

Codage en ligne

Table des matières

1. TTL.....	2
2. NRZ.....	2
3. NRZI.....	2
4. NRZM.....	2
5. Codage Manchester.....	3
6. Codage Manchester différentiel.....	4
7. Le codage Miller.....	4
8. Codage bipolaire.....	4
9. Codage BHDn.....	5

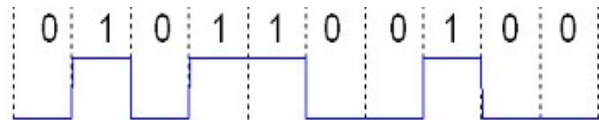
Pour le transport de données numériques, le codage en ligne est souvent utilisé. Il consiste à représenter le signal numérisé à transporter, par un autre signal qui présente des variations d'amplitude régulièrement espacées dans le temps. Après le codage en ligne, le signal peut être directement émis sur le canal de transmission, sous la forme de variations de la tension ou du courant.



1. TTL

La technologie Transistor-Transistor Logic est normalisée pour une tension d'alimentation de 5 V. Un signal TTL est défini comme niveau logique bas entre 0 et 1,4 V, et comme niveau logique haut entre 2,4 V et 5 V.

On ne peut transmettre les signaux émis par les circuits TTL sans circuits de transmission additionnels sur une longueur maximum environ 15 m.



2. NRZ

Le codage Non Return to Zero est à deux états : le signal se trouve dans un état (par exemple à l'état haut) lorsque des 1 logiques sont transmis, et dans l'autre état (à l'état bas dans l'exemple) lorsque des 0 logiques sont transmis. Ces deux états correspondent à deux niveaux de tension symétriques par rapport à 0.

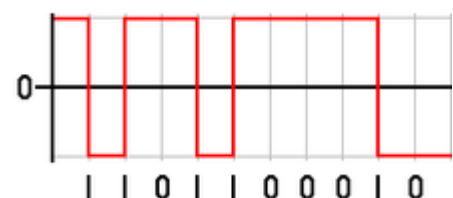


NB : l'état haut ne correspond pas toujours au 1 logique. Notamment dans le cas de la liaison série RS-232, le 0 est codé +12 V et le 1 est codé -12 V.

3. NRZI

Le codage Non Return to Zero Inverted est une variante du codage NRZ. Le inverted (inversé) fait référence au code NRZM créé en premier. On trouve également la notation NRZS (S pour Space) à la place de NRZI.

Le NRZI contrairement au NRZ crée une transition d'état si le bit est 1, et reste à l'état précédent si le bit est 0.



Le codage est à deux états. Il n'existe pas d'état intermédiaire.

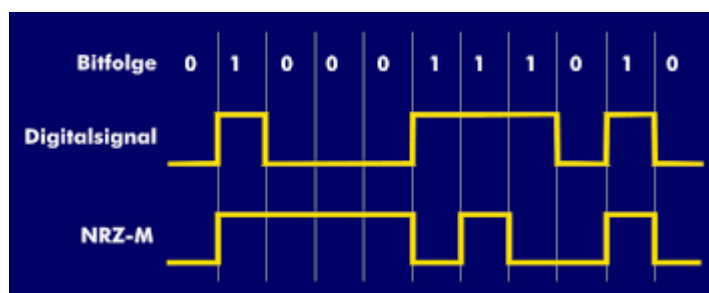
Le bus USB utilise le codage NRZI1. Le 0 a été choisi comme élément de changement. Pour éviter la perte d'horloge évoquée ci-dessous, un 0 est envoyé après six 1 consécutifs. Le récepteur doit prendre en compte ces éléments de remplissage (stuffing).

Attention toutefois, cela peut être l'inverse. Un 1 est alors envoyé après six 0 consécutifs. D'où l'importance que l'émetteur et le récepteur s'accordent sur la même logique.

4. NRZM

Le codage Non Return to Zero Mark est une variante du codage NRZ.

La valeur 1 crée une transition.

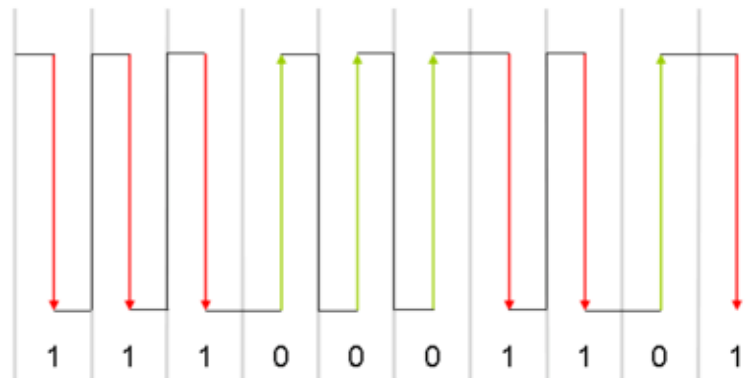


5. Codage Manchester

Le codage Manchester est un codage synchrone, ce qui signifie que, outre les données à transmettre, les signaux transmis intègrent également l'horloge de synchronisation nécessaire à leur décodage. Il est utilisé dans les réseaux informatiques pour injecter sur le média physique (couche 1 du modèle OSI) les valeurs logiques correspondant au flux d'entrée.

Les transitions du signal codé transmettent à la fois la valeur logique du bit (0 ou 1) et l'instant de son échantillonnage. Une transition intermédiaire est ajoutée lorsque des bits de même valeur se suivent.

Toutefois le schéma du code diffère entre le côté émetteur et le côté récepteur, même si leur référence commune est le signal de synchronisation :

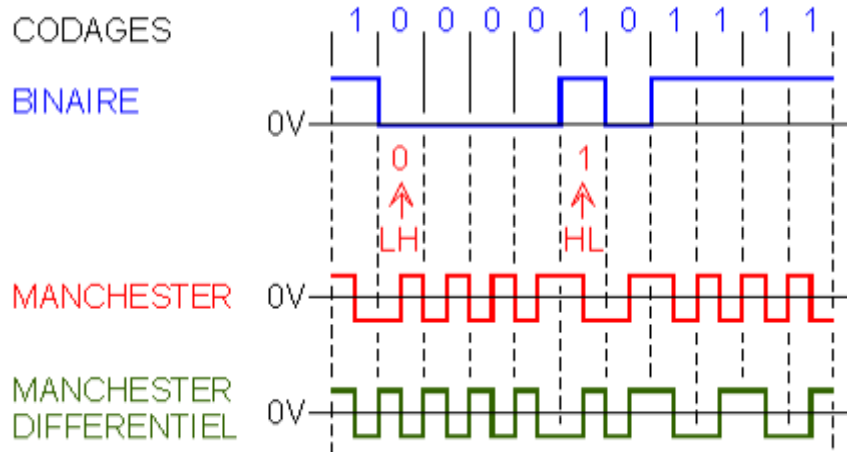


- du côté émetteur, le signal doit être stabilisé à la valeur du bit à transmettre au moment du top de l'horloge qui provoque la transition de synchronisation ;
- du côté récepteur, la transition marque le top d'échantillonnage de la valeur du bit : celui-ci est égal à la valeur qui précédait cet instant ou l'inverse de la nouvelle valeur qui suit la transition.

Si un codage simple peut s'effectuer en plaçant temporellement la synchronisation au centre du bit comme illustré ci-dessous, ce qui donne un signal "haut" durant 50% du temps, des émetteurs alimentés par piles peuvent réduire leur durée d'émission (état ON) en raccourcissant les périodes "hautes" avant ou après la transition de synchronisation, voir en ajoutant des transitions intercalaires.

6. Codage Manchester différentiel

Valeurs à coder	Valeurs transmises
0 logique	Transition dans le même sens que la précédente au début de l'intervalle.
1 logique	Transition dans le sens inverse de la précédente au milieu de l'intervalle.

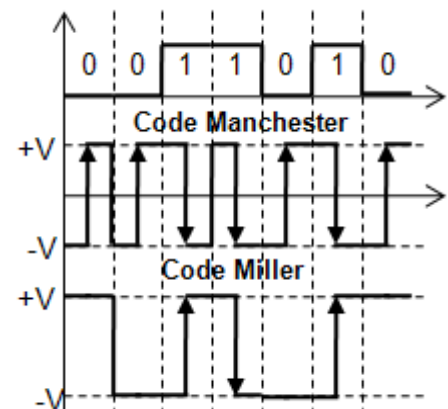


7. Le codage Miller

Signal intermédiaire identique au codage Manchester, puis suppression d'une transition sur deux.

Le codage peut être réalisé de la manière suivante :

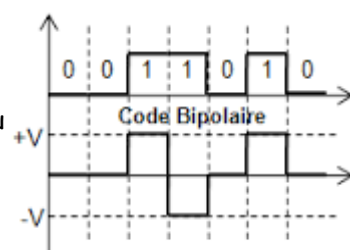
- transition (front montant ou descendant) au milieu du bit "1"
- pas de transition au milieu du bit "0"
- une transition en fin de bit "0" si celui-ci est suivi d'un autre "0"



8. Codage bipolaire

Le codage bipolaire est un codage à 3 niveaux comme les codages BHDn, c'est-à-dire que les valeurs utilisées pour coder le signal composé de 0 et de 1 varieront entre $-a$ et a . Il est utilisé dans les réseaux informatiques pour injecter sur le média physique (couche 1 du modèle OSI) les valeurs logiques correspondant au flux d'entrée.

Valeurs à coder	Valeurs transmises
0 logique	0
1 logique	opposé de la valeur transmise au 1 précédent.



9. Codage BHDn

Le codage BHDn (bipolaire à haute densité d'ordre n) est un codage à 3 niveaux comme les codages bipolaires, c'est-à-dire que les valeurs utilisées pour coder le signal composé de 0 et de 1 varieront entre $-a$, 0 et a . Il est utilisé dans les réseaux informatiques pour injecter sur le média physique (couche 1 du modèle OSI) les valeurs logiques correspondant au flux d'entrée.

Le 'n' de BHDn indique le nombre de 0 que l'on peut envoyer. On le choisit en fonction de la fiabilité du support et du matériel. La valeur pour le premier 1 à envoyer est fixée par convention entre l'émetteur et le récepteur.

- Si on doit envoyer un 1, on envoie l'inverse de la valeur envoyée pour le pas précédent.
- Si on doit envoyer un 0 :
 - Si les n+1 bits suivants ne sont pas tous à 0, on continue le codage comme en codage bipolaire simple.
 - Si les n+1 bits suivants sont tous à 0, les n bits suivants sont codés à 0 et le n+1 sera codé avec la même valeur que le code du 1 précédent (on viole alors l'alternance).

