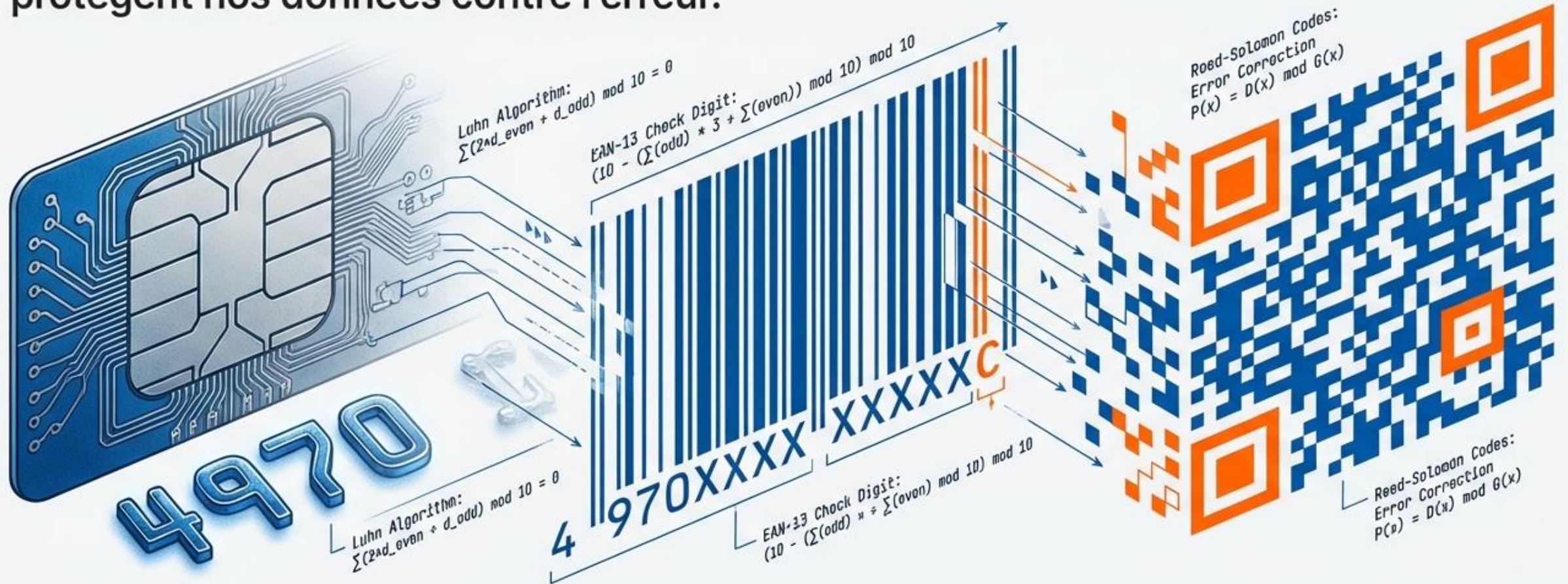


Codes Secrets du Quotidien : L'Invisible Architecture de la Fiabilité

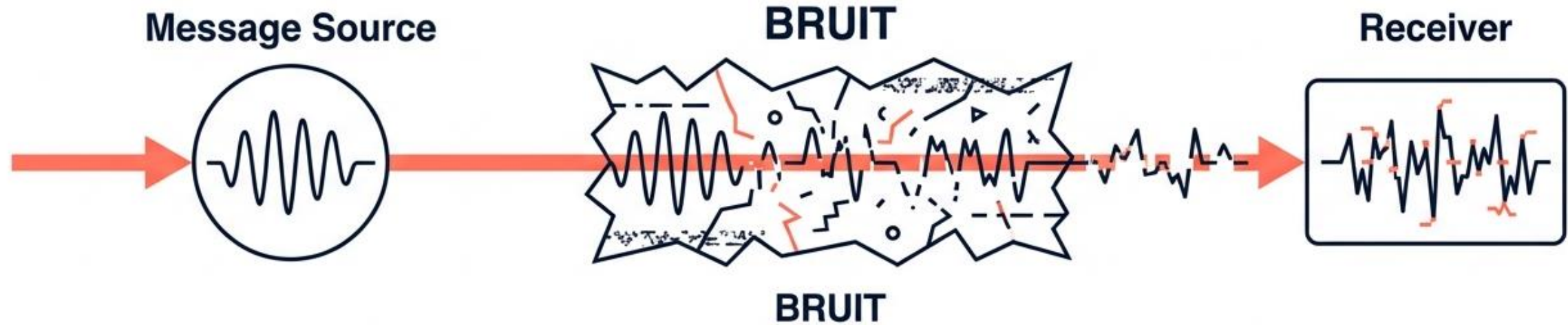
De la Carte Vitale au QR Code : comment les mathématiques protègent nos données contre l'erreur.



Une exploration de la détection et de la correction d'erreurs.

L'Ennemi : Le Bruit et l'Erreur Humaine

La **Redondance** n'est pas du gaspillage, c'est une **sécurité**.



La Menace

- **Canal Bruité (BSC)** : Inversion de bits, effacement, ou bruits électromagnétiques lors de la transmission.
- **Faillibilité Humaine** : Erreurs de saisie, inversion de chiffres adjacents (ex: taper 96 au lieu de 69).

0101001 1010
0110110 1010
01101 101010

96 → 96

La Stratégie

- **Détection (L'alerte)** : Le système sait que le message est corrompu mais ne peut pas le réparer (Ex: Carte Vitale, Carte Bancaire).
- **Correction (La réparation)** : Le système reconstitue le message original grâce à une forte redondance mathématique (Ex: QR Code, CD/DVD).



La Carte Vitale : La Robustesse du Nombre Premier



NIR (Numéro d'Inscription
au Répertoire) : 13 chiffres

La Clé :
60

$$\text{Clé} = 97 - (\text{NIR modulo } 97)$$

Pourquoi 97? 97 est le plus grand nombre premier < 100 . Il maximise la détection d'erreurs de saisie.

⚙ Exception Corse

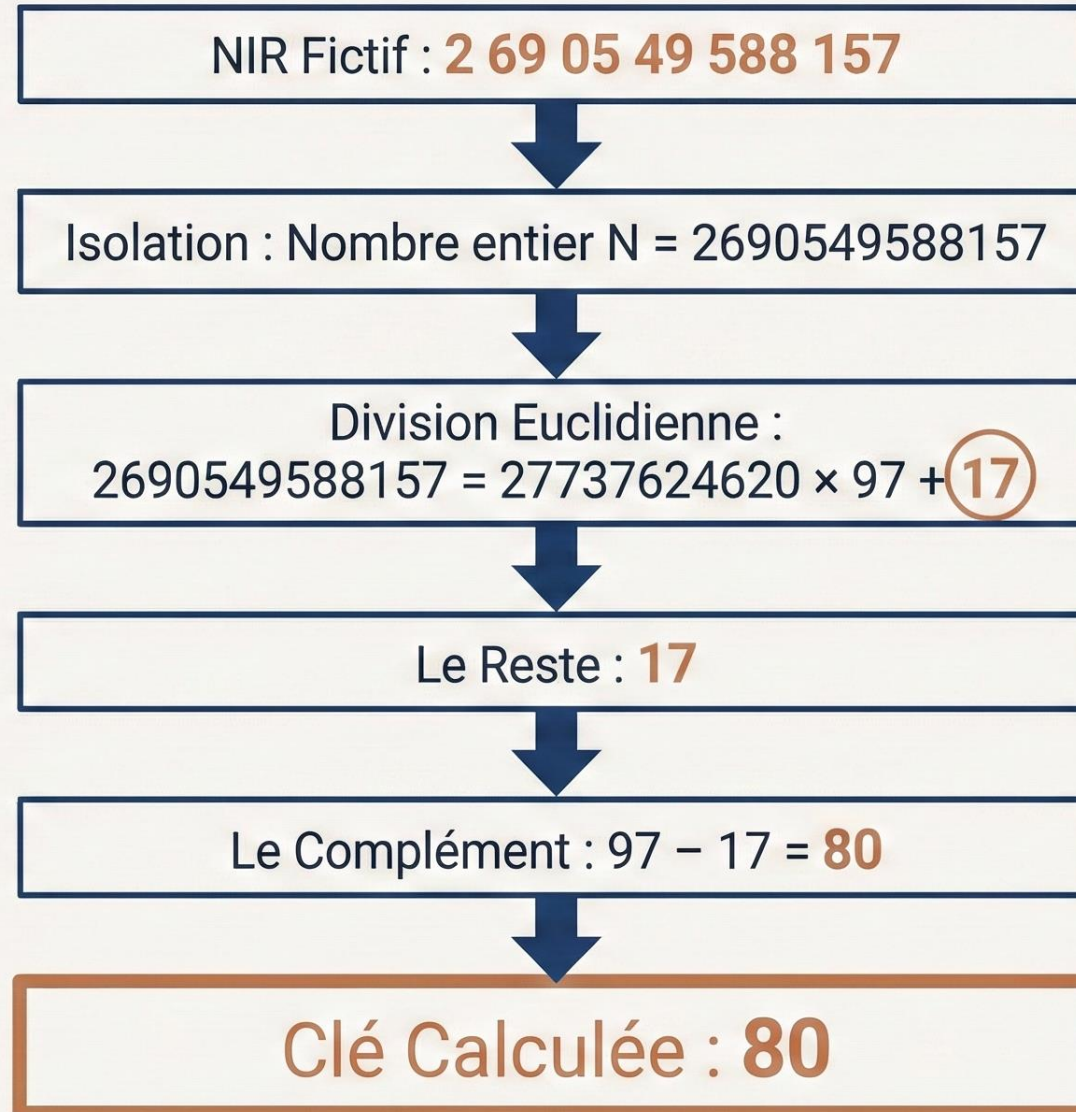
Départements 2A et 2B :

2A → 19

2B → 18

Nécessaire pour le calcul arithmétique.

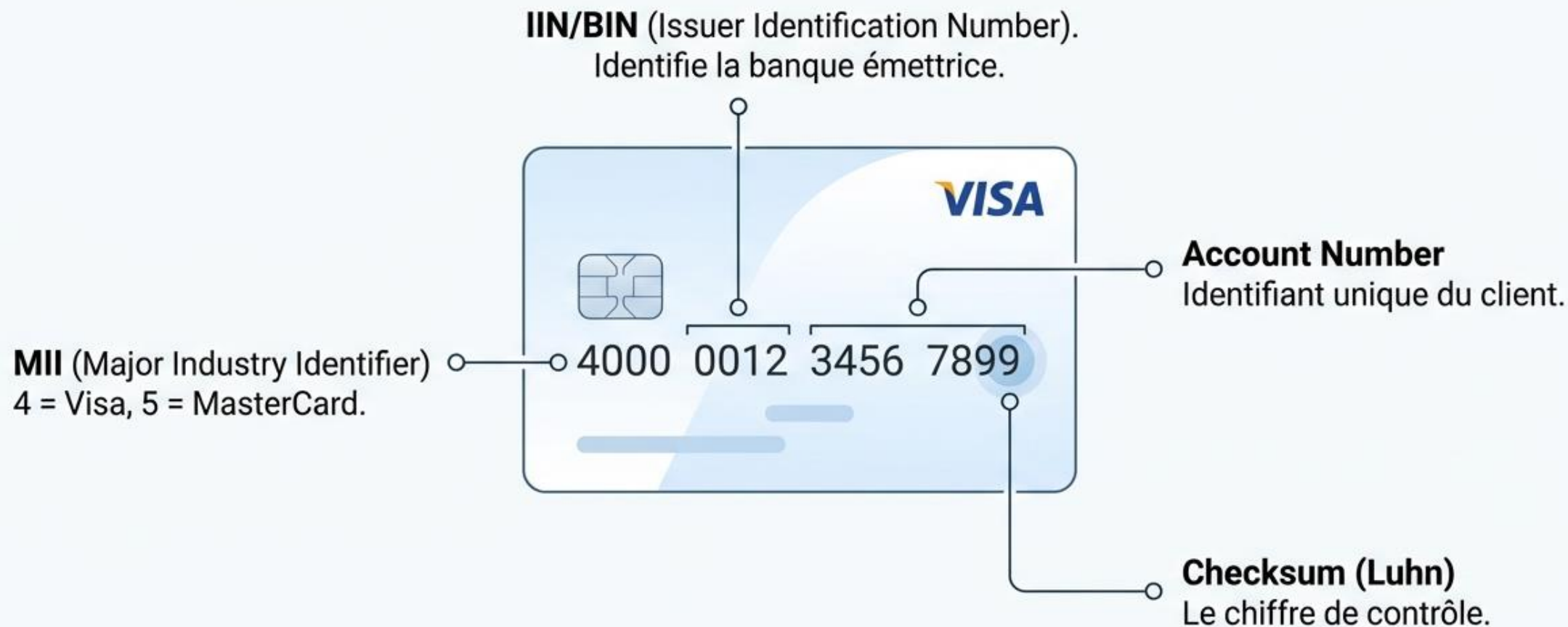
Calcul en Action : Vérification d'un NIR



Le numéro complet sur la carte est 2 69 05 49 588 157 80. Une seule erreur de frappe changerait le reste, invalidant la clé.

La Carte Bancaire : Anatomie du PAN

Primary Account Number (16 chiffres)



L'Algorithme de Luhn (Modulo 10)

Inventé par Hans Peter Luhn (IBM), 1954.

Contexte



Mechanical vs. Digital

Conçu pour l'ère mécanique. Simple, rapide, efficace contre les erreurs humaines courantes (inversions de chiffres).

Les Règles (Checklist)

- ✓ **1. Direction** : Parcourir le numéro de droite à gauche.

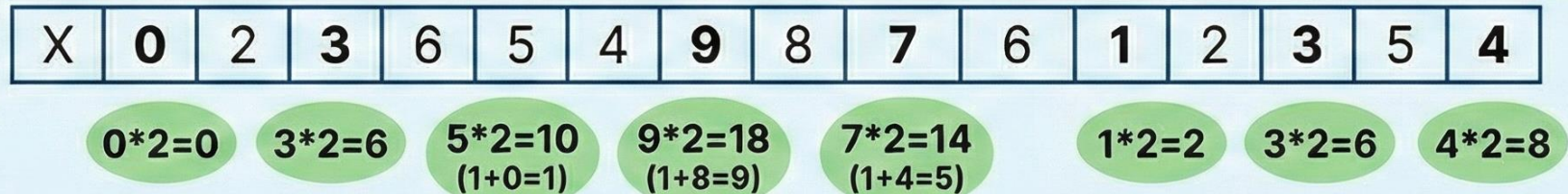
- ✓ **2. Doubler** : Multiplier par 2 un chiffre sur deux (positions paires).

- ✓ **3. Règle du > 9** : Si le résultat > 9, additionner les chiffres (ex: 16 → 1+6=7).
- ✓ **4. Validation** : La somme totale doit être un multiple de 10.
☐

Détection d'Erreur sur Carte Bancaire: L'Algorithme de Luhn



1. Inverser et Doubler
(Chiffres en gras)



2. Additionner
Tous les Chiffres
(y compris ceux non doublés)

$$X + 0 + 2 + 6 + 6 + 1 + 4 + 9 + 8 + 5 + 6 + 2 + 2 + 6 + 5 + 8 = \text{Somme Totale (S)}$$

3. Vérifier la Validité
(Somme Totale modulo 10 = 0)

Somme Totale (S)

$$S \% 10 == 0?$$

✓ **Numéro VALIDE**
(Pas d'erreur détectée)

✗ **Numéro INVALIDE**
(Erreur détectée)

4. Calculer le Chiffre de Contrôle
(Si X est inconnu)

Somme Partielle (sans X) = S'

$$X = 10 - (S' \% 10)$$

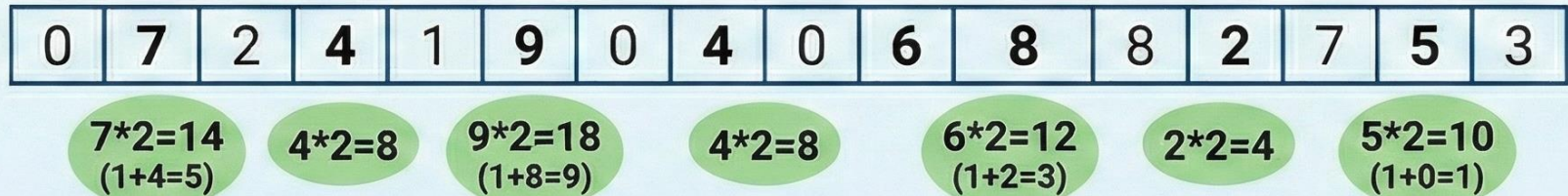
$$\text{Si } S'=73, \text{ alors } X = 10 - (73\%10) = 10 - 3 = 7$$

Détection d'Erreur sur Code IMEI: L'Algorithme de Luhn



Clé de Contrôle

1. Inverser et Doubler
(Chiffres en gras)



2. Additionner
Tous les Chiffres
(y compris ceux non doublés)

$$0 + 5 + 2 + 8 + 1 + 9 + 0 + 8 + 0 + 3 + 8 + 4 + 7 + 1 + 3 = \text{Somme Totale (S)}$$

3. Vérifier la Validité
(Somme Totale modulo 10 = 0)

Somme Totale (S)

$$S \% 10 == 0?$$

IMEI VALIDE

IMEI INVALIDE

4. Calculer la Clé de Contrôle
(Si Clé inconnue)

Somme Partielle (sans Clé) = S'

$$\text{Clé} = (10 - (S' \% 10)) \% 10$$

$$\text{Si } S' = 57, \text{ alors Clé} = (10 - (57 \% 10)) \% 10 = (10 - 7) \% 10 = 3$$

PRINCIPAUX TYPES DE CODES-BARRES

CODES-BARRES 1D (LINÉAIRES)



EAN-13 / UPC

Commerce de détail
(Standard mondial).
Vente en caisse.



EAN-8

Petits produits
(Surface limitée).



Code 128

Logistique & Transport
(Haute densité, flexible).

CODES-BARRES 2D (MATRICIELS)



QR Code

Marketing, Liens Web,
Paiement mobile
(Lecture rapide).



Data Matrix

Industrie, Pharma,
Marquage direct
(Très petite taille, sécurisé).



PDF417

Documents, Cartes
d'embarquement
(Stockage de données élevées).

Les codes 1D sont lus linéairement, les 2D sont lus par caméra et stockent plus de données.

EAN-13 : La Logistique du Quotidien



Structure : 12 chiffres de données + 1 clé.

Impact : Lecture optique rapide en caisse, tolérance aux déformations.

Pondération Alternée

Digits	4	7	1	9	5	1	2	0	0	2	8	8
Weights	x1	x3	x1	x3	x1	x3	x1	x1	x3	x1	x1	x3

$$(4 \times 1) + (7 \times 3) + (1 \times 1) + (9 \times 3) + (5 \times 1) + (1 \times 3) + (2 \times 1) + (0 \times 3) + (0 \times 1) + (2 \times 3) + (8 \times 1) + (8 \times 3) = 101$$

Règle : Le total + Clé doit être un multiple de 10.

Au-delà de la Détection : La Correction d'Erreur

Quand le code doit survivre aux dommages.



Les codes précédents rejettent l'erreur (Bip d'erreur).



Le QR Code est conçu pour **reconstituer** l'information manquante.

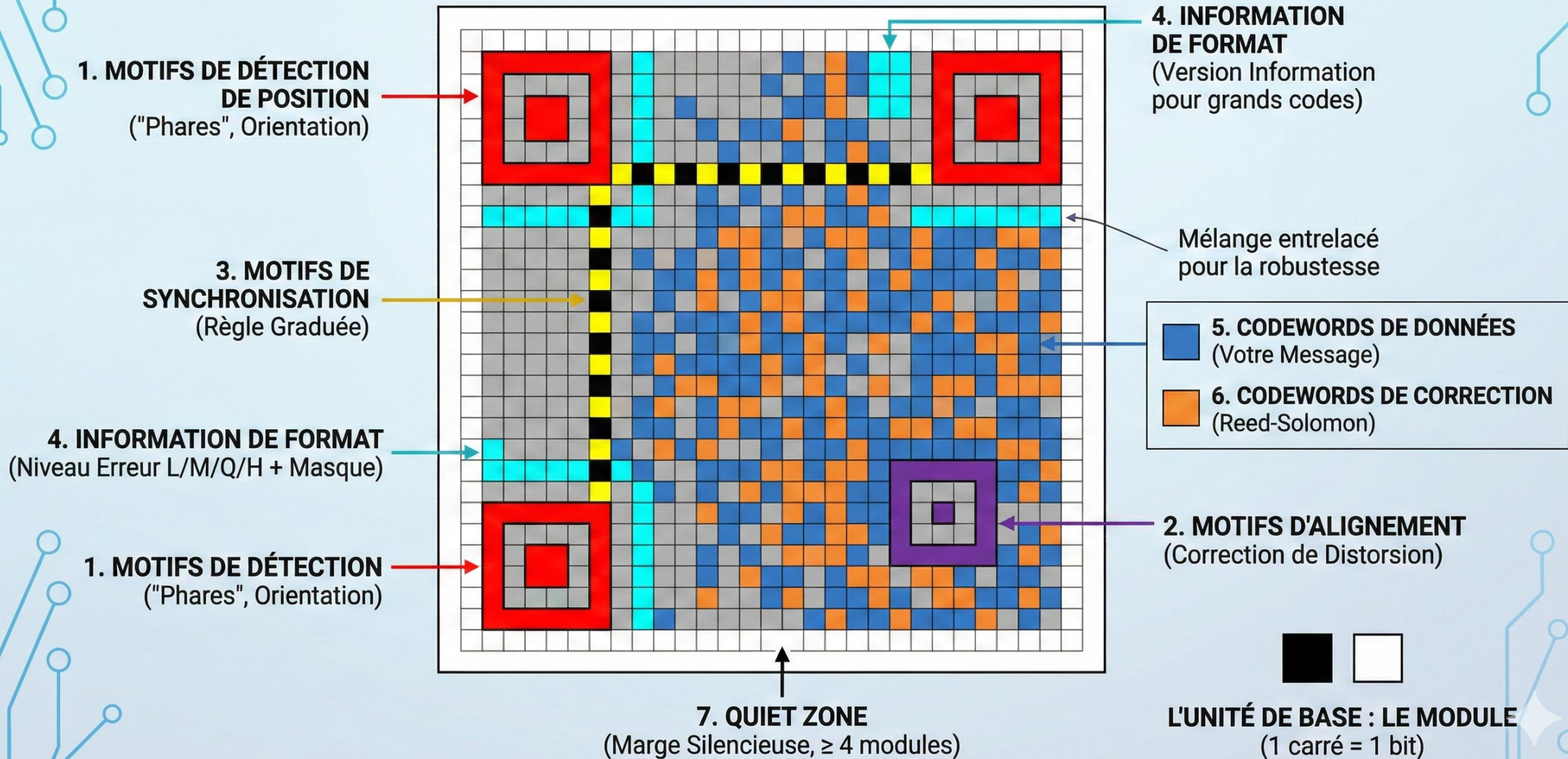


✓ Toujours Lisible



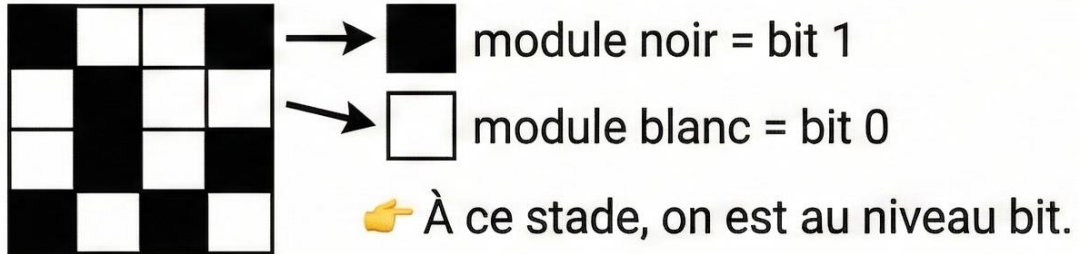
Un QR code peut rester lisible même avec **30%** de sa surface détruite (Niveau H).

ANATOMIE D'UN QR CODE : DÉCONSTRUCTION EN COUCHES

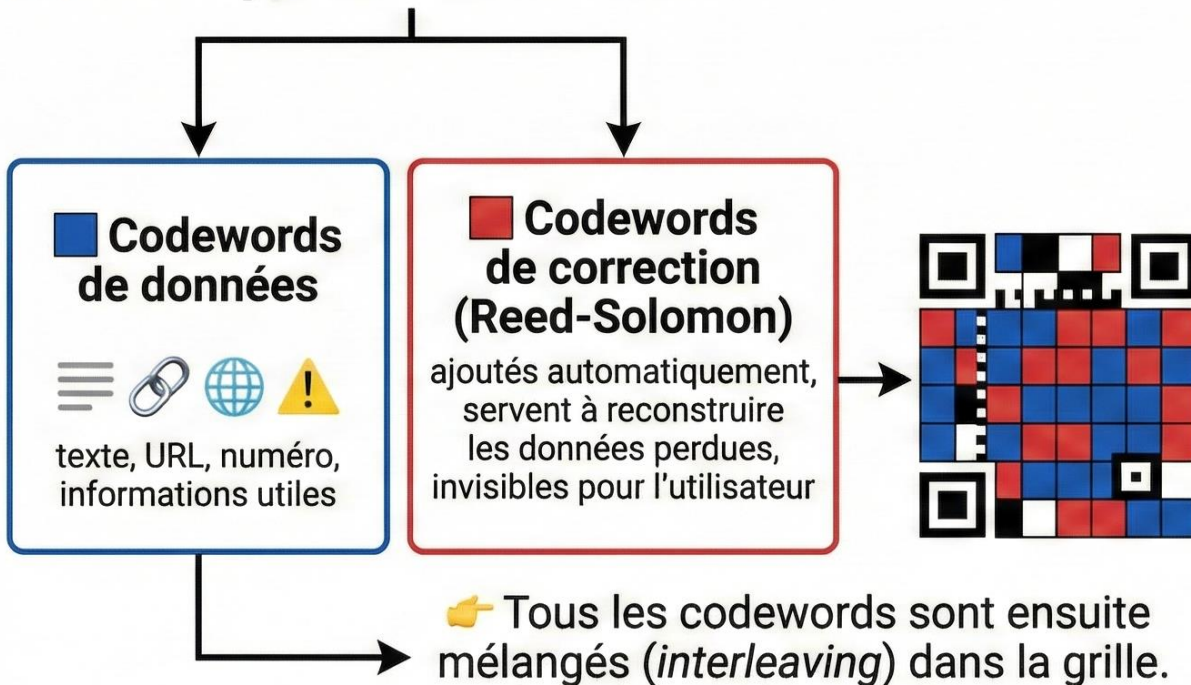


LA STRUCTURE D'UN QR CODE : DE L'OCTET À REED-SOLOMON

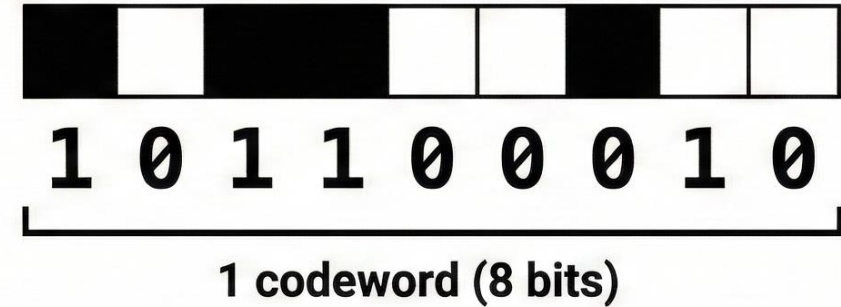
1 La base : modules noirs et blancs



3 Deux types de codewords

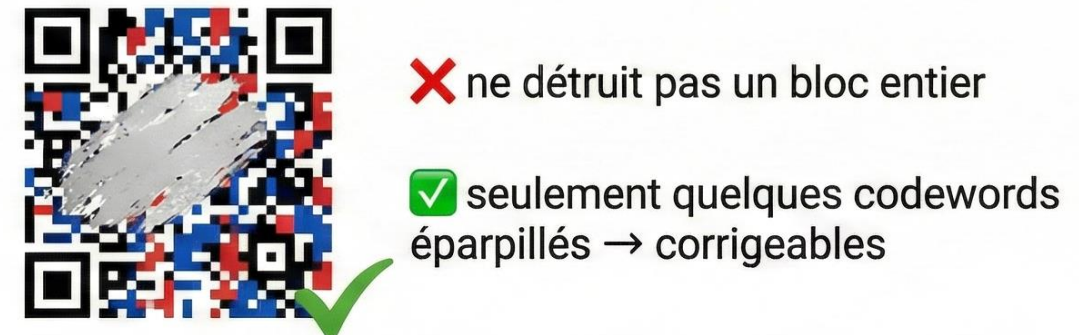


2 Du bit au codeword



Un codeword = une petite "brique" de données.

4 Pourquoi l'interleaving est important (intuition)



« Un QR code n'est pas lu carré par carré, mais octet par octet. Ces octets s'appellent des codewords, et Reed-Solomon permet d'en corriger certains s'ils sont faux. »

Reed-Solomon en Action : Les Niveaux de Protection



Level L (Low) - 7% restored.



Level M (Medium) - 15% restored.



Level Q (Quartile) - 25% restored.



Level H (High) - 30% restored.

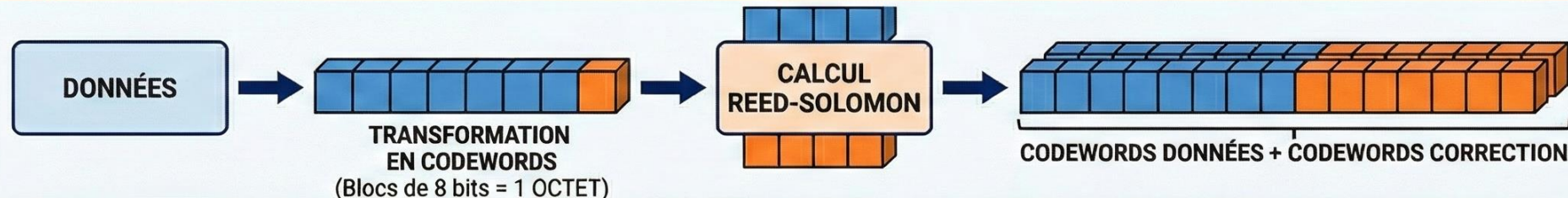
Plus le niveau de correction est élevé, plus le QR code est dense (contient plus de redondance).

Le niveau H permet d'insérer des logos ou de survivre à une tache de café.

QR Code et Correction d'Erreur : Le Secret des Reed-Solomon

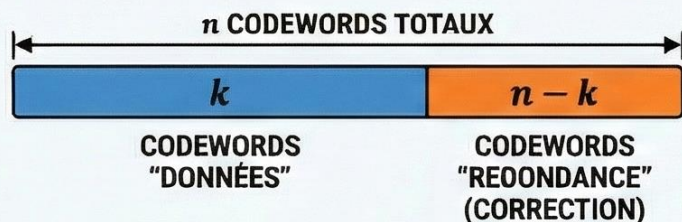


💡 **CONCEPT CLÉ : QR CODE = REED-SOLOMON SUR DES CODEWORDS**



LE MODÈLE RS(n, k) (EN BTS)

RS(n, k)

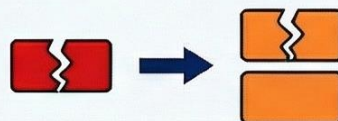


k = nombre de codewords "données".
 $n - k$ = nombre de codewords "redondance"

CAPACITÉ DE CORRECTION (RÈGLE BTS)

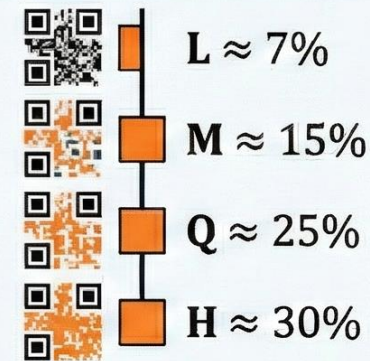
$$t = \frac{n - k}{2}$$

ON PEUT CORRIGER t CODEWORDS FAUX (ENVIRON)



Idée Simple : Pour corriger 1 codeword erroné, il faut ~ 2 codewords de redondance.

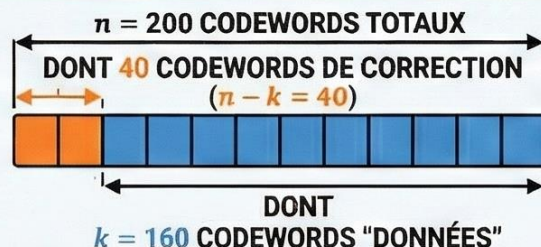
LES NIVEAUX D'ERREUR QR (CHOIX DE REDONDANCE)



Pourcentages exprimés sur le **TOTAL DES CODEWORDS** (Données + Correction)

EXEMPLE CONCRET

Exemple Pédagogique : UN QR CODE
(Après Mise en Forme)



CALCUL :

$$n - k = 40$$
$$t = \frac{40}{2} = 20$$

→ LE QR CODE PEUT CORRIGER ENVIRON 20 CODEWORDS CORROMPUS (≈ 20 octets "FAUX") DANS CE BLOC



COMPROMIS :
NIVEAU PLUS FORT (Q/H) = PLUS DE REDONDANCE ($n - k$ augmente)
= MOINS DE PLACE POUR LES DONNÉES (QR plus dense / version plus grande).



NIVEAU H



NIVEAU L

APPLICATIONS DES CODES DE REED-SOLOMON (RS) :

Correction d'Erreurs et Récupération de Données

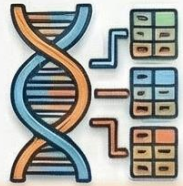


STOCKAGE OPTIQUE & NUMÉRIQUE

Disques (CD/DVD/BD) : Corrige les rayures et défauts de surface.

Disques Durs (HDD/RAID) & SSD : Répare les secteurs défectueux et erreurs de mémoire flash.

Codes-barres 2D (QR Code, Data Matrix) : Permet la lecture même si endommagé ou sale.



ARCHIVAGE & HAUTE FIABILITÉ

Stockage sur ADN (Futur) : Protège les données stockées dans des molécules biologiques.

Archivage Long Terme : Garantit l'intégrité des données sur des décennies (bandes, disques optiques spéciaux).

Systèmes Critiques : Utilisé pour une fiabilité maximale.



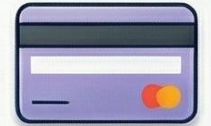
TÉLÉCOMMUNICATIONS & TRANSMISSION

Satellites & Espace Lointain : Répare les données corrompues par le bruit cosmique (ex: sondes Voyager).

Réseaux Mobiles & Wi-Fi : Améliore la fiabilité des transmissions sans fil.

Diffusion TV Numérique (TNT, Satellite) : Assure une réception stable de l'image et du son.

Modems ADSL/VDSL : Corrige les erreurs sur les lignes téléphoniques.



LECTURE & CAPTURE DE DONNÉES

Scanners de Codes-barres : Permet la lecture rapide et fiable même avec des impressions imparfaites.

Systèmes RFID & Cartes Magnétiques : Corrige les erreurs lors de la lecture sans contact ou par balayage.

L'ALGORITHME OACI DOC 9303 : CONTRÔLE DES PASSEPORTS & CNIe (Intégrité MRZ)

OÙ IL S'APPLIQUE (Documents Standards)



**Passeports biométriques,
Nouvelles CNI électroniques** (depuis 2021).
Vérifie l'intégrité des données lues (OCR).

LE CŒUR DU CALCUL (Détection d'Erreurs de Lecture)



1. CONVERSION DES CARACTÈRES

A-Z → 10-35	→	
0-9 → 0-9	→	7 4 0 5 1 2
< → 0	→	



2. PONDÉRATION RÉPÉTITIVE (7, 3, 1)

7	4	0	5	1	2
x	x	x	x	x	x
7	3	1	7	3	1...



3. SOMME & MODULO 10



EXEMPLE PRATIQUE : CALCUL D'UNE DATE (AAMMJJ)

1. Données (Ex: 740512)

7	4	0	5	1	2
---	---	---	---	---	---

2. Poids (Motif 7-3-1)

x7	x3	x1	x7	x3	x1
----	----	----	----	----	----

3. Produits

49	12	0	35	3	2
----	----	---	----	---	---

Somme = 49+12+0+35+3+2 = **101**

Résultat Final (Modulo 10) : 101 % 10 = **1**

Note: Différent de Luhn (poids 1,2). L'ancienne CNI française (avant 2021) utilisait un système propriétaire pour sa 1^{ère} ligne.