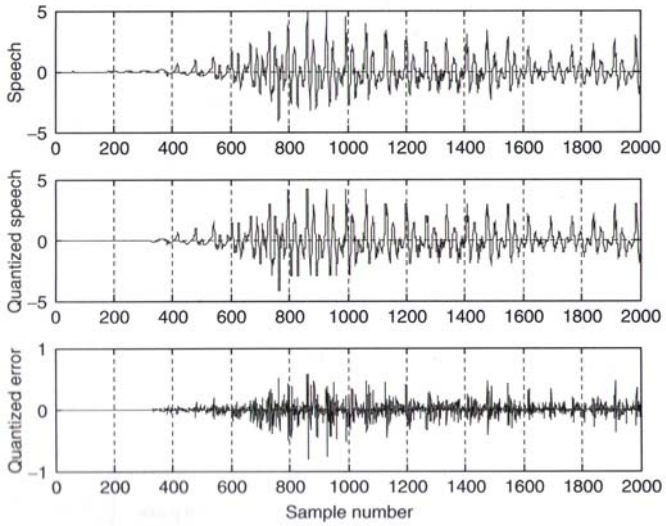
Les lois de compression-expansion

- La loi-μ est définie par la fonction $g(x) = \frac{\log(1+\mu|x|)}{\log(1+\mu)}$
- La loi-A est définie par la fonction $g(x) = \frac{\log(1+A|x|)}{\log(1+A)}\text{sgn}(x)$



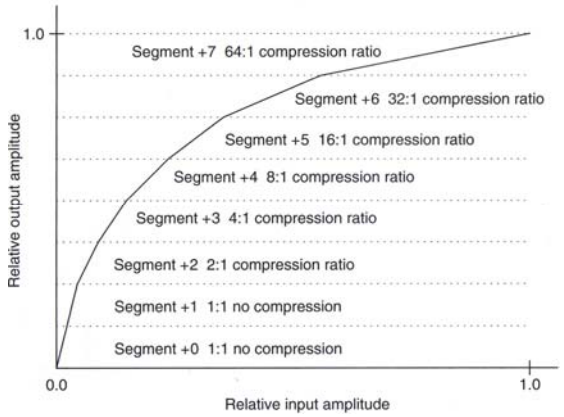


Exemple loi μ analogue



Loi μ binaire

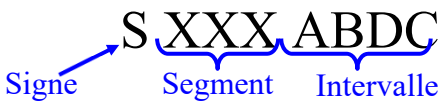
- Réduit la taille d'un signal modulé par codage d'impulsion (MCI ou PCM), typiquement de 12 à 8 bits



La pente de chaque segment est ½ celle du précédent

Compression par loi μ binaire

- Format du code compressé :



Segment	Code linéaire 12 bits*	Amplitude en décimale	Code compressé 8 bits
0	S0000000ABCD	0 à 15	S000ABCD
1	S0000001ABCD	16 à 31	S001ABCD
2	S000001ABCDX	32 à 63	S010ABCD
3	S00001ABCDXX	64 à 127	S011ABCD
4	S0001ABCDXXX	128 à 255	S100ABCD
5	S001ABCDXXXX	256 à 511	S101ABCD
6	S01ABCDXXXXX	512 à 1023	S110ABCD
7	S1ABCDXXXXXX	1024 à 2047	S111ABCD

* X: Valeurs homogénéisées par la division progressive de la pente par 2

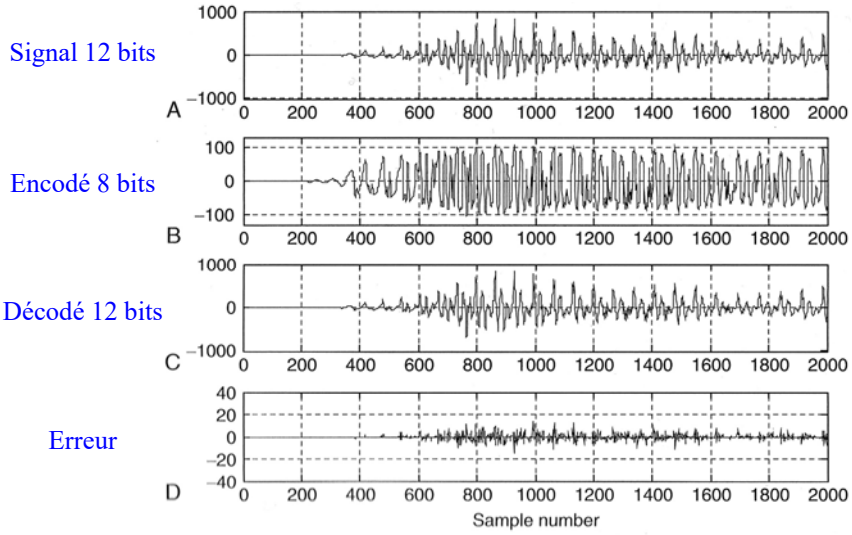
Extension par loi μ binaire

- Pour μ = 255:

Code compressé 8 bits	Amplitude en décimale	Segment	Code linéaire 12 bits
S000ABCD	0 à 15	0	S0000000ABCD
S001ABCD	16 à 31	1	S0000001ABCD
S010ABCD	32 à 47	2	S000001ABCD1
S011ABCD	48 à 63	3	S00001ABCD10
S100ABCD	64 à 79	4	S0001ABCD100
S101ABCD	80 à 95	5	S001ABCD1000
S110ABCD	96 à 111	6	S01ABCD10000
S111ABCD	112 à 127	7	S1ABCD100000

- Ex: 0000 1110 1010 → 0 1001101→ 0000 1110 1100
- Codage avec pertes!

Exemple loi μ binaire

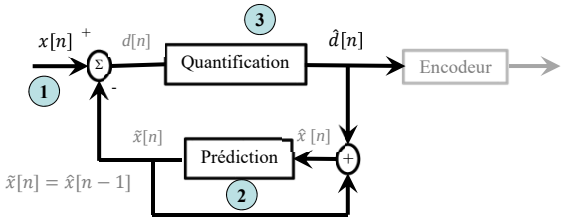


MIC différentiel

- En MIC, la quantification d'un échantillon se fait indépendamment de celle des autres échantillons
- On peut exploiter la corrélation entre échantillons adjacents pour un résultat avec moins de bits
- Le MICD (DPCM) de base quantifie la différence entre deux échantillons adjacents (⇒ gamme dynamique moindre ⇒ moins de bits de quantification)

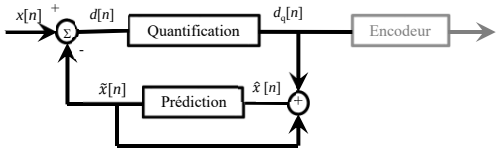
MIC différentiel (DPCM en anglais)

- Encode la différence quantifiée entre la valeur actuelle du signal et celle prédite
  - Dans la forme la plus simple, la valeur prédite est la valeur précédente; sinon on combine plusieurs échantillons précédents



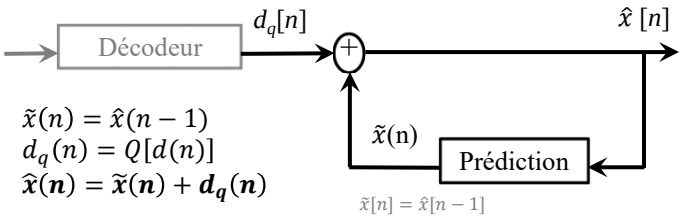
$$d(n) = x(n) - \tilde{x}(n)$$
$$\hat{x}(n) = \tilde{x}(n) + \hat{d}(n)$$
$$\hat{d}(n) = Q[d(n)]$$

Démodulation MICD



$$\tilde{x}(n) = \hat{x}(n-1)$$
$$d(n) = x(n) - \tilde{x}(n)$$
$$\hat{x}(n) = \tilde{x}(n) + d_q(n)$$
$$\boldsymbol{d_q(n) = Q[d(n)]}$$

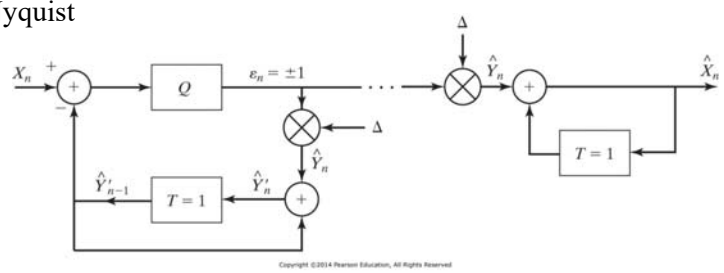
- Démodulation



$$\tilde{x}(n) = \hat{x}(n-1)$$
$$d_q(n) = Q[d(n)]$$
$$\boldsymbol{\hat{x}(n) = \tilde{x}(n) + d_q(n)}$$

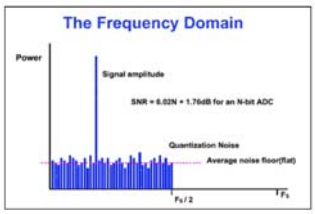
La modulation delta

- Version simplifiée de MICD avec 1 bit par échantillon
- Avec seulement deux niveaux, le bruit de quantification élevé à moins de réduire la gamme dynamique de Y_n
  - Réduire la gamme dynamique de Y_n rend X_n et X_{n-1} plus corrélés; cela est possible si on utilise un taux d'échantillonnage >> taux de Nyquist

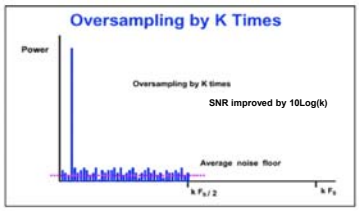


Effet de suréchantillonner

Bruit de quantification pour $f_S=2f_{max}$

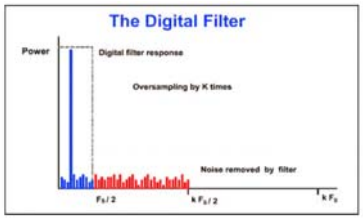


Bruit de quantification pour $f_S=2kf_{max}$



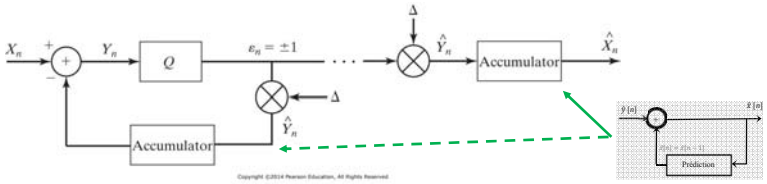
- Le suréchantillonnage étale le spectre du bruit tout en conservant son énergie
  - Le niveau moyen du bruit baisse
- $K=4^m$ améliore le SQNR de $6.02m$ dB après filtrage (revient à ajouter m bits de résolution au signal)

Spectre après filtrage

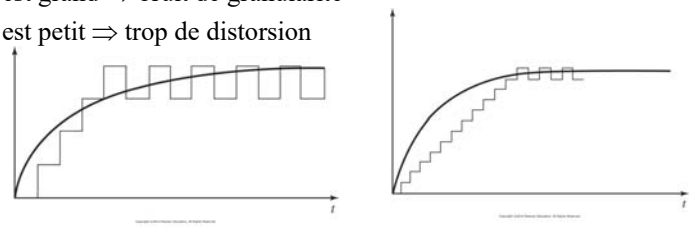


La modulation delta

- La MD est possible avec des accumulateurs.

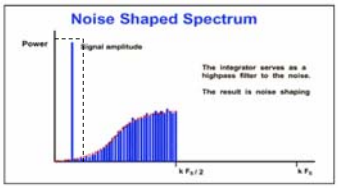


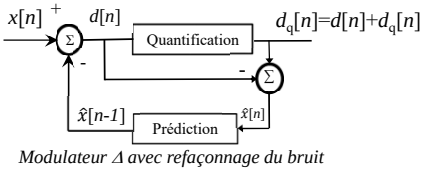
- Le pas de quantification Δ joue un rôle important
  - Δ est grand ⇒ bruit de granularité
  - Δ est petit ⇒ trop de distorsion



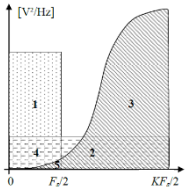
MD avec Refaçonnage du bruit (noise shaping)

- Suréchantillonner peut coûte cher en ressources !
  - K=4^{16} pour avoir l'équivalent de 16 bits
- Une meilleure approche consiste à rejeter le bruit hors des fréquences d'intérêt ⇒ Modulateur avec refaçonnage du bruit





Modulateur Δ avec refaçonnage du bruit



Fonctions de densité spectrale de puissance du bruit de quantification pour : 1) échantillonnage de Nyquist ; 2) sur-échantillonnage de facteur K; 3) avec modulation Δ; 4) avec filtrage numérique passe bas et sans modulation Δ; 5) avec filtrage numérique et modulation Δ.

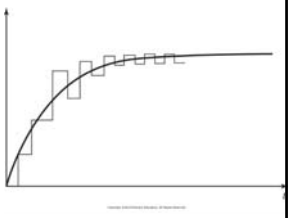
- On a $D_q[z] = X[z]+(1-z^{-1})B_q[z]$

Signal non affecté

Filtrage passe haut du bruit

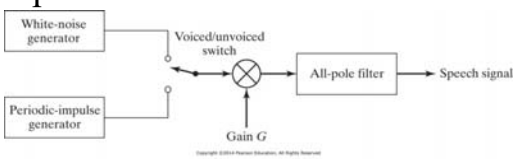
La modulation delta adaptative

- Adapte la valeur de Δ aux changements du signal d'entrée
- On agit suivant les signes de deux valeurs successives de $\hat{Y}_n$:
  - Même signe : on augmente Δ
  - Signes différents : on diminue Δ
- Une règle simple $\Delta_n = \Delta_{n-1} K^{\epsilon_n \times \epsilon_{n-1}}$ où $K>1$ est une constante et ϵ_n est la sortie du quantificateur
- L'amélioration est de 5-10 dB



Techniques d'analyse-synthèse

- Au lieu d'échantillonner et coder le signal d'origine, on paramètre un modèle pour le générer.
  - Les paramètres du modèle sont extrait du signal d'origine, quantifiés, encodés et transmis
- À la réception, le même modèle est synthétisé et utilisé pour reproduire le signal de la source
- Un exemple connu est le codage linéaire prédictif (LPC) utilisé pour la voix

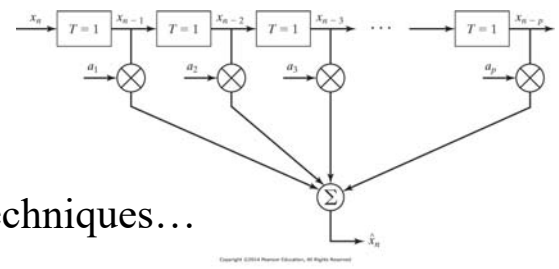


Techniques d'analyse-synthèse

- Pour la parole voisée, le signal à générer est modelé par une séquence périodique d'impulsions de fréquence f_0.
  - La hauteur (*pitch)* du signal généré dépend de f_0
- Le filtre est défini par $X_n = \sum_{i=1}^{P} a_i X_{n-1} + G w_n$
  - P est le nombre de pôles
  - $a_i$sont les coefficients du filtre
  - w_n est la séquence d'entrée (le processus d'innovation)
  - G est le paramètre de gain.

Filtre LPC

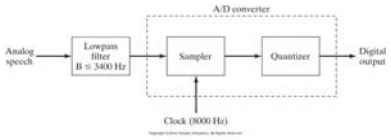
- Les coefficient du filtre sont déterminésr à partir de trames du durée 20-30 ms
  - Pour un filtre d'ordre p, $\hat{x}_n = \sum_{i=1}^{P} a_i x_{n-1}$
  - Il faut minimiser l'erreur $e_n = x_n - \hat{x}_n$ par rapport aux a_i.

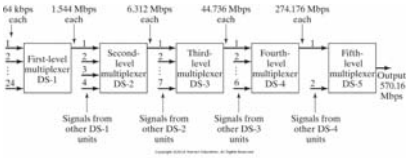


- Plusieurs techniques…

Transmission numérique de la voix

- Filtre passe bas avec f_c=3400Hz et échantillonnage avec f_s=8000Hz
- PCM non uniforme (loi-μ) à 64kps ou DPCM à 32kps dans les réseaux terrestres
  - Les appels sont regroupés par multiplexage temporel
- Dans les réseaux cellulaires, le codage LPC permet de transmettre à 4800-9600 bps

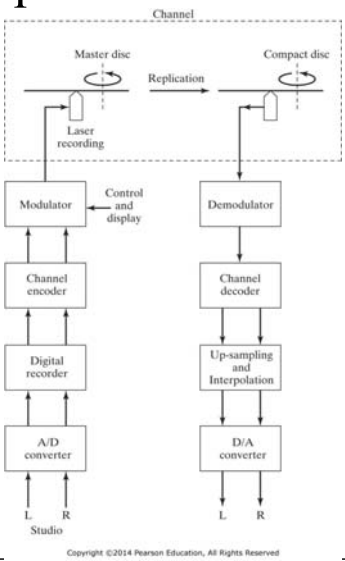




Hiérarchie TDM en Amérique du Nord

Enregistrement numérique de l'audio

- Deux canaux G et D
- Échantillonnage PCM uniforme
  - 44.1kHz, 16 bits/échantillon, SQNR>90dB, D = 0.005%
- Codage en blocs correcteur d'erreurs de lecture (Reed-Solomon)
- DAC de 16 bits à coût réduit grâce au suréchantillonnage et à l'usage d'un modulateur delta modifié.
- Modulateur avec insertion de champs d'information

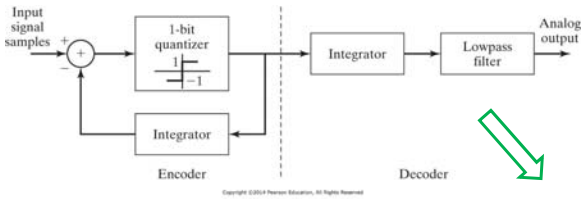


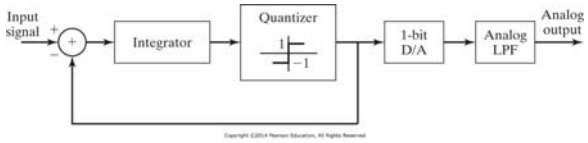
Copyright ©2014 Pearson Education, All Rights Reserved

Suréchantillonnage

- On insère *U*-1 zéros entre les échantillons de 16 bits, pour un taux d'échantillonnage effectif de *U*x44,1 kHz
  - U=256 donne f_s=11,2896 Mhz
- Les canaux G et D sont ensuite filtrés numériquement par un filtre RIF
  - Phase linéaire, largeur de bande 20 kHz
  - Élimine les répliques créées par le sur-échantillonnage
- Le suréchantillonnage et filtrage mène à une interpolation numérique où les échantillons ont des valeurs très proches (forte corrélation)

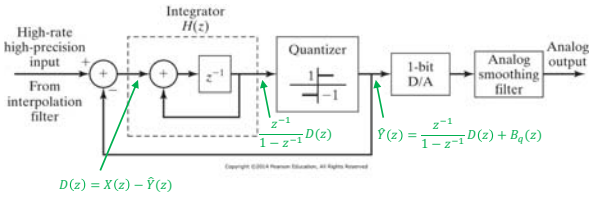
Modulateur sigma-delta





- Le déplacement de l'intégrateur simplifie le design tout en augmentant le SQNR
  - Élimine un intégrateur et réduit le bruit dans la bande passante

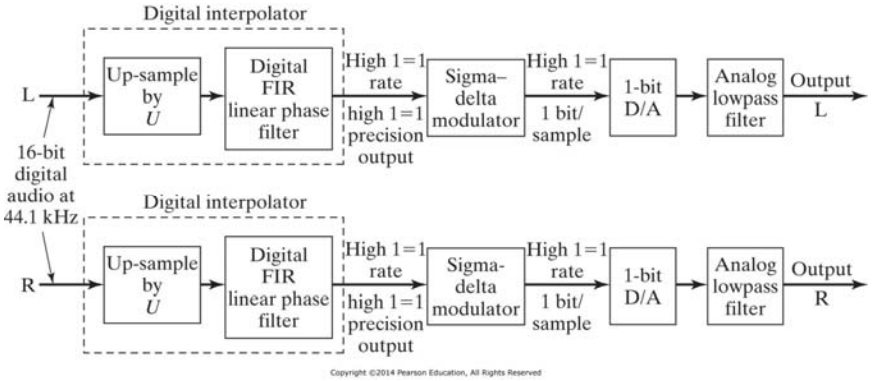
Modulateur sigma-delta



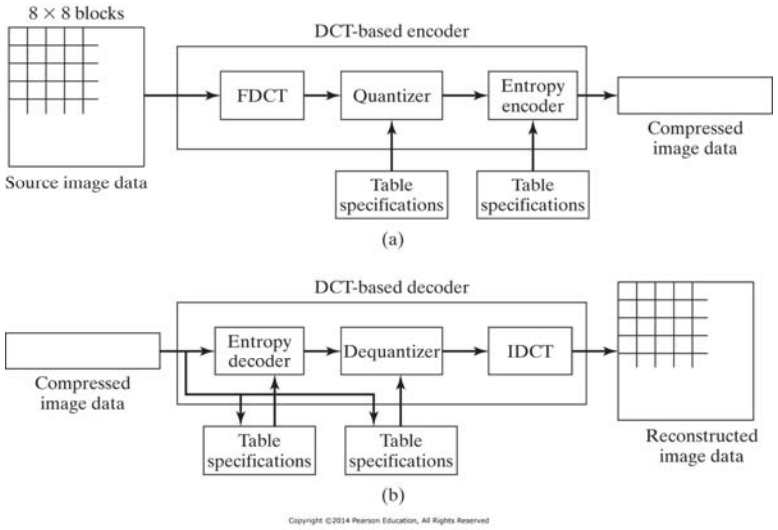
Délai Filtre passe haut

- On obtient : $\hat{Y}(z) = z^{-1}X(z) + (1 - z^{-1})B_q(z)$
- Le signal analogique peut être récupéré avec un simple comparateur et un filtre passe bas avec $f_c \sim 20$ kHz

Suréchantillonnage et modulation Σ-Δ



Codage et décodage JPEG



Copyright ©2014 Pearson Education, All Rights Reserved

Implémentation de la TCD

- Une TCD à 2D est requise
- Pour un bloc de pixels de taille 8x8, on a:
  - TCD :

$$F(u,v) = \frac{1}{4}\left(\frac{1}{\sqrt{2}}\right)^{\delta(u)+\delta(v)} \sum_{y=0}^{7}\sum_{x=0}^{7} f(x,y)\cos\left[\frac{(2x+1)\pi u}{16}\right]\cos\left[\frac{(2y+1)\pi v}{16}\right]$$

  - TCDI :

$$f(x,y) = \sum_{v=0}^{7}\sum_{u=0}^{7}\left(\frac{1}{\sqrt{2}}\right)^{\delta(u)+\delta(v)} F(u,v)\cos\left[\frac{(2x+1)\pi u}{16}\right]\cos\left[\frac{(2y+1)\pi v}{16}\right]$$

- À l'instar de la TFD, la TCD décompose un signal en une série pondérée de fonctions harmoniques, dans ce cas des cosinus.

Quantification des coefficients

- Chaque coefficient de la TCD obtenue est divisé par un entier entre 1 and 255 et le résultat est arrondi (codage perceptuel).
  - Les diviseurs sont emmagasinés dans un tableau choisi pour donner juste la précision nécessaire (# bits) pour chaque coefficient.
- Le tableau de quantification accompagne le fichier compressé.
- La quantification est la raison principale faisant que la compression JPEG est à pertes

Example de quantification

128	128	128	128	128	128	128	128
118	111	112	117	120	123	123	122
125	121	115	111	119	119	118	117
120	121	113	113	125	124	115	108
120	120	116	119	124	120	115	110
117	113	111	122	120	110	116	119
109	113	111	122	120	110	116	119
111	121	124	118	115	121	117	113

Données brutes

① DCT

-80	4	-6	6	2	-2	-2	0
24	-8	8	12	0	0	0	2
10	-4	0	-12	-4	4	4	-2
8	0	-2	-6	10	4	-2	0
18	4	-4	6	-8	-4	0	0
-2	8	6	-4	0	-2	0	0
12	0	6	0	0	0	-2	-2
0	8	0	-4	-2	0	0	0

② Division

16	11	10	16	24	40	51	61
12	12	14	19	26	58	60	55
14	13	16	24	40	57	69	56
14	17	22	29	51	87	80	62
18	22	37	56	68	109	103	77
24	35	55	64	81	104	113	92
49	64	78	87	103	121	120	101
72	92	95	98	112	100	103	99

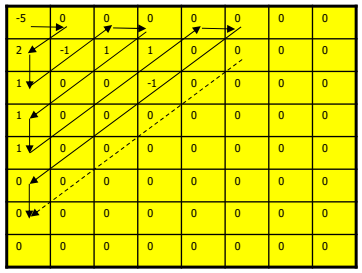
Matrice de quantification

-5	0	0	0	0	0	0	0
2	-1	1	1	0	0	0	0
1	0	0	-1	0	0	0	0
1	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

Valeurs quantifiées

Arrangement en zigzag

- Crée une séquence de valeurs où les basses fréquences sont groupées au début
- On obtient aussi des suites de valeurs égales (e.g. 0) ou de valeurs proches, ce qui simplifie les codages
  - RLE pour les coefficients DC
  - Huffman pour les autres coefficients



0,2,1,-1,0,0,1,0,1,1,0,0,1,0,0,0,-1,0,0,… 0

Peut être encodé en RLE sous le format {# zéros, prochaine valeur $\neq 0$} :

(1,2),(0,1),(0,-1),(2,1),(1,1),
(0,1),(0,1),(2,1),(3,1),EOB

Résumé

- Conversion des signaux analogiques en numériques
  - Échantillonnage, théorème de Nyquist
  - Quantification, erreur de quantification, ratio SQNR
  - Encodage
- Conversion des signaux numériques en analogique
  - Modulation impulsionnelle codée
  - Modulation delta
- Numérisation de la voix
- Prochain chapitre: modulation numérique pour les canaux AWGN