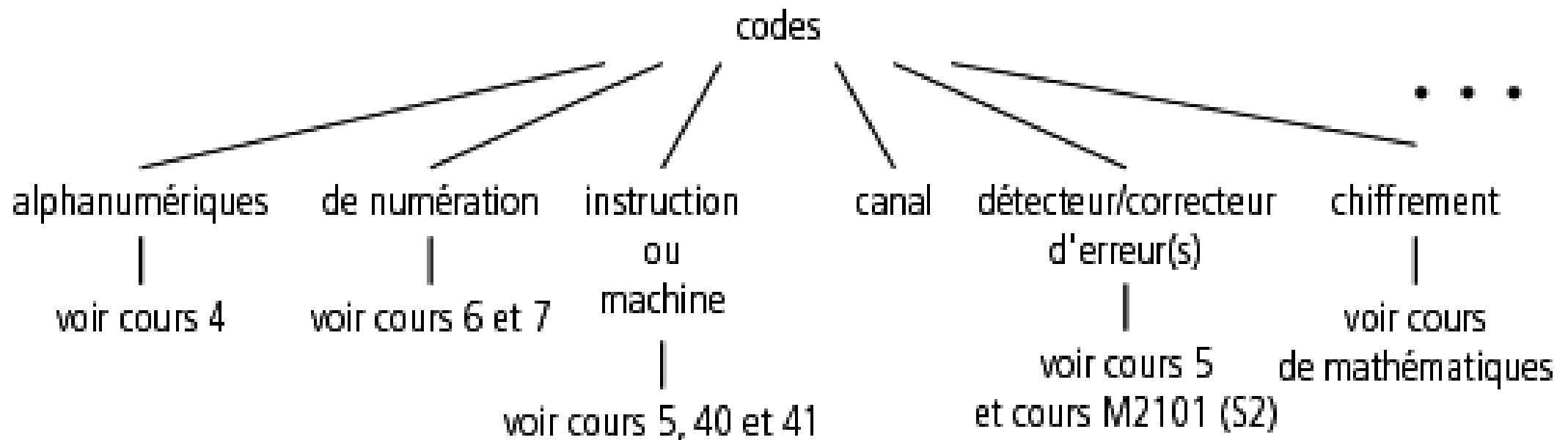


Architecture des ordinateurs

5 - Les autres familles de codes

Philippe Darche
IUT Paris Descartes

Classification des autres familles



Définitions complémentaires

- Code pondéré (*weighted code*)
 - poids positif et/ou négatif
- Code décimal
 - chaque chiffre décimal codé sur n bits ($n \geq 4$)
 - Exemple : code Lippel (non pondéré)

Décimal	Code Lippel			
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	1	0
5	1	1	1	1
6	1	1	1	0
7	1	1	0	1
8	1	1	0	0
9	1	0	0	1

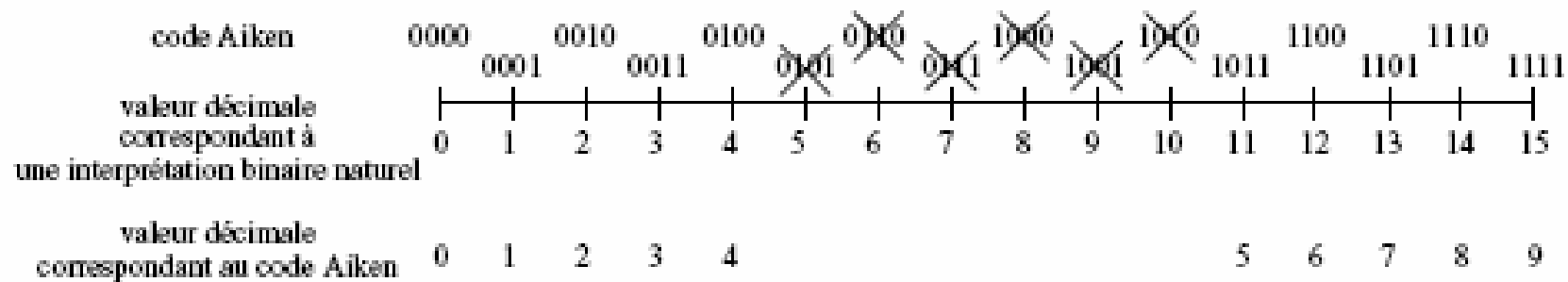
Définitions complémentaires

□ Code auto-complémentaire

- complémentation restreinte

- exemple : le code Aiken

- Code 2421 autocomplémenté



□ Code à excédent E

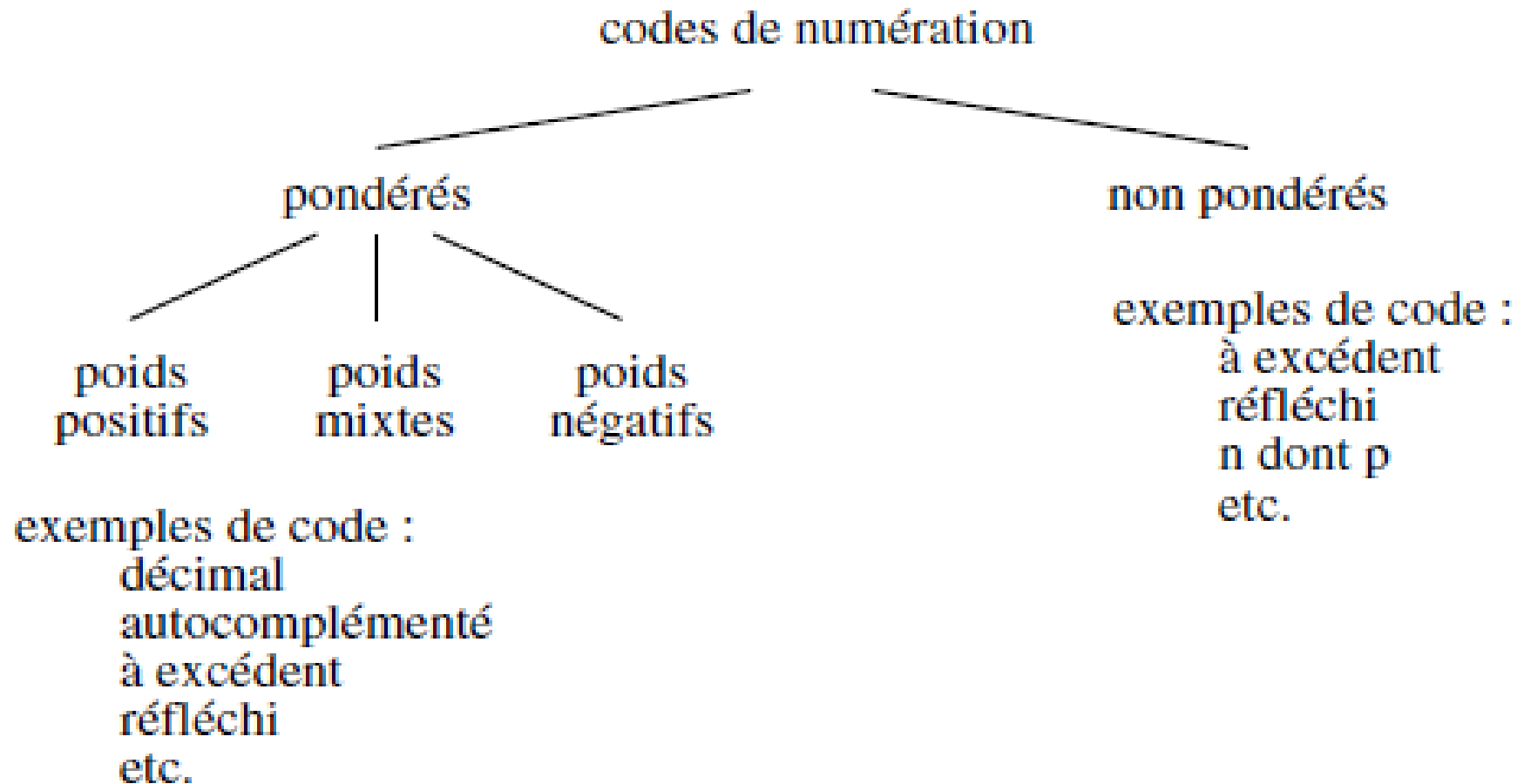
- $A_{\text{code_excédent}} = A_{\text{code}} + E$

Codes décimaux à poids positifs (n = 4)

- Symbole * : code auto-complémentaire
- Case colorée : unicité

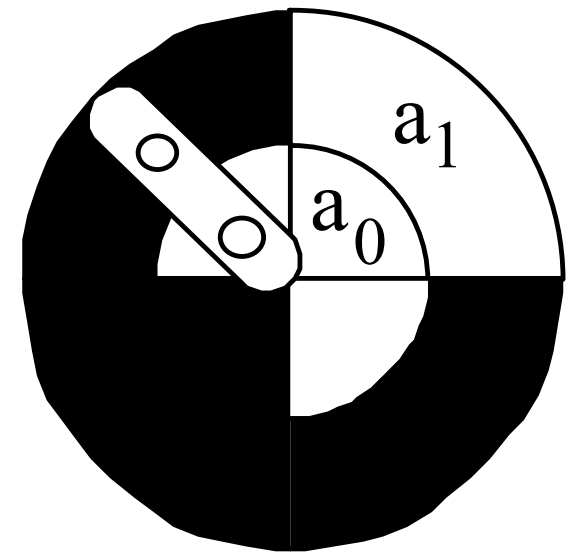
Code	Somme des poids	Etendue de l'excédent	Code	Somme des poids	Etendue de l'excédent
8742	21	6	*6321	12	0-3
8643	21	6	*6311	11	0-2
8641	19	4-6	*6221	11	0-2
8542	19	4-6	5433	15	3
8432	17	2-6	5432	14	2 et 3
*8421	15	0-6	*5421	12	0-3
7653	21	5-7	*5321	11	0-2
7541	17	4	*5311	10	0 et 1
7531	16	3 et 4	*5221	10	0 et 1
*7421	14	0-5	*5211	9	0
*7321	13	0-4	4432	13	2
6543	18	3-6	*4421	11	0-2
6542	17	4	*4321	10	0 et 1
6531	15	3	*4311	9	0
6432	15	2-4	*4221	9	0
*6421	13	0-4	*3321	9	0
6322	13	2			

Classement des codes de numération



Le code binaire réfléchi (*reflex code*)

- De ses inventeurs Louis Gros et Frank Gray (code Gros-Gray)
- L'état d'une seule variable change lors du passage d'un mot du code à un autre
- Code cyclique
- Utilisations classiques :
 - le tableau de Karnaugh
 - le codeur de position



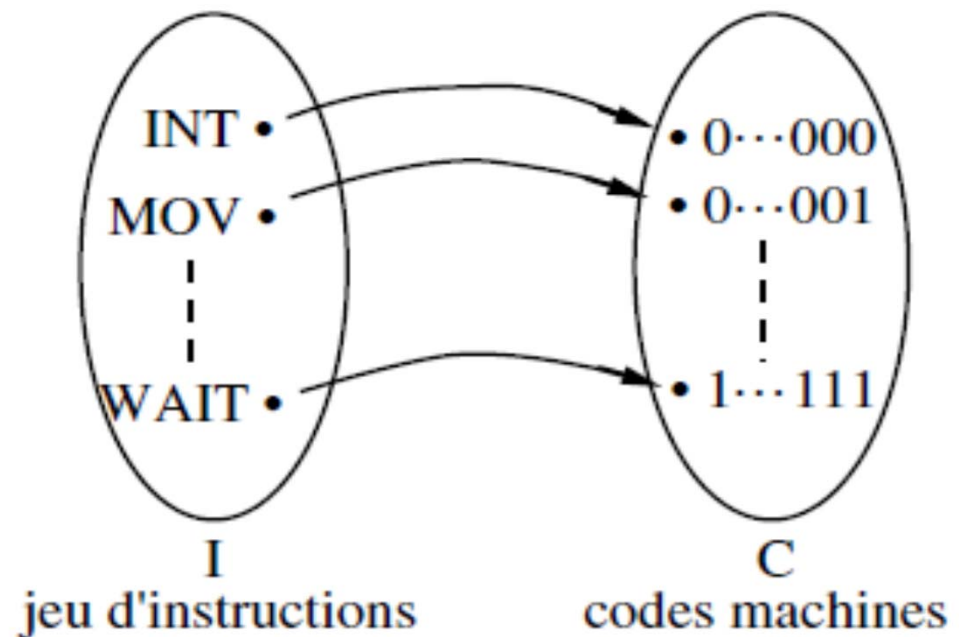
Le code binaire réfléchi

- Code réfléchi = présence d'un ou plusieurs axes spécifiques
 - données identiques de part et d'autres

0	0	0	0	
0	0	0	1	axe 1
0	0	1	1	
0	0	1	0	axe 2
0	1	1	0	
0	1	1	1	axe 1
0	1	0	1	
0	1	0	0	axe 3
1	1	0	0	
1	1	0	1	axe 1
1	1	1	1	
1	1	1	0	axe 2
1	0	1	0	
1	0	1	1	axe 1
1	0	0	1	
1	0	0	0	

Le code d'instruction

- Ou code opératoire
- Utilisation de mnémoniques (= abrégé d'instruction) pour faciliter la programmation



- Programmation de la partie contrôle
 - spécifique à un (ou une famille de) microprocesseur

Exemple d'un code en langage d'assemblage

```
.....
29                                     ; traitement de la valeur absolue
30
31 0005 8A 1E 0001r                    mov bl,[A]  ; chargement du multiplicande
32 0009 8A 0E 0003r                    mov cl,[B]  ; chargement du multiplicateur
33 000D 3A D9                          cmp bl,cl
34 000F 73 02                          jae suite  ; saut si le compteur de boucle est minimal
35 0011 86 D9                          xchg bl,cl
36
37
38 0013 32 ED                          suite: xor ch,ch
39 0015 E3 1A                          jcxz fin   ; saut si un des deux opérandes est nul
40 0017 32 FF                          xor bh,bh
41 0019 33 C0                          xor ax,ax
42
43 001B 03 C3                          boucle: add ax,bx
44 001D E2 FC                          loop boucle
45
46 001F A3 0005r                        mov [resultat],ax ; rangement du résultat
.....
```

Le code d'instruction

- Plusieurs champs:
 - le champ instruction
 - et éventuellement le(s) champs opérande(s)
- Exemples
 - NOP
 - ADD ax,2

Les codes détecteur (et correcteur) d'erreur(s)

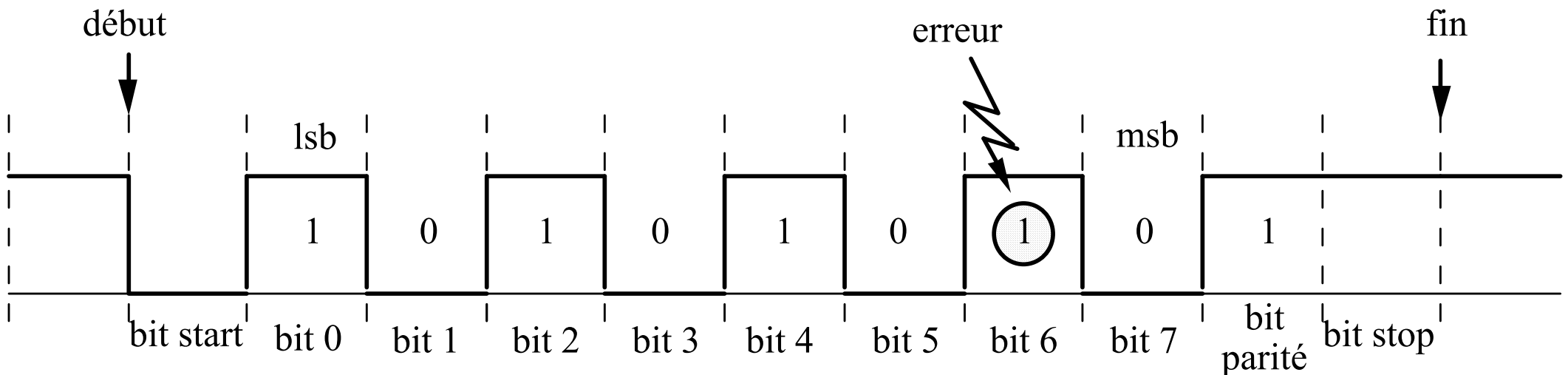
- Postulat de départ
 - une erreur = un bit complémenté
 - vrai qu'en base 2 !
- Un exemple : le contrôle par parité logique
 - détection d'erreur uniquement

La parité logique

- A ne pas confondre avec la parité numérique
- Parité logique paire ou impaire
 - nombre pair ou impair de '1' du mot binaire
- Exemples :
 - $4 = (0000\ 0100)_2 \rightarrow$ parité logique impaire
 - $6 = (0000\ 0110)_2 \rightarrow$ parité logique paire
 - $13 = (0000\ 1101)_2 \rightarrow$ parité logique impaire
 - $15 = (0000\ 1111)_2 \rightarrow$ parité logique paire

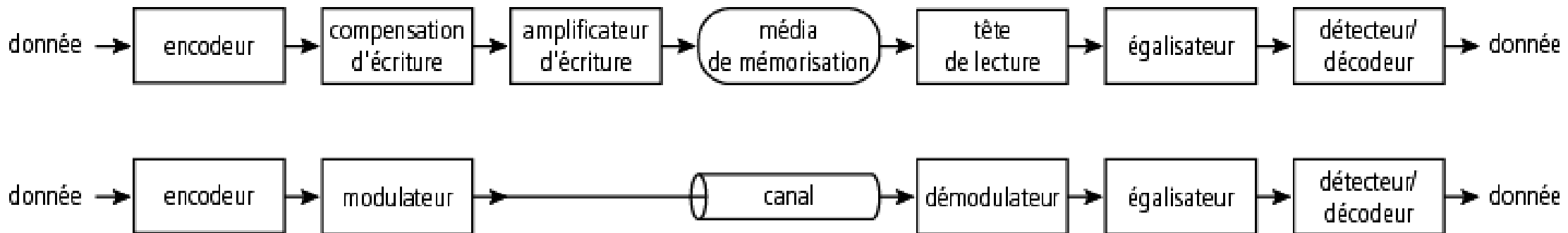
La parité logique

- Une erreur = un bit inversé
- Bit de parité ajouté à l'information
- Domaines privilégiés : la mémoire centrale et l'interface série
- si bit de parité émis \neq bit de parité recalculé sur l'information alors erreur



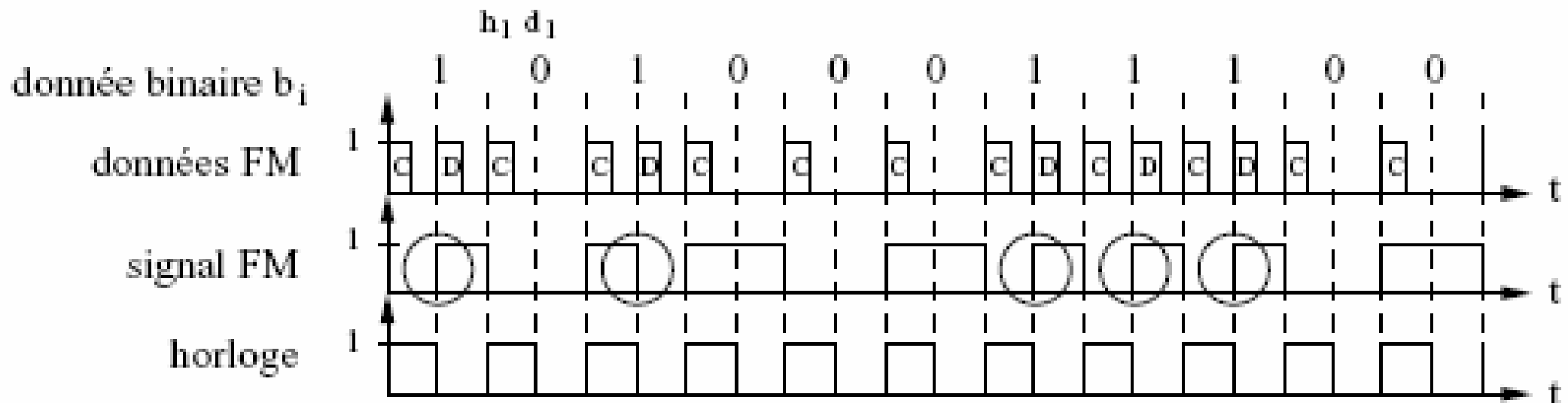
Les codes de canal

- *i.e.* utilisés en mémorisation de données pour maximaliser la densité d'information ou en transmission de données pour maximaliser la bande passante et minimiser le taux d'erreurs



Un exemple dans les unités à disques durs

- Codage FM (*Frequency Modulation*) ou Double Fréquence (DF)
 - une impulsion d'horloge par cellule de donnée
 - une impulsion supplémentaire pour un « 1 »



Conclusion

- Le codage est omniprésent en informatique
- Il permet d'améliorer les performances
 - débit
 - taux d'erreur
 - sécurité
 - échange facilité
 - langage pivot