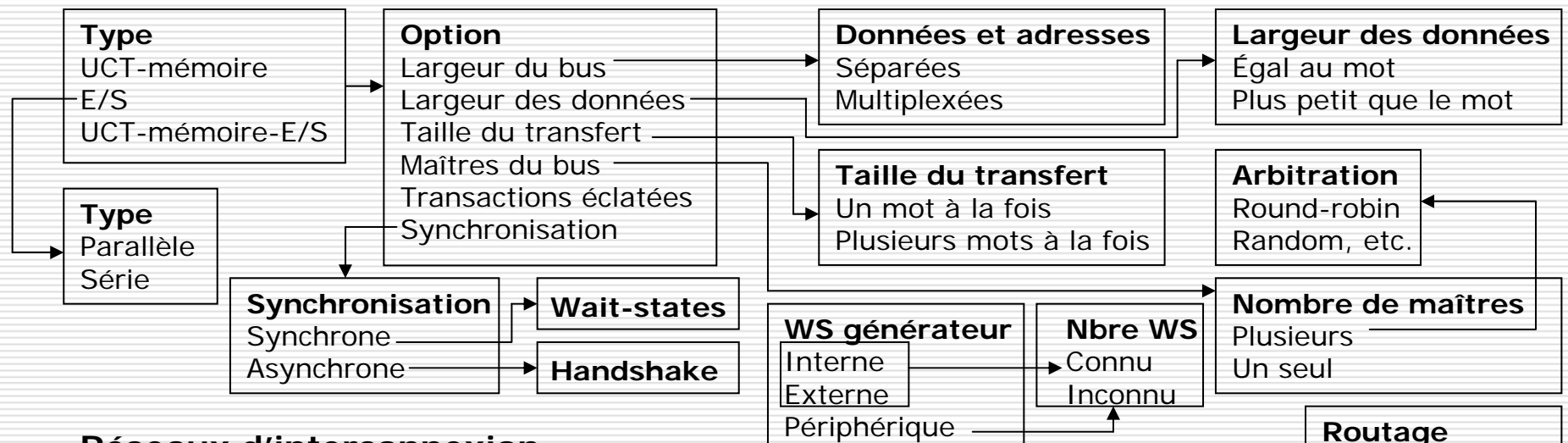


INF6500 :

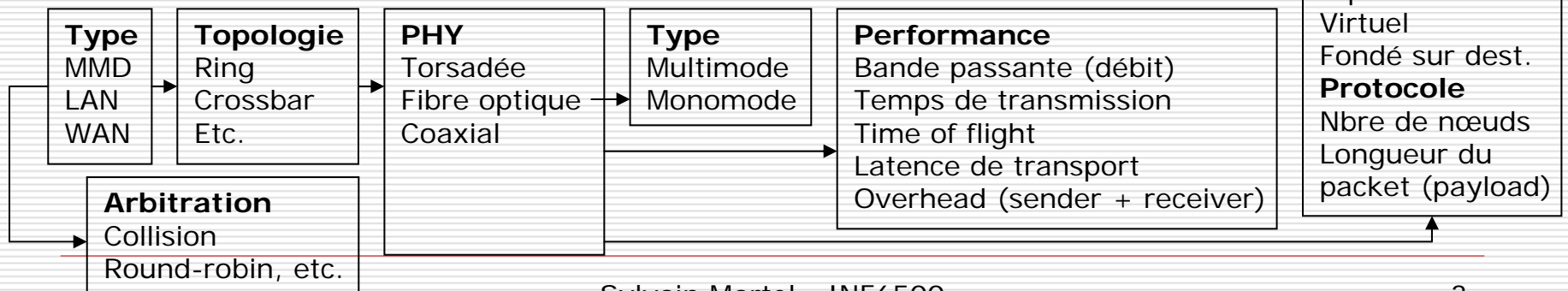
Structures des ordinateurs

Cours 3 : Bus, contrôleurs, et réseaux d'interconnexion

Bus



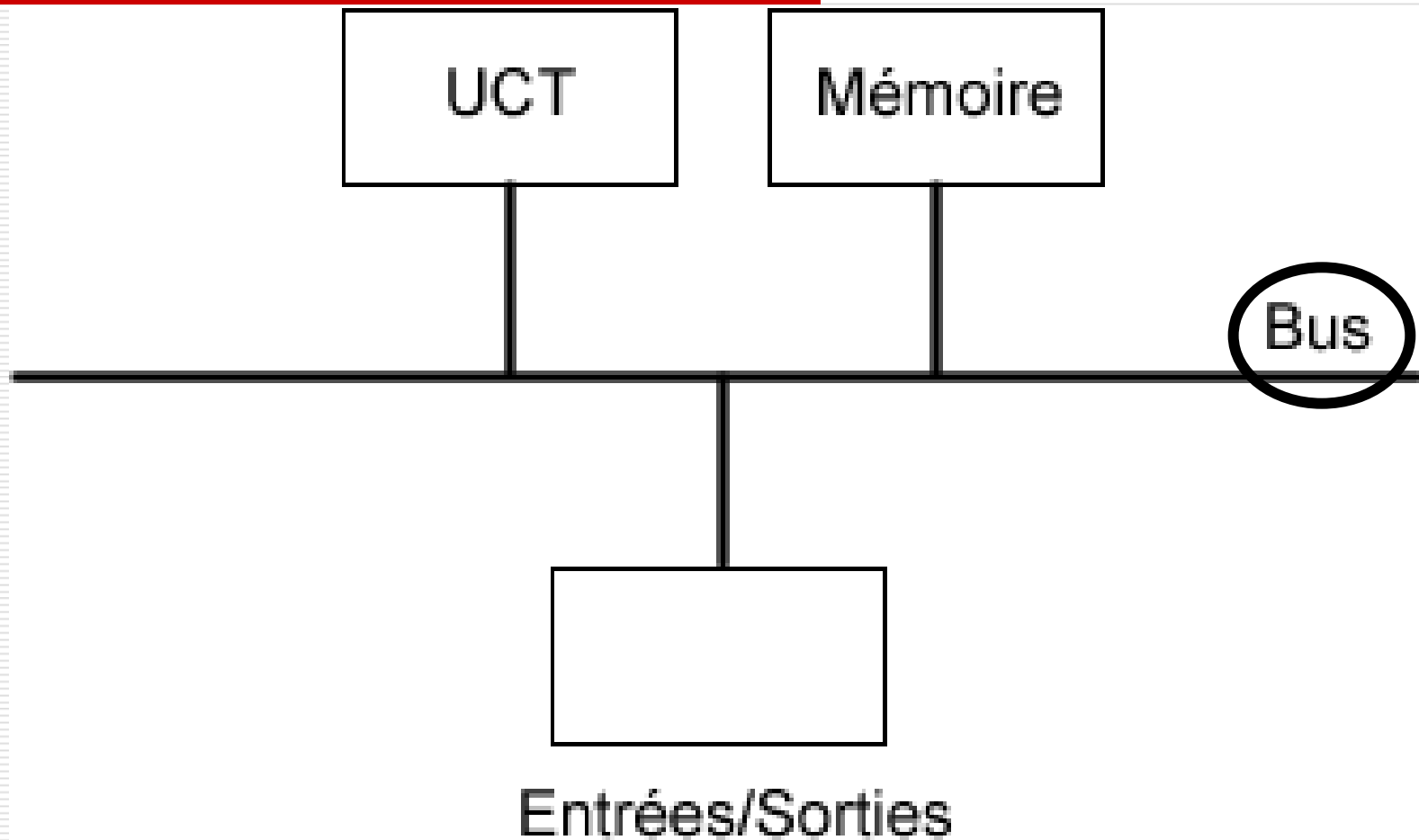
Réseaux d'interconnexion



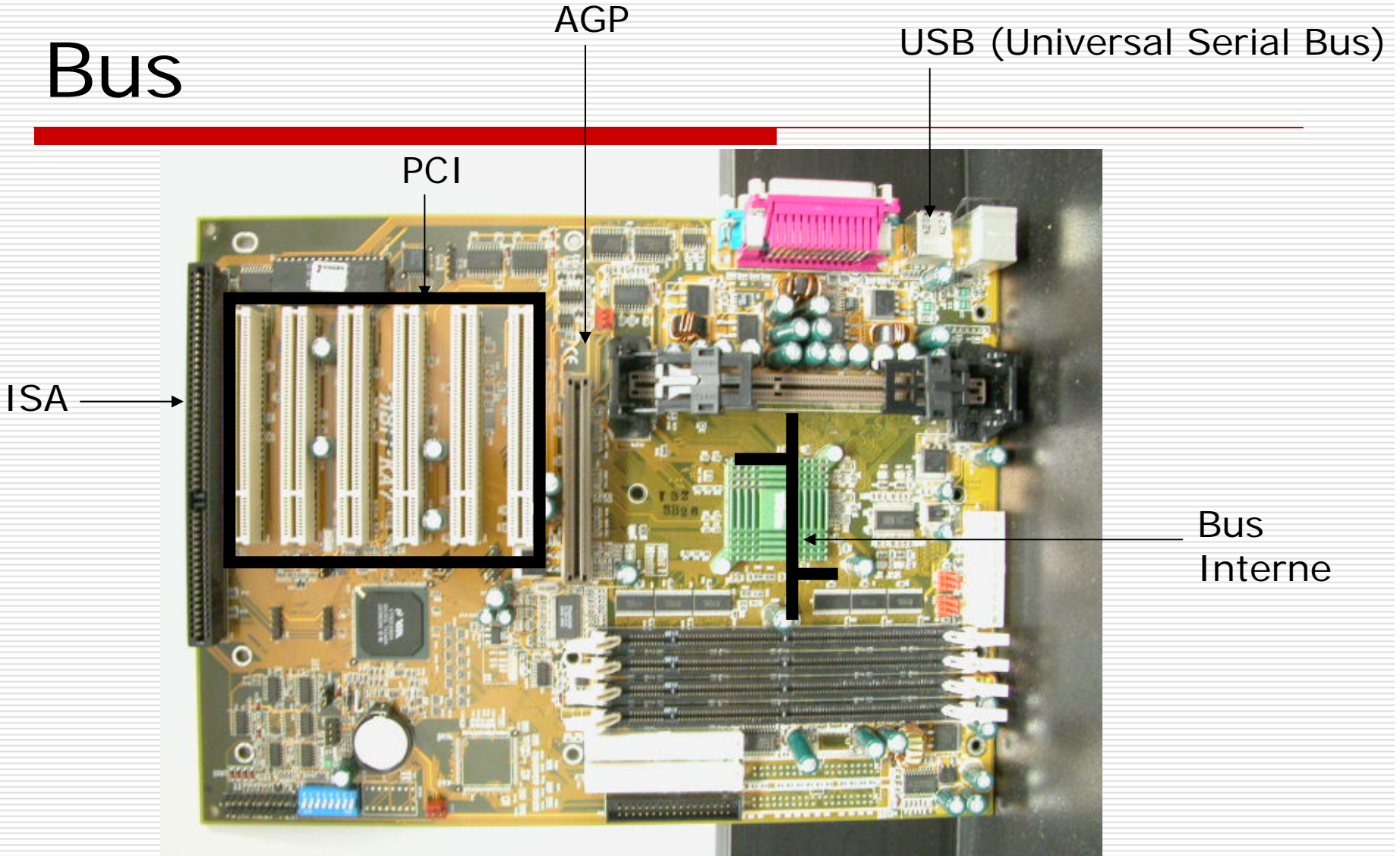
Latence vs. débit

- ❑ Pour temps réels, systèmes de contrôle de haute performance, courte latence est désirable, haut débit désirable
- ❑ Internet: Longue latence OK (unidirectionnelle), haut débit désirable

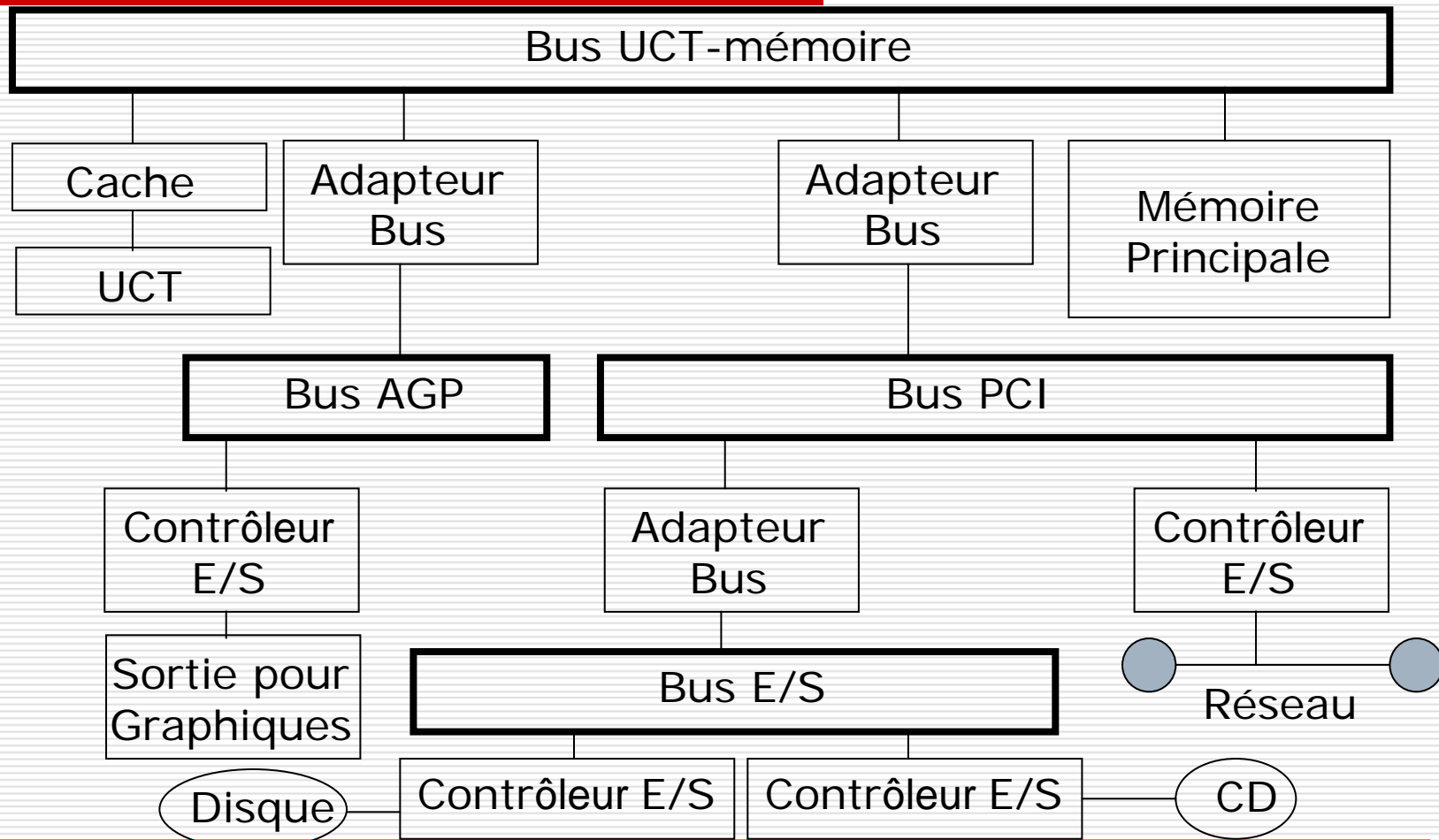
L'ordinateur moderne: Bus



Bus

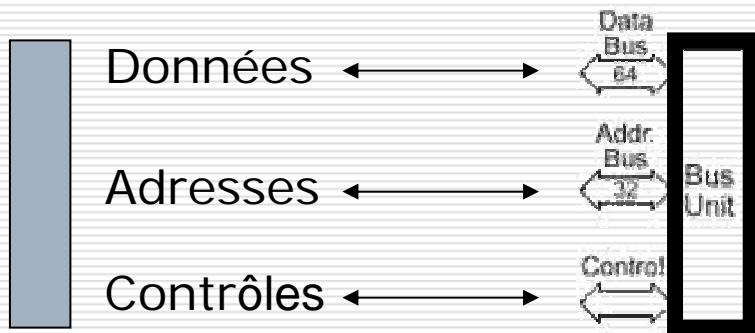


Bus (Exemple)



Pentium

BUS



Interface pour le bus

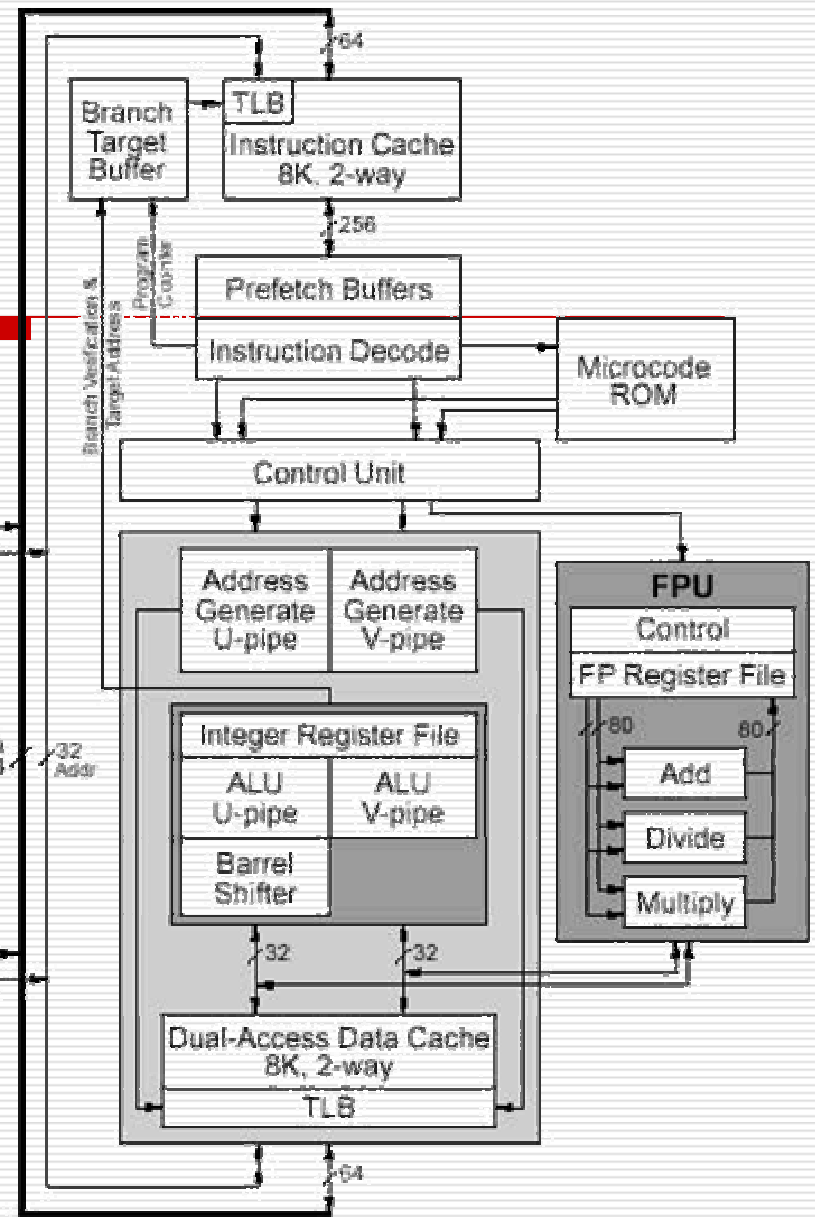
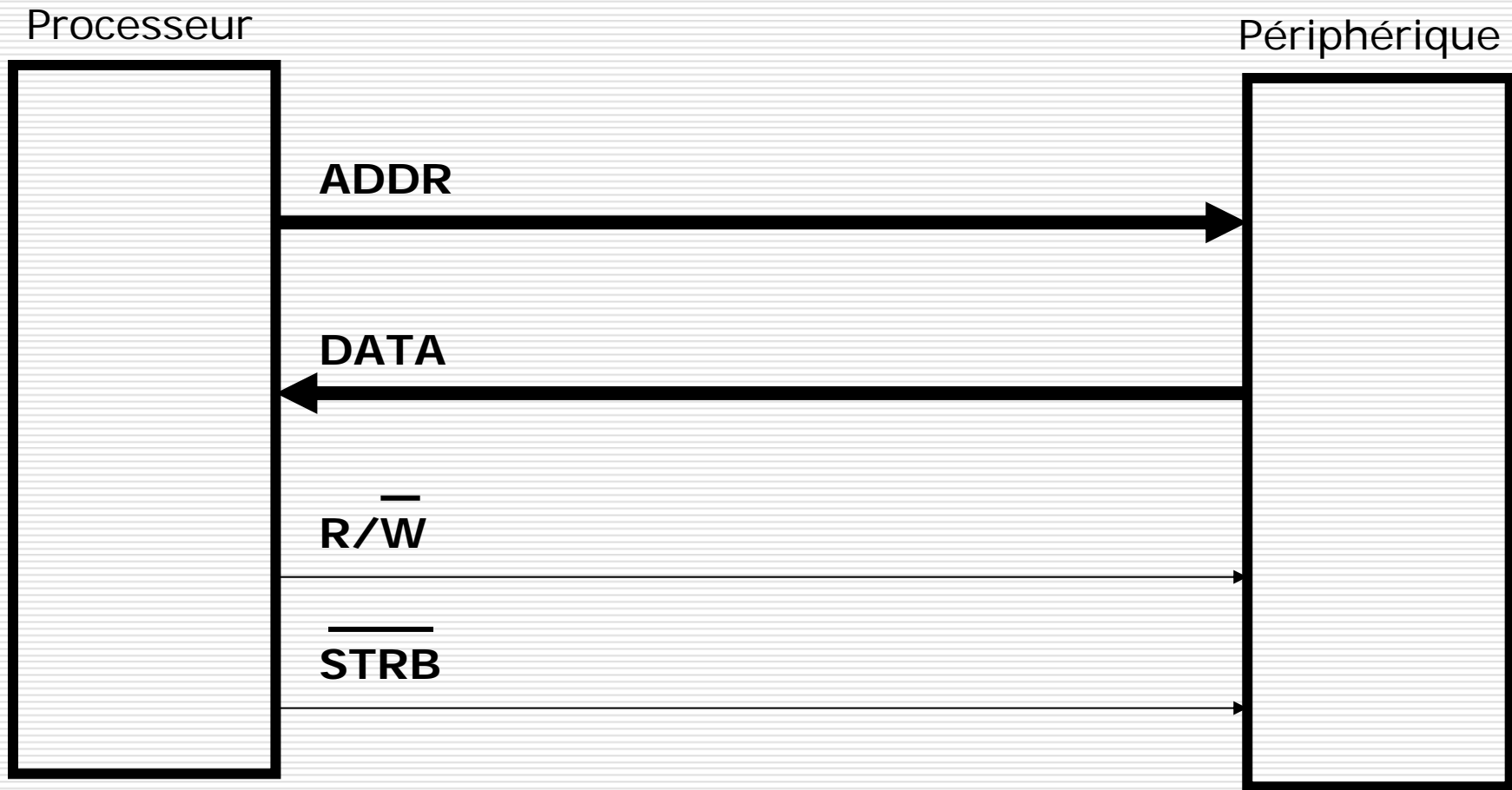
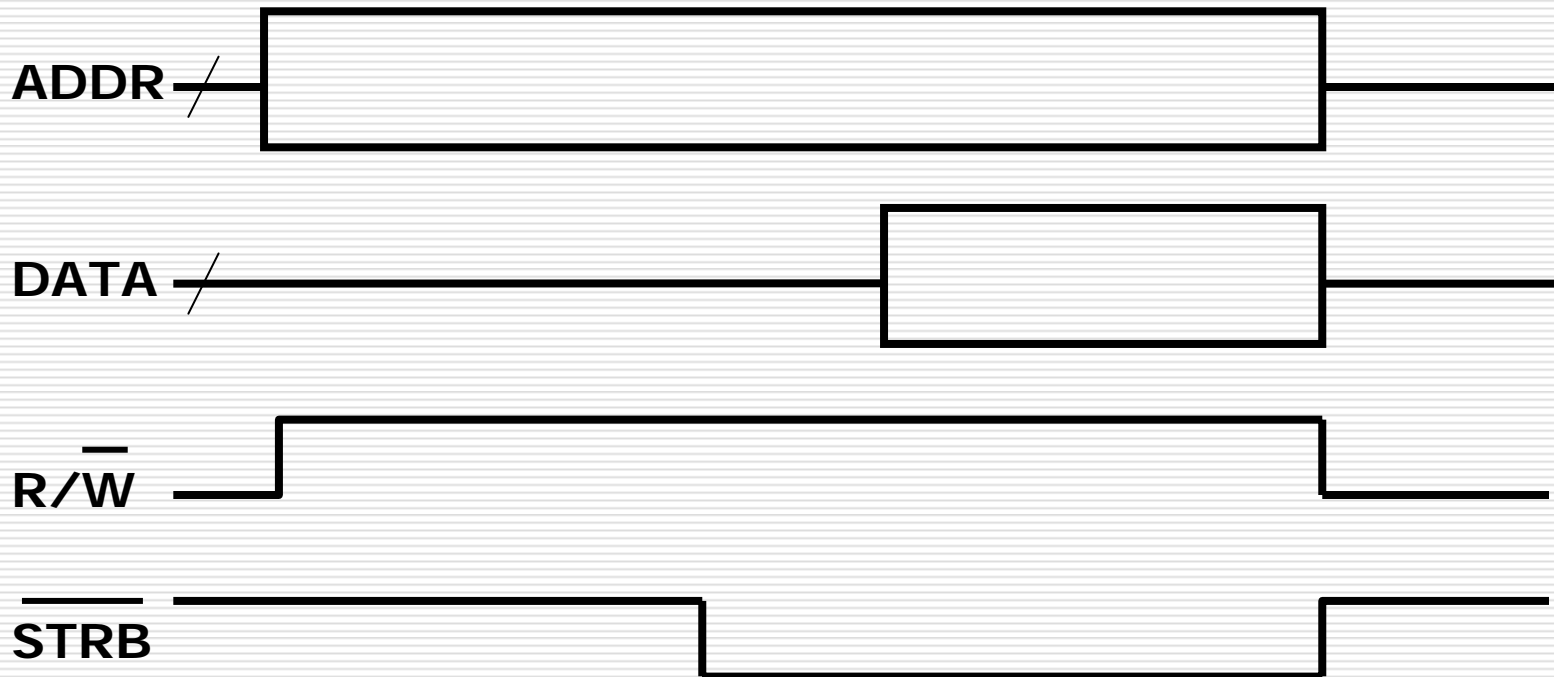


Figure 1. Pentium block diagram.

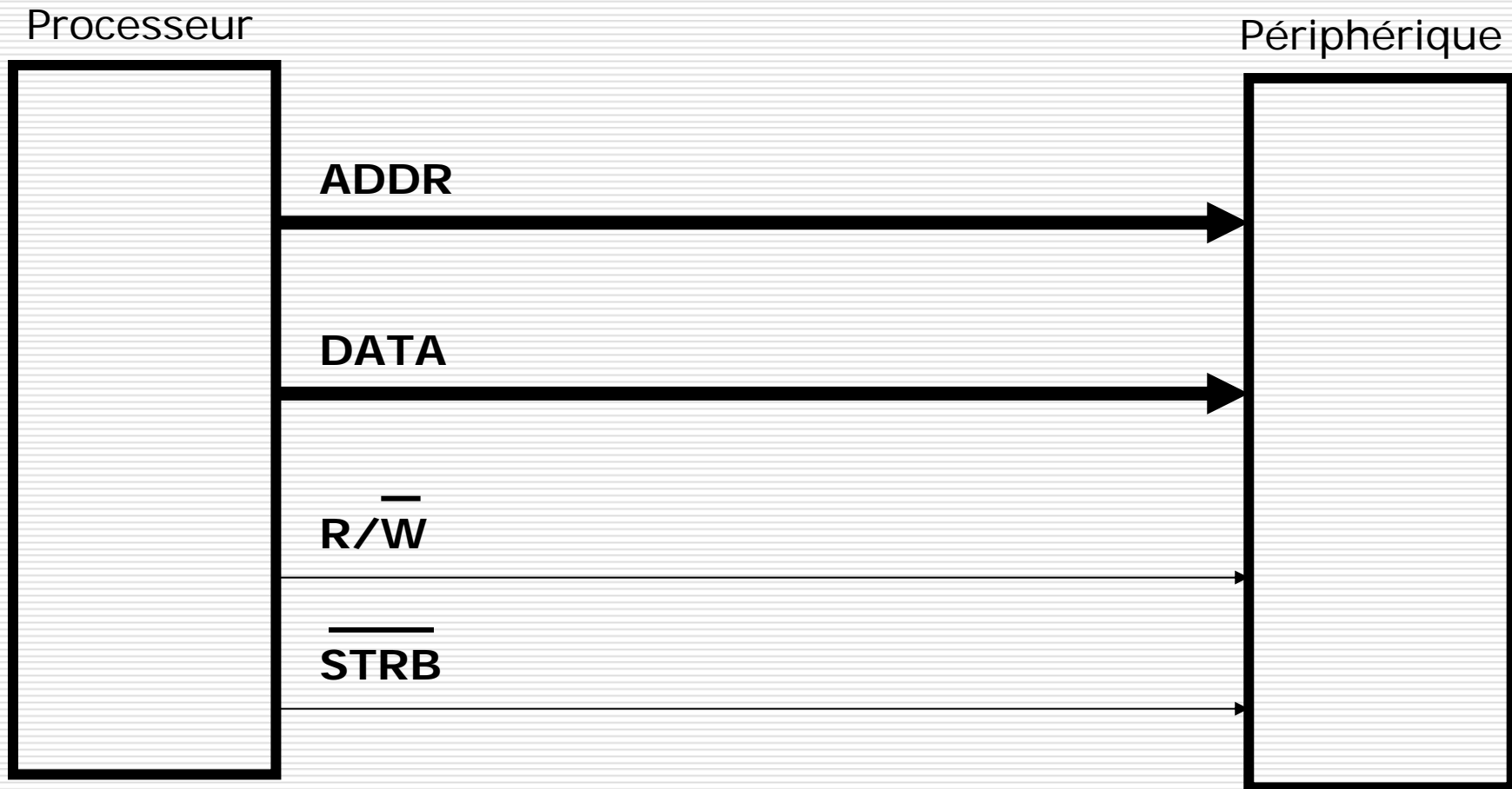
Lecture standard sur le bus



Lecture standard sur le bus



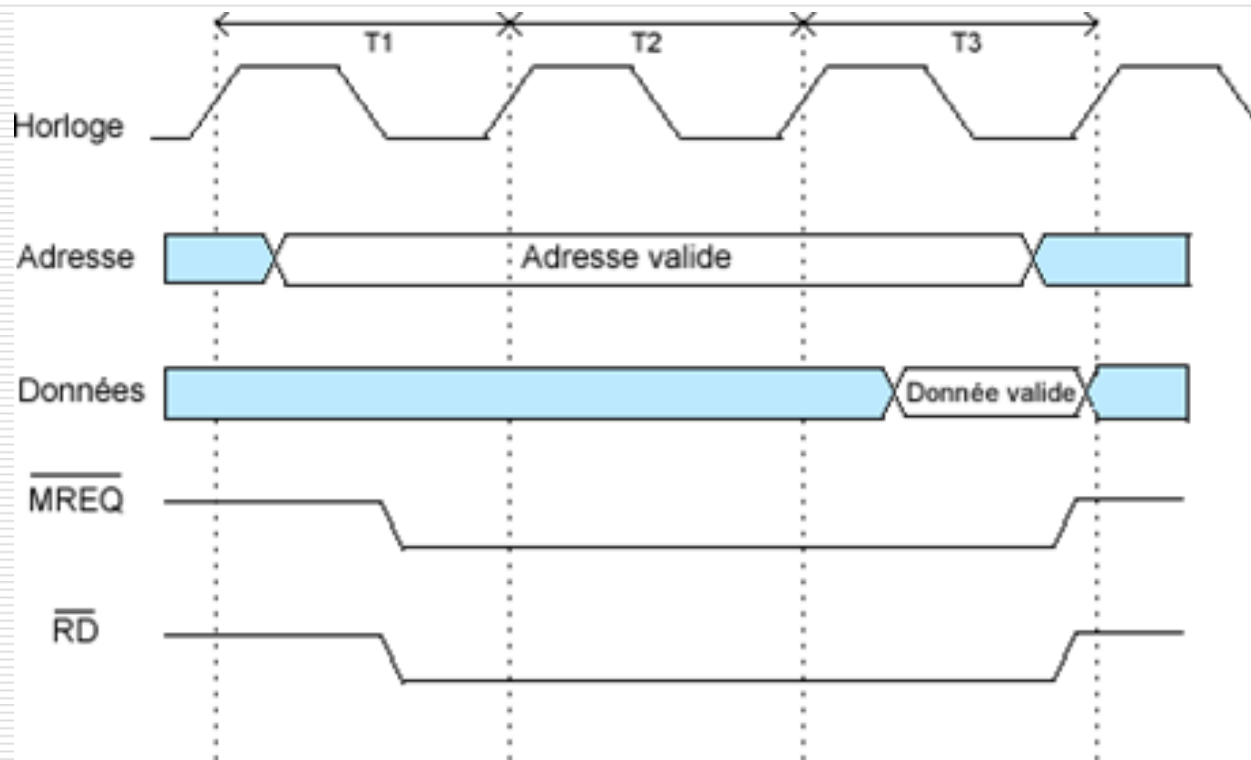
Écriture standard sur le bus



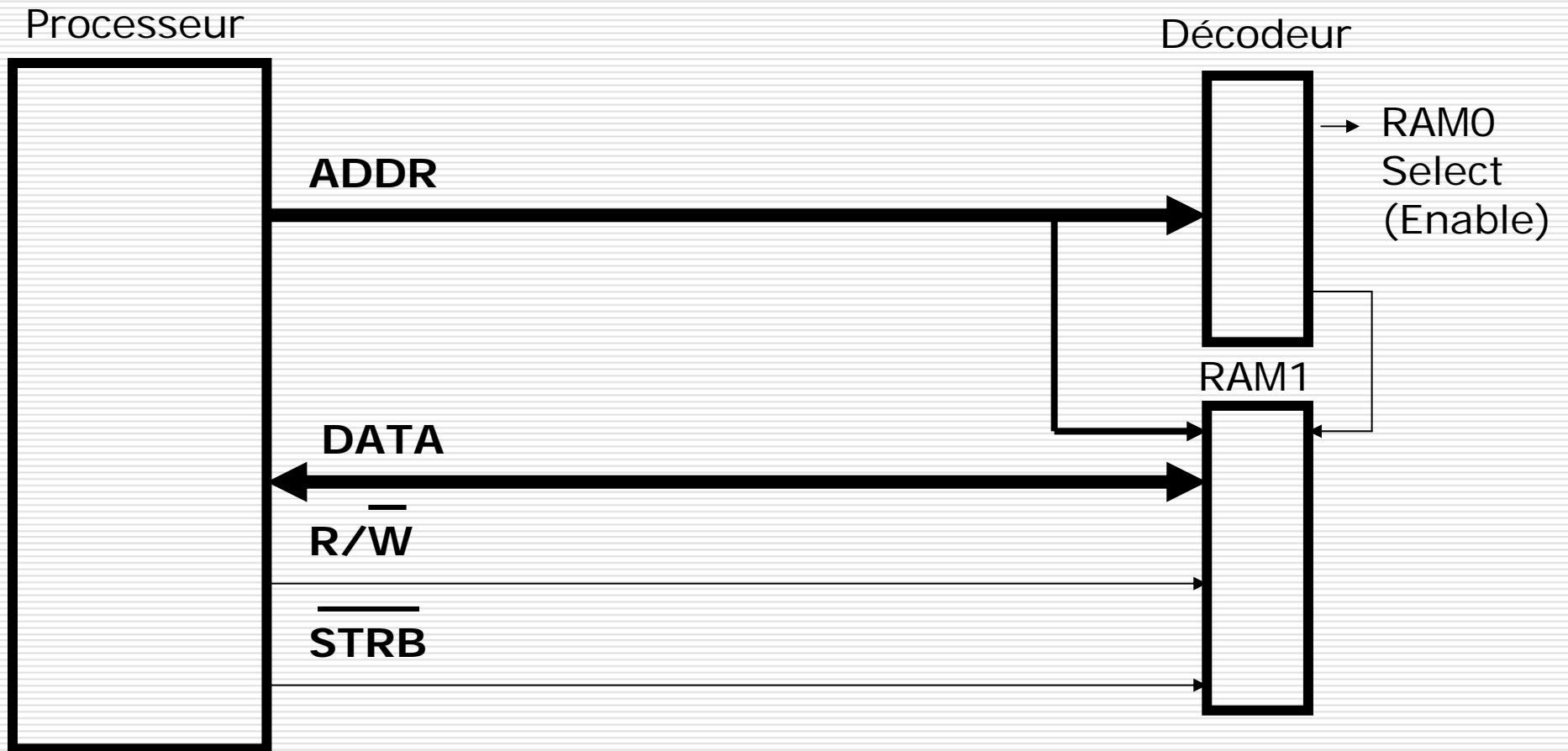
Écriture standard sur le bus



Bus



Déterminer la destination



Principales options pour un bus

Option	Haute Performance	Moins Dispendieux
Largeur du bus	Lignes de données et d'adresses séparées	Lignes de données et d'adresses multiplexées
Largeur des données	Plus large est plus rapide (ex. 64 bits)	Moins large est moins dispendieux
Taille du transfert	Plusieurs mots ont moins d' « overhead »	Transfert d'un mot à la fois est plus simple

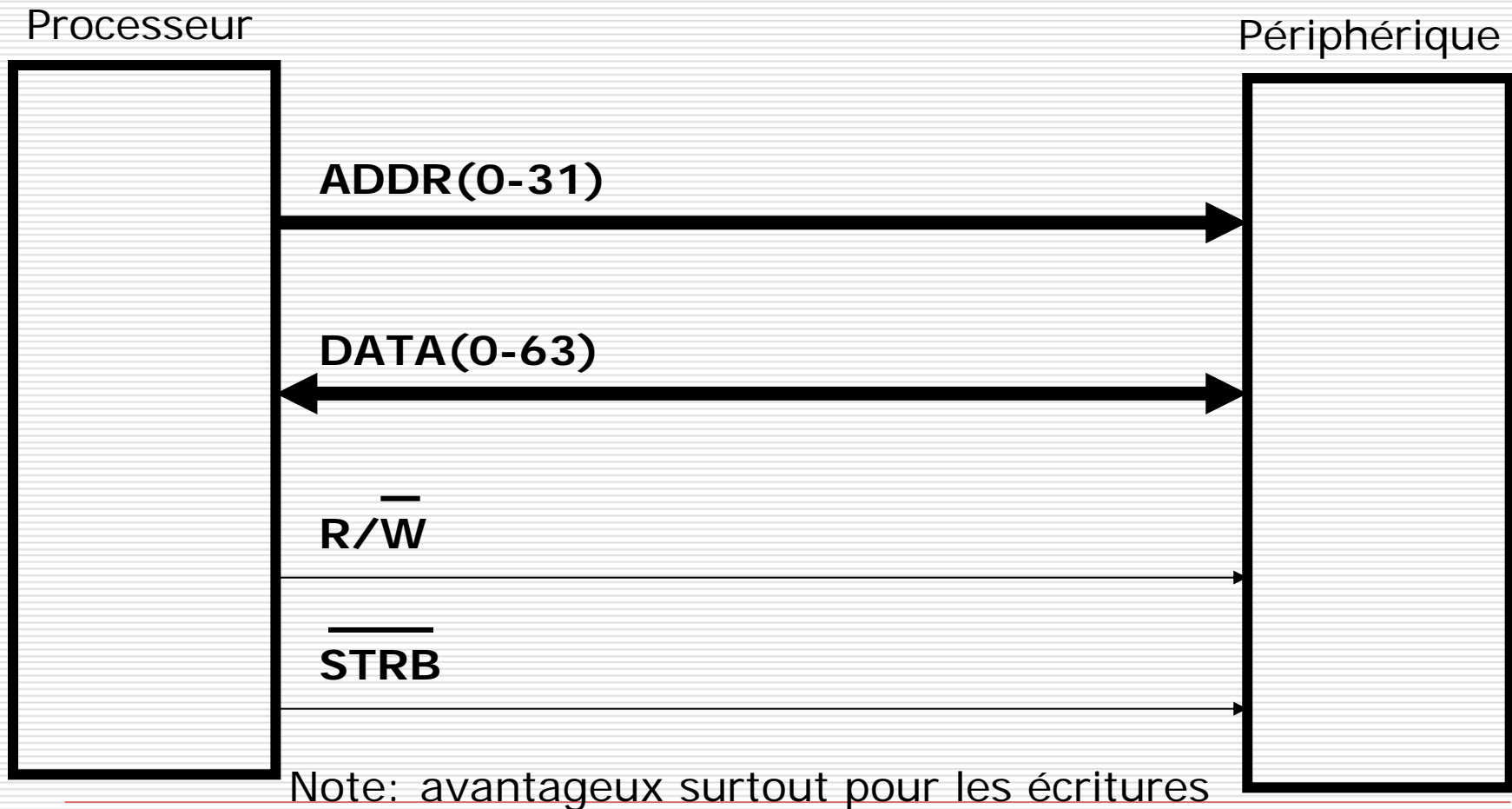
Principales options pour un bus (suite)

Option	Haute Performance	Moins Dispendieux
Maîtres du bus	Plusieurs (requiert arbitration)	Un seul maître (pas d'arbitration)
Transactions éclatées (split transactions)	Oui – paquets de requête and réponse (reply) séparés pour une bande passante plus grande (plusieurs maîtres)	Non – connection continue est moins dispendieuse et a moins de latence
Horloge de synchronization (Clocking)	Synchrone	Asynchrone

Principales options pour un bus

Option	Haute Performance	Moins Dispendieux
Largeur du bus	<u>Lignes de données et d'adresses séparées</u>	Lignes de données et d'adresses multiplexées
Largeur des données	Plus large est plus rapide (ex. 64 bits)	Moins large est moins dispendieux
Taille du transfert	Plusieurs mots ont moins d' « overhead »	Transfert d'un mot à la fois est plus simple

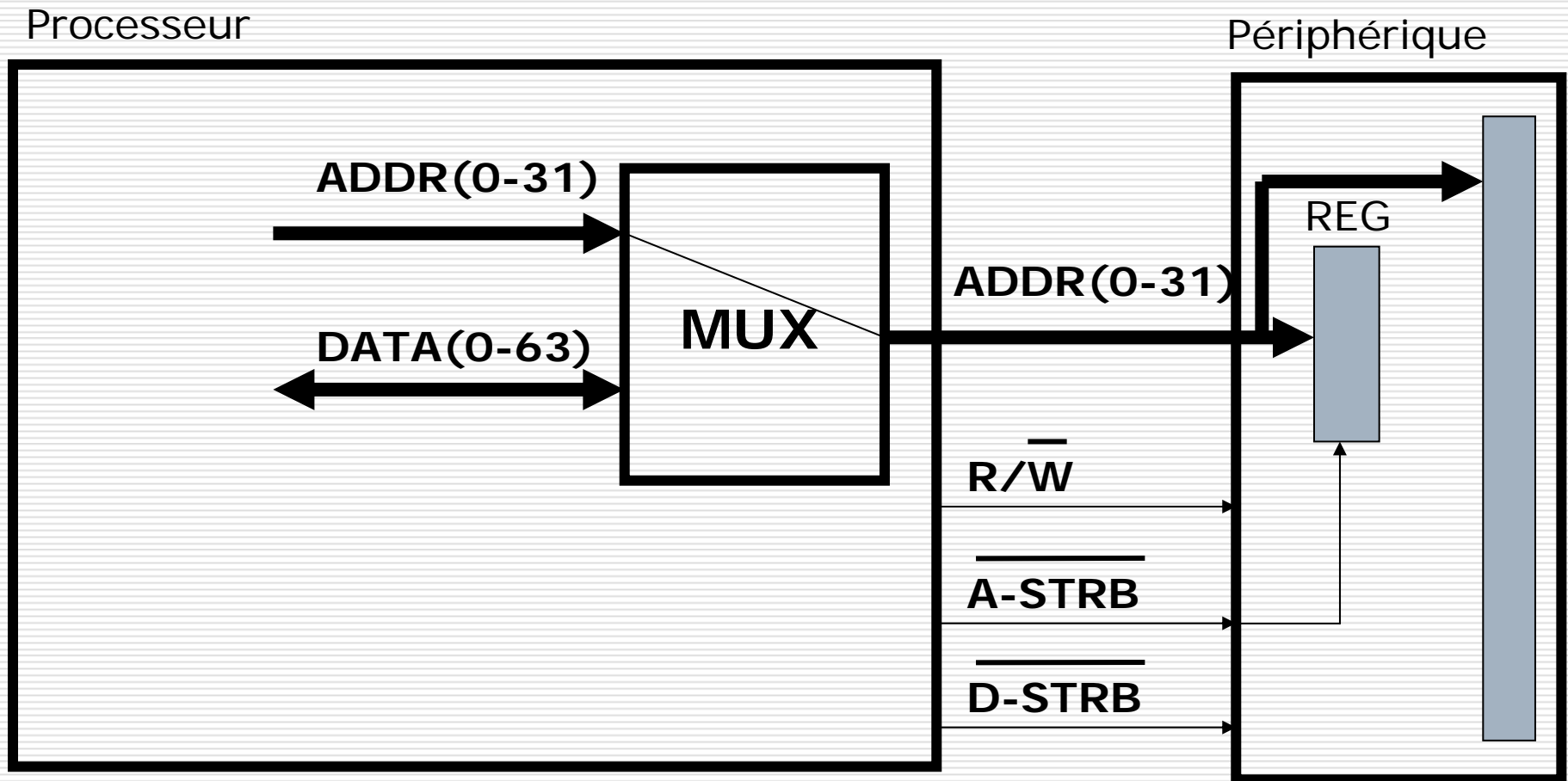
Lignes de données et d'adresses séparées



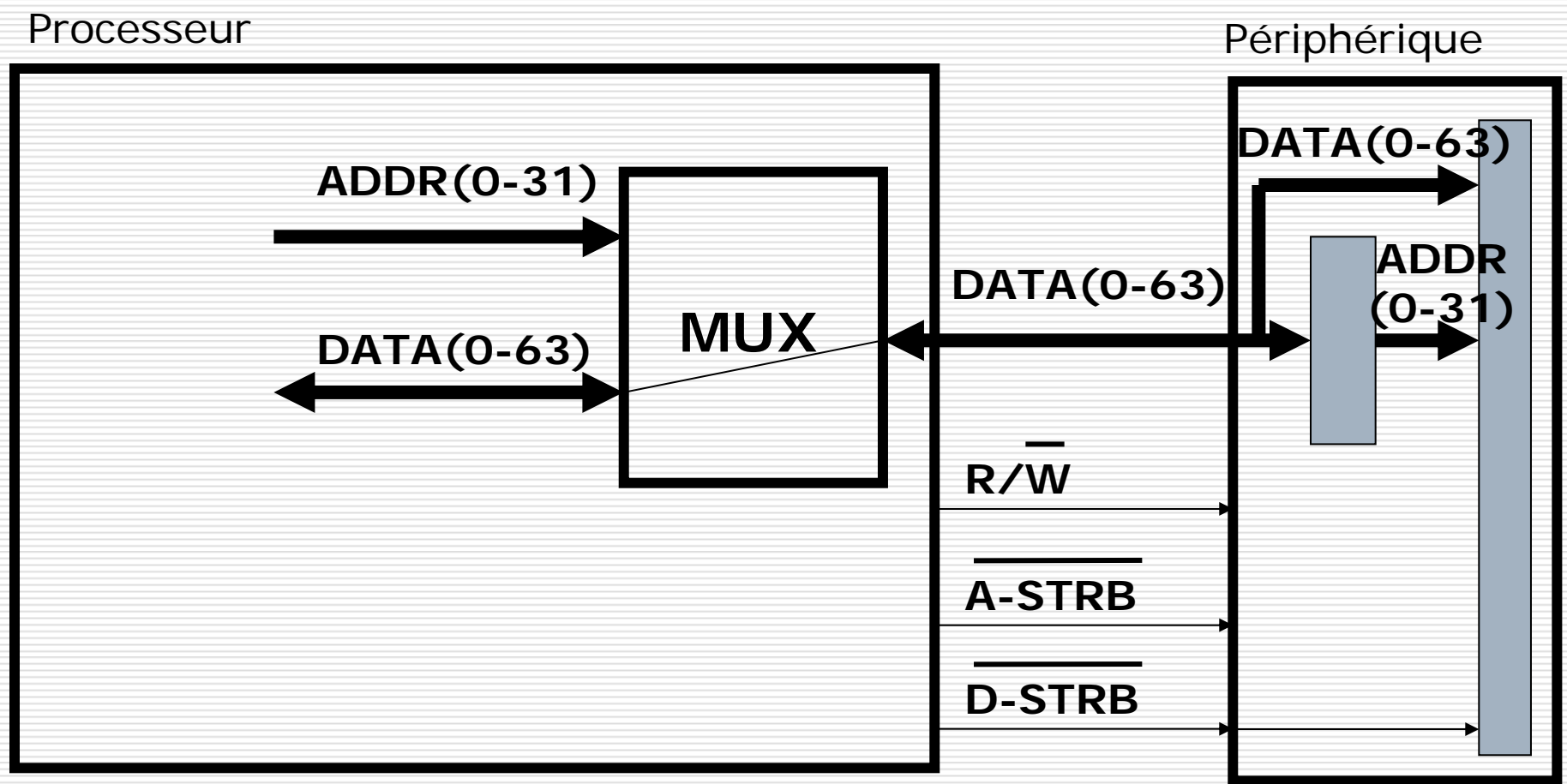
Principales options pour un bus

Option	Haute Performance	Moins Dispendieux
Largeur du bus	Lignes de données et d'adresses séparées	<u>Lignes de données et d'adresses multiplexées</u>
Largeur des données	Plus large est plus rapide (ex. 64 bits)	Moins large est moins dispendieux
Taille du transfert	Plusieurs mots ont moins d' « overhead »	Transfert d'un mot à la fois est plus simple

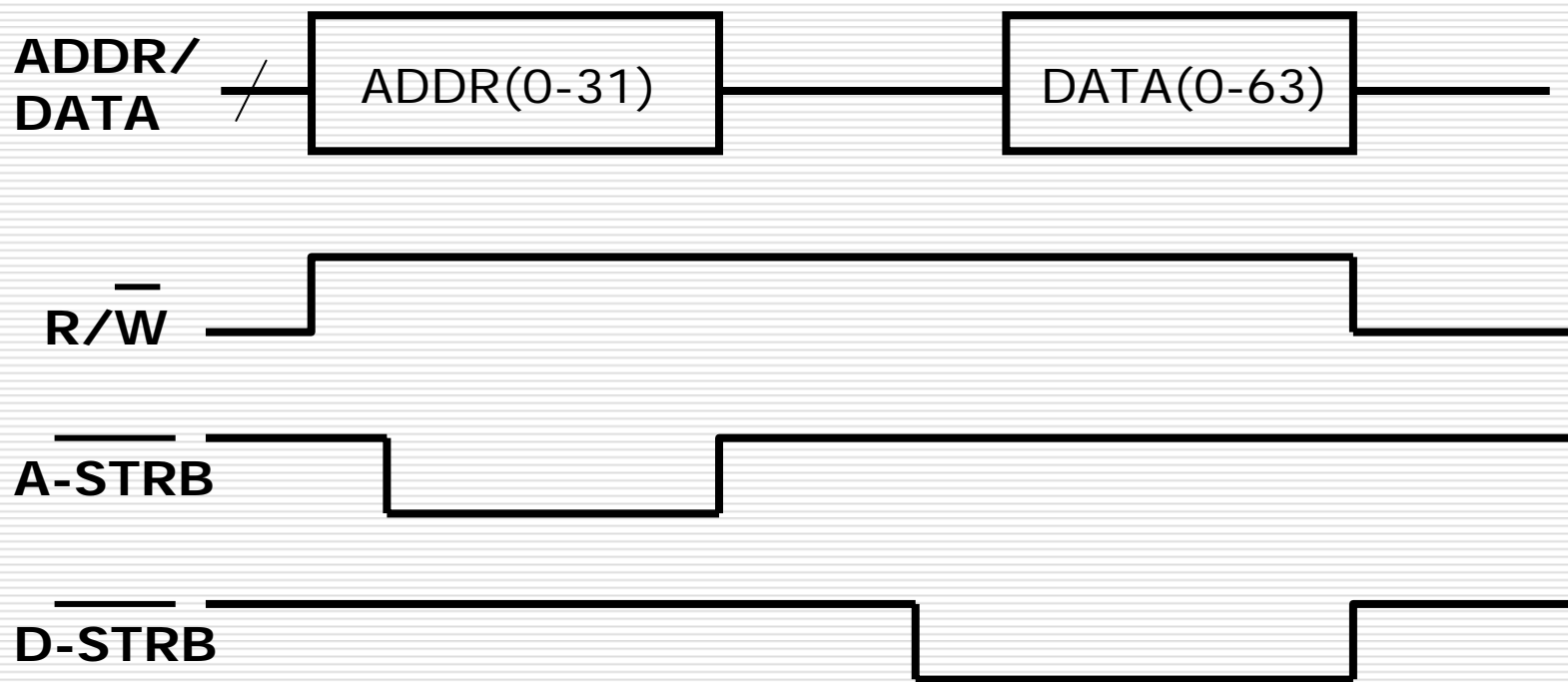
Lignes de données et d'adresses multiplexées (T1)



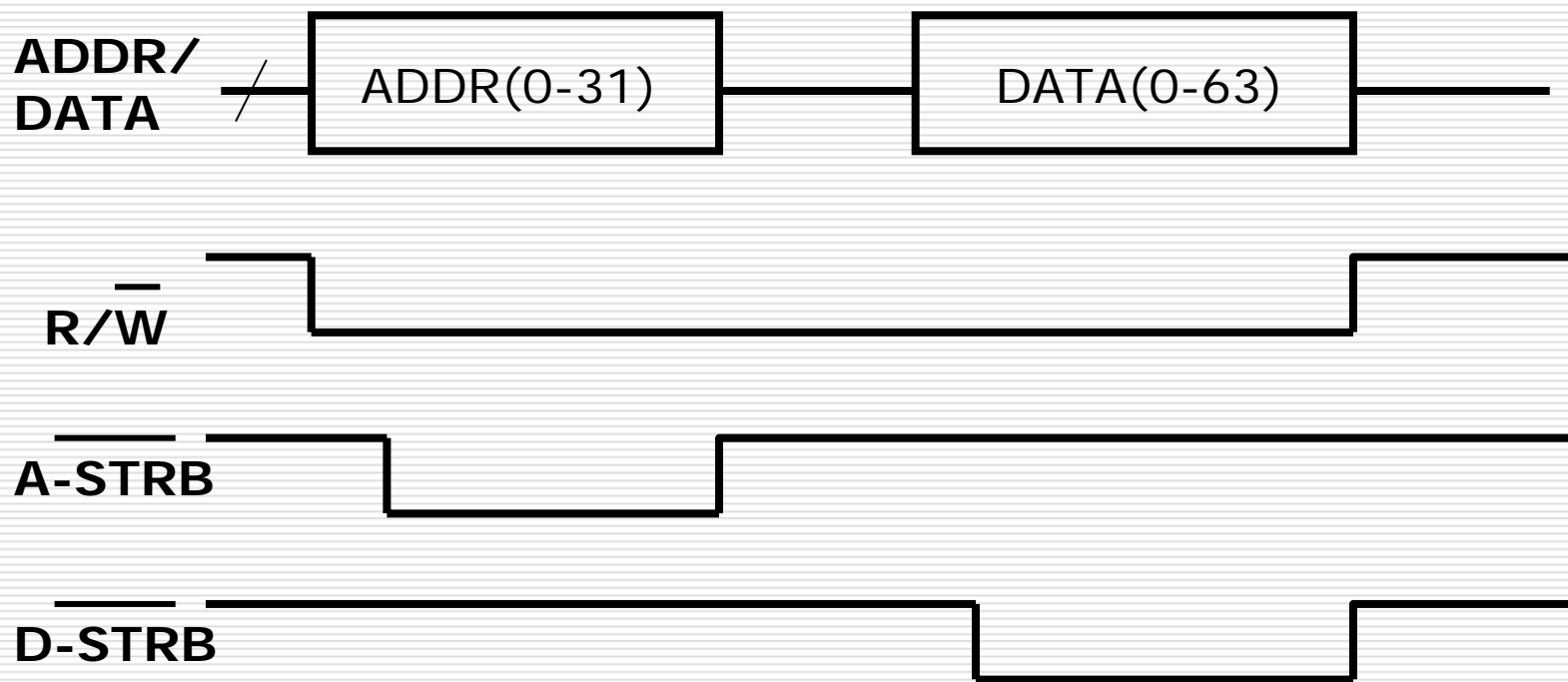
Lignes de données et d'adresses multiplexées (T2)



Lignes de données et d'adresses multiplexées - Lecture sur le bus



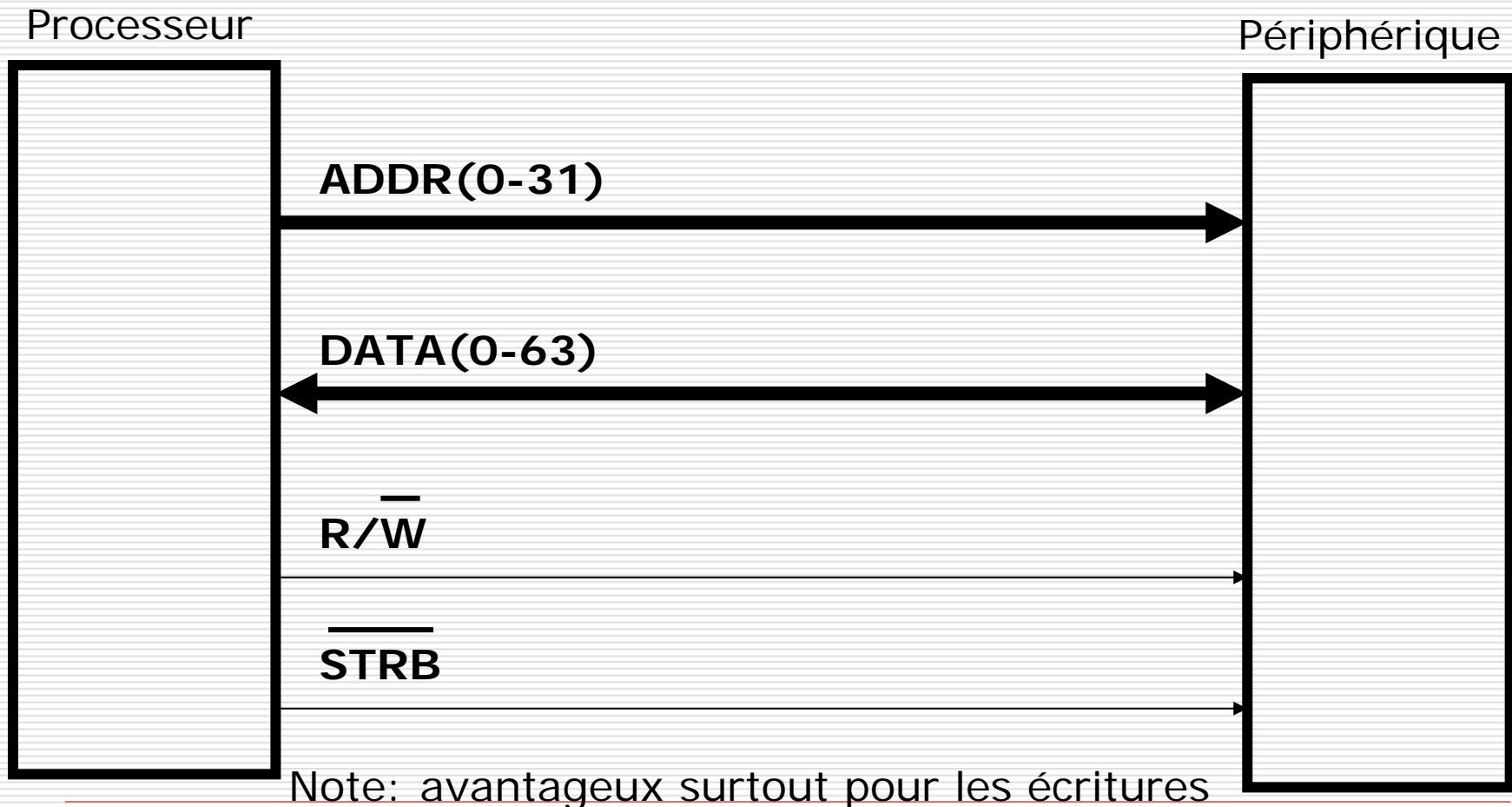
Lignes de données et d'adresses multiplexées - Écriture sur le bus



Principales options pour un bus

Option	Haute Performance	Moins Dispendieux
Largeur du bus	Lignes de données et d'adresses séparées	Lignes de données et d'adresses multiplexées
Largeur des données	<u>Plus large est plus rapide (ex. 64 bits)</u>	Moins large est moins dispendieux
Taille du transfert	Plusieurs mots ont moins d' « overhead »	Transfert d'un mot à la fois est plus simple

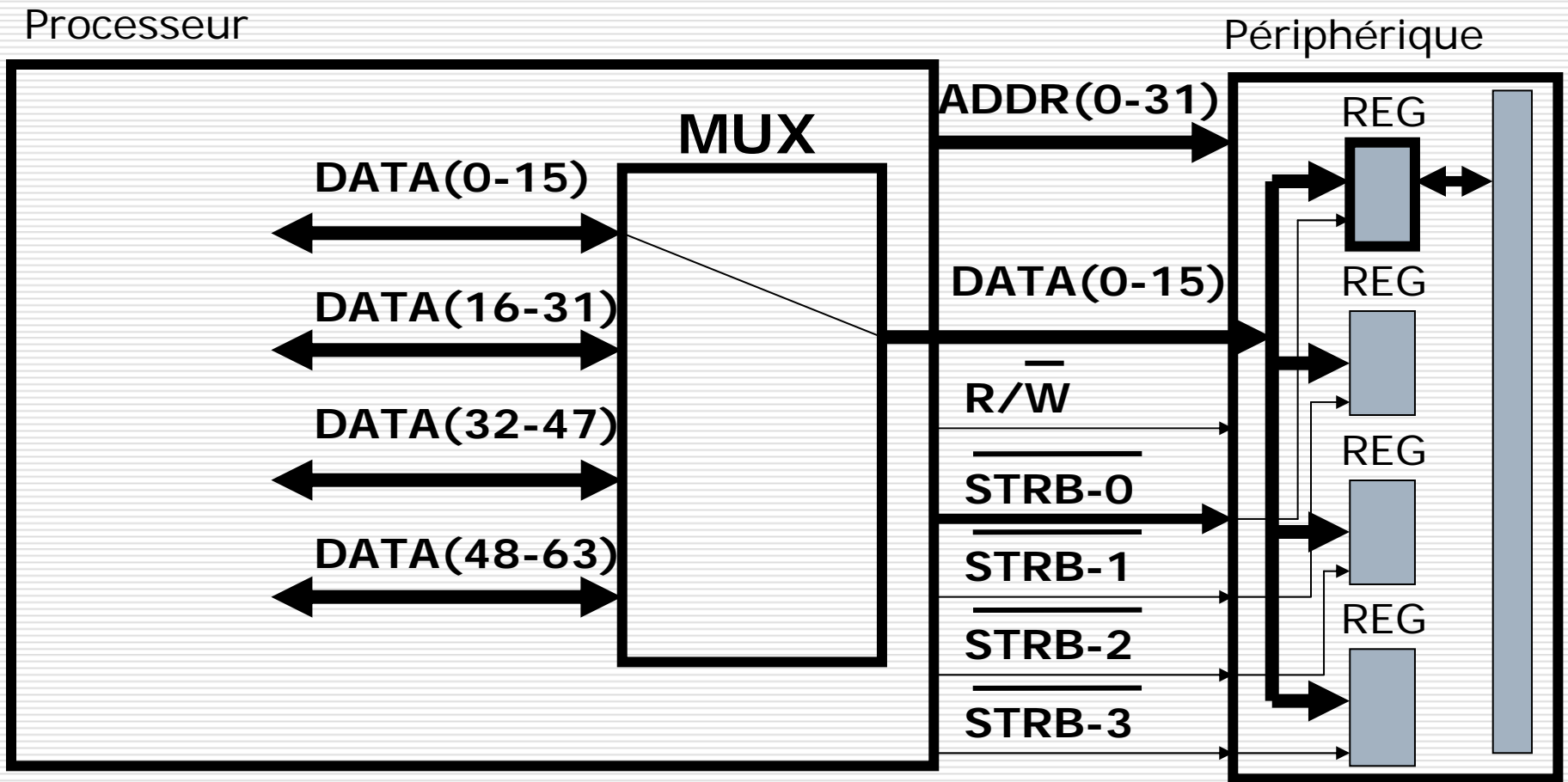
Lignes de données et d'adresses séparées – Données plus larges



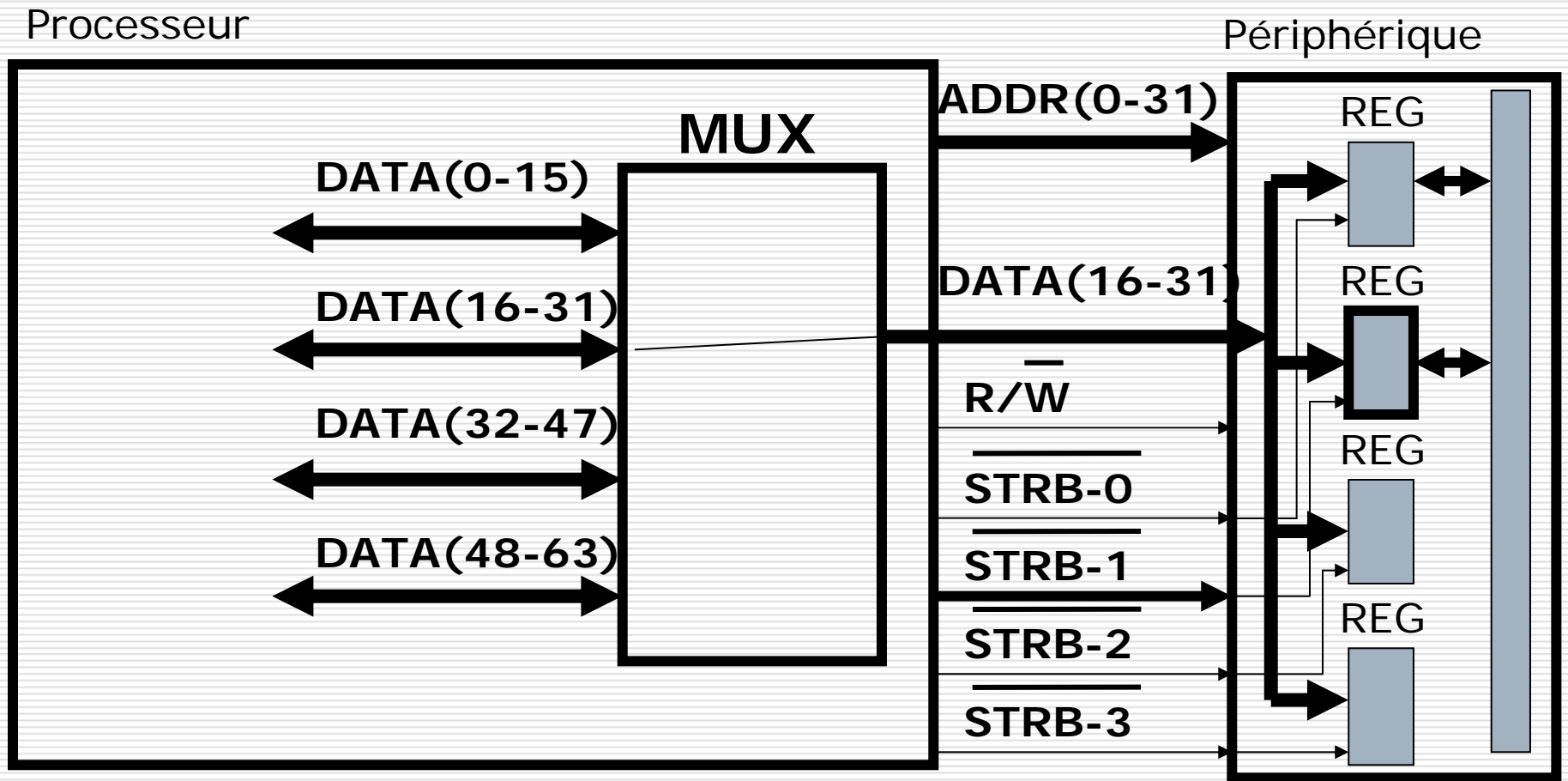
Principales options pour un bus

Option	Haute Performance	Moins Dispendieux
Largeur du bus	Lignes de données et d'adresses séparées	Lignes de données et d'adresses multiplexées
Largeur des données	Plus large est plus rapide (ex. 64 bits)	<u>Moins large est moins dispendieux</u>
Taille du transfert	Plusieurs mots ont moins d' « overhead »	Transfert d'un mot à la fois est plus simple

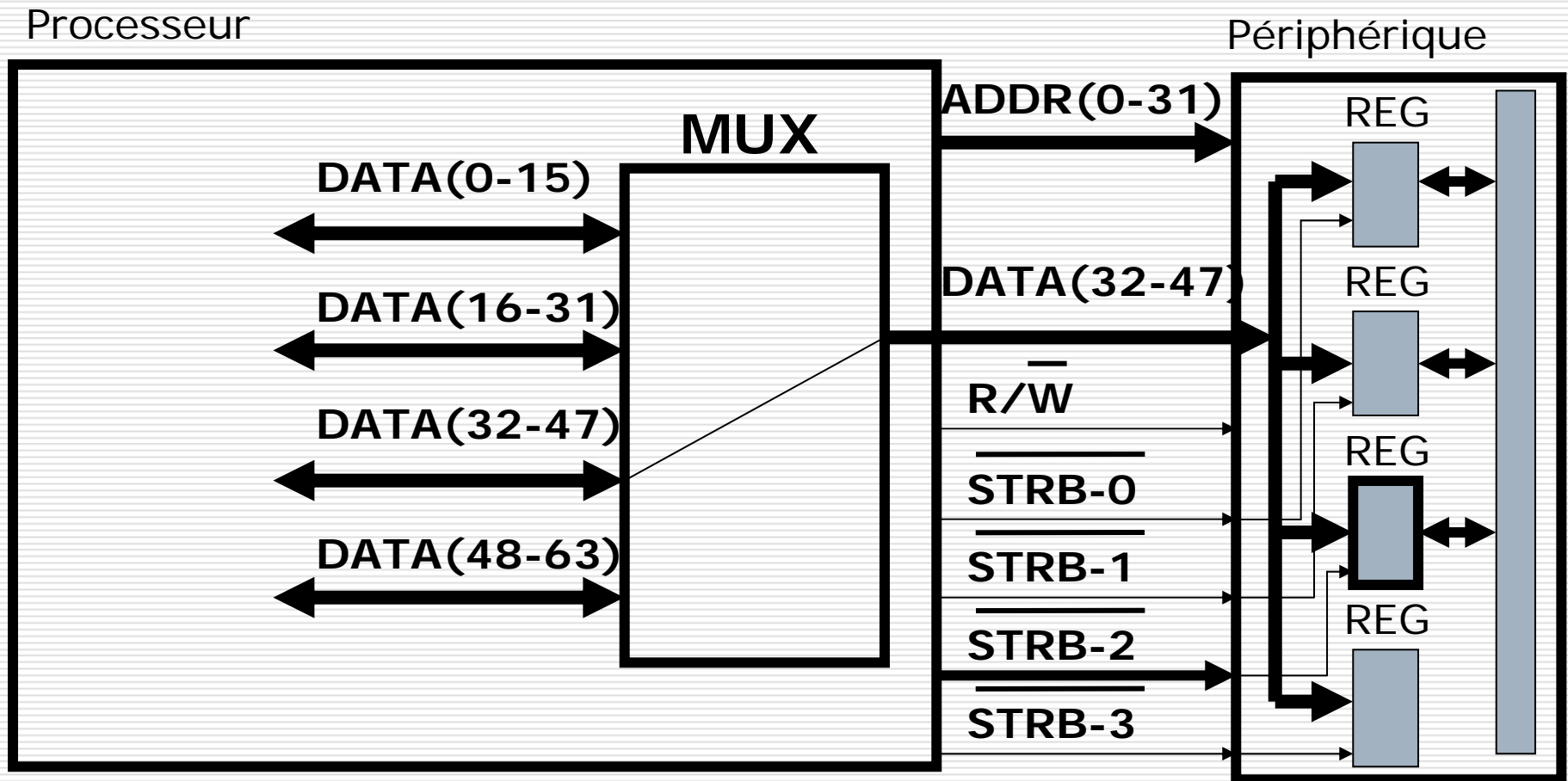
Lignes de données et d'adresses séparées – Données moins larges (T1)



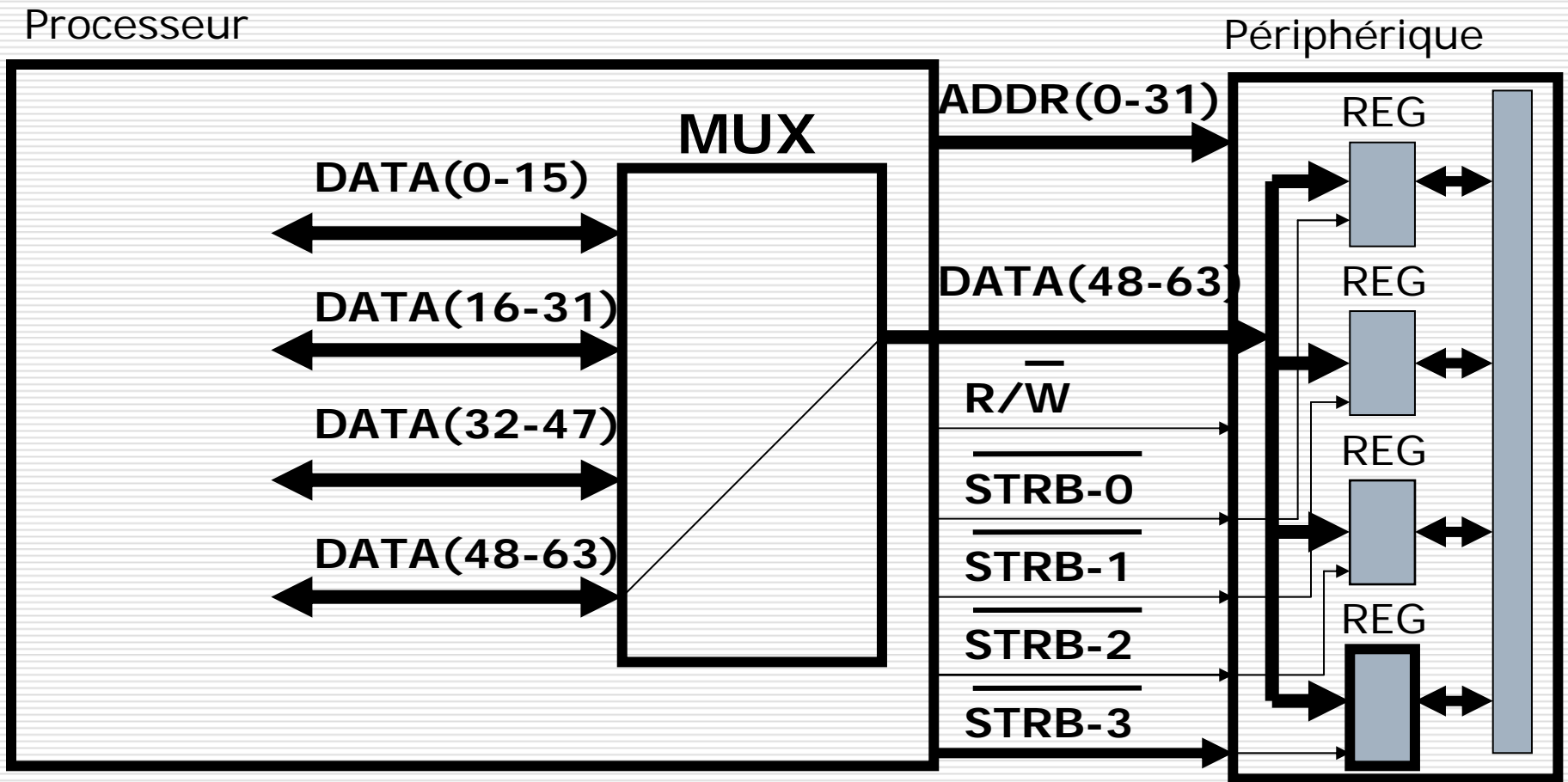
Lignes de données et d'adresses séparées – Données moins larges (T2)



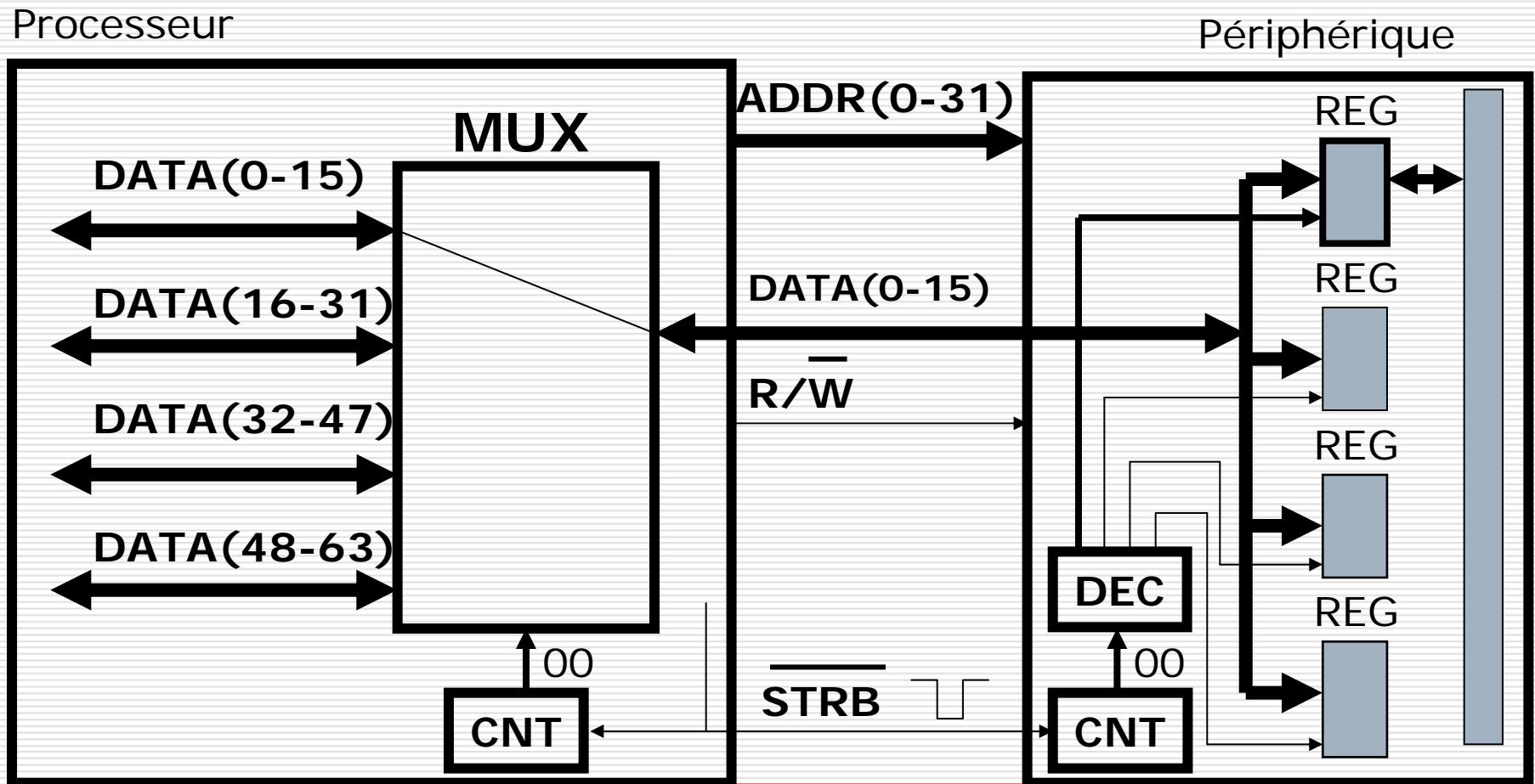
Lignes de données et d'adresses séparées – Données moins larges (T3)



Lignes de données et d'adresses séparées – Données moins larges (T4)

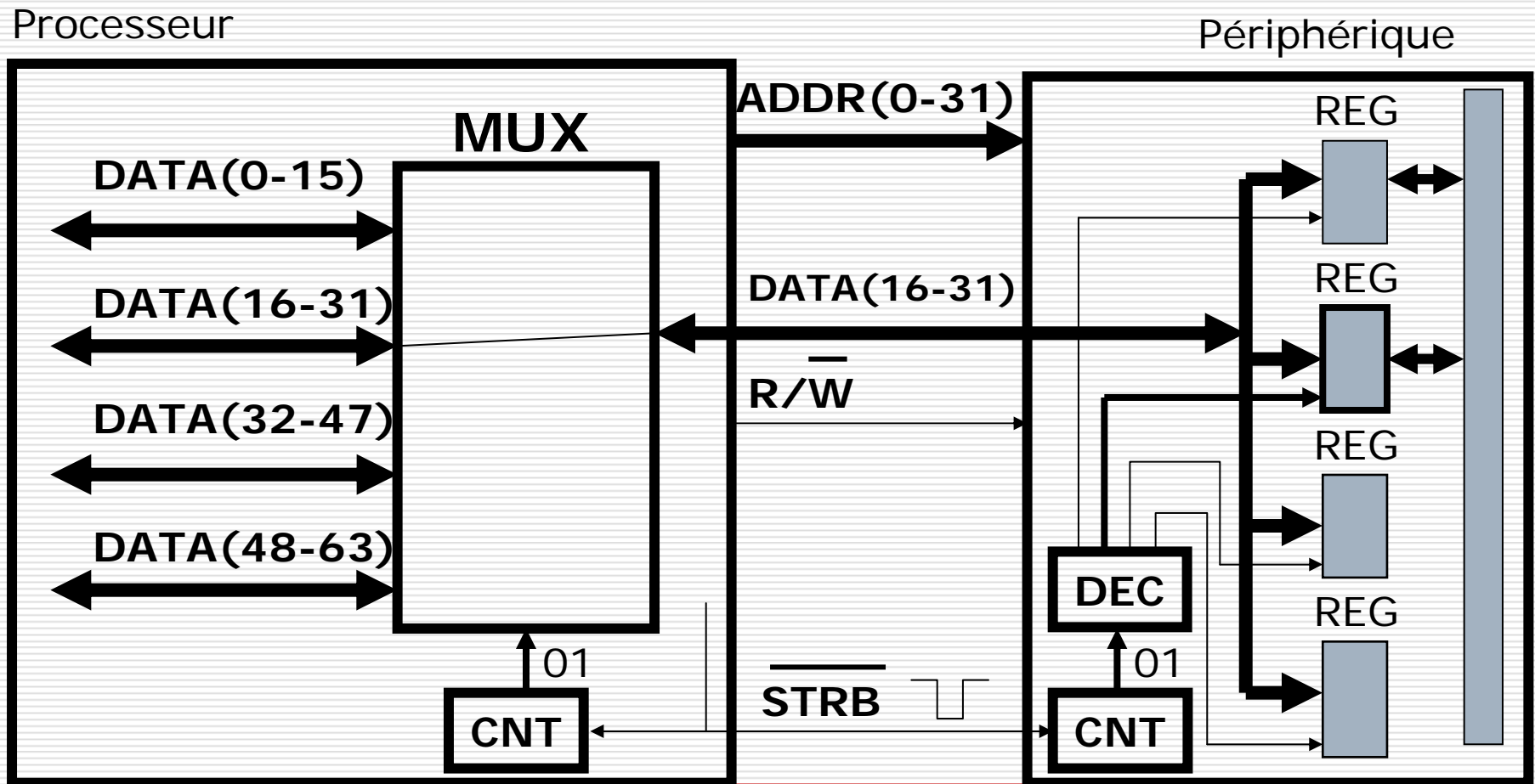


Lignes de données et d'adresses séparées – Données moins larges (réduire le nombre de lignes STRB) – T1



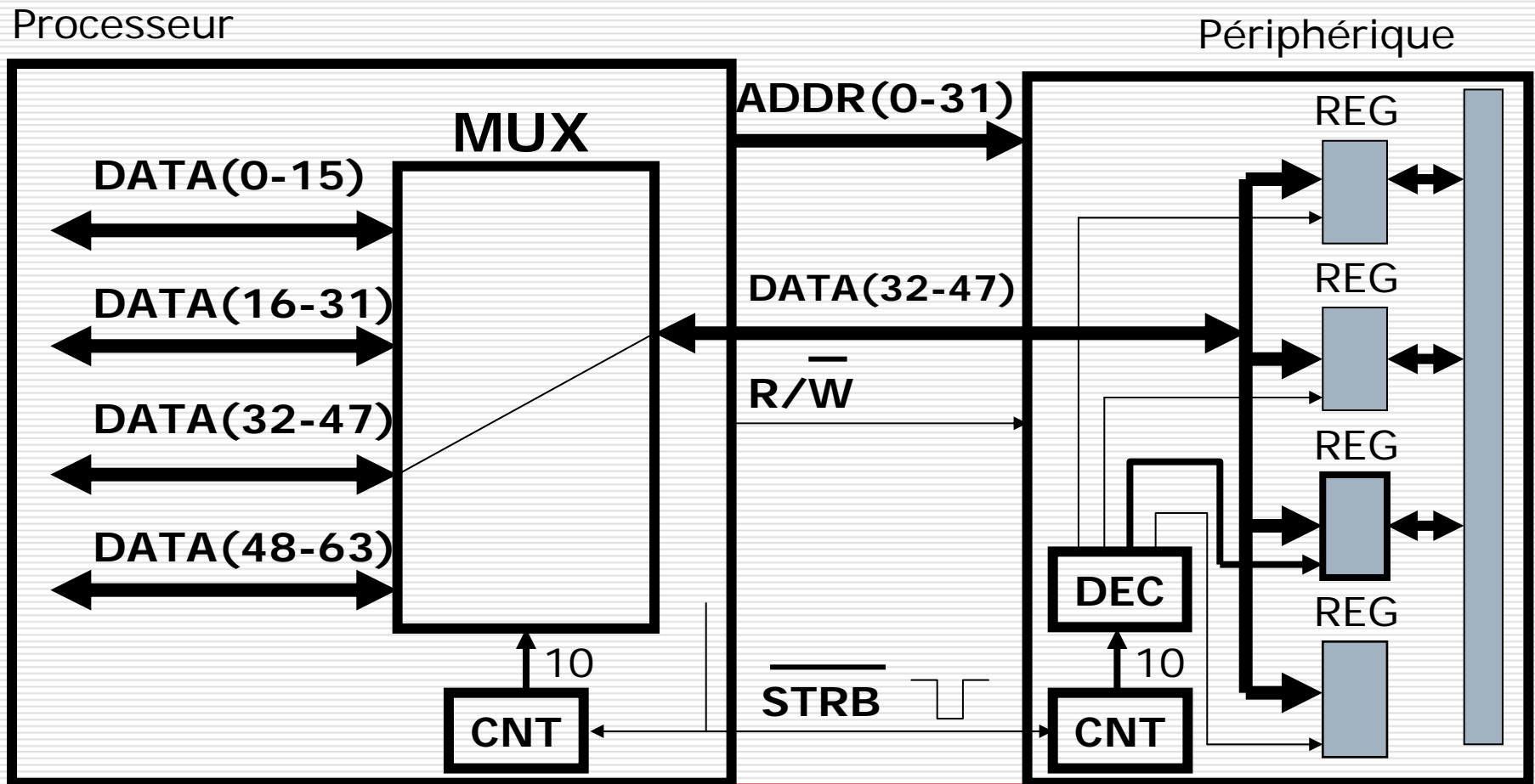
CNT: Compteur binaire
DEC: Décodeur

Lignes de données et d'adresses séparées – Données moins larges (réduire le nombre de lignes STRB) – T2



CNT: Compteur binaire
DEC: Décodeur

Lignes de données et d'adresses séparées – Données moins larges (réduire le nombre de lignes STRB) – T3

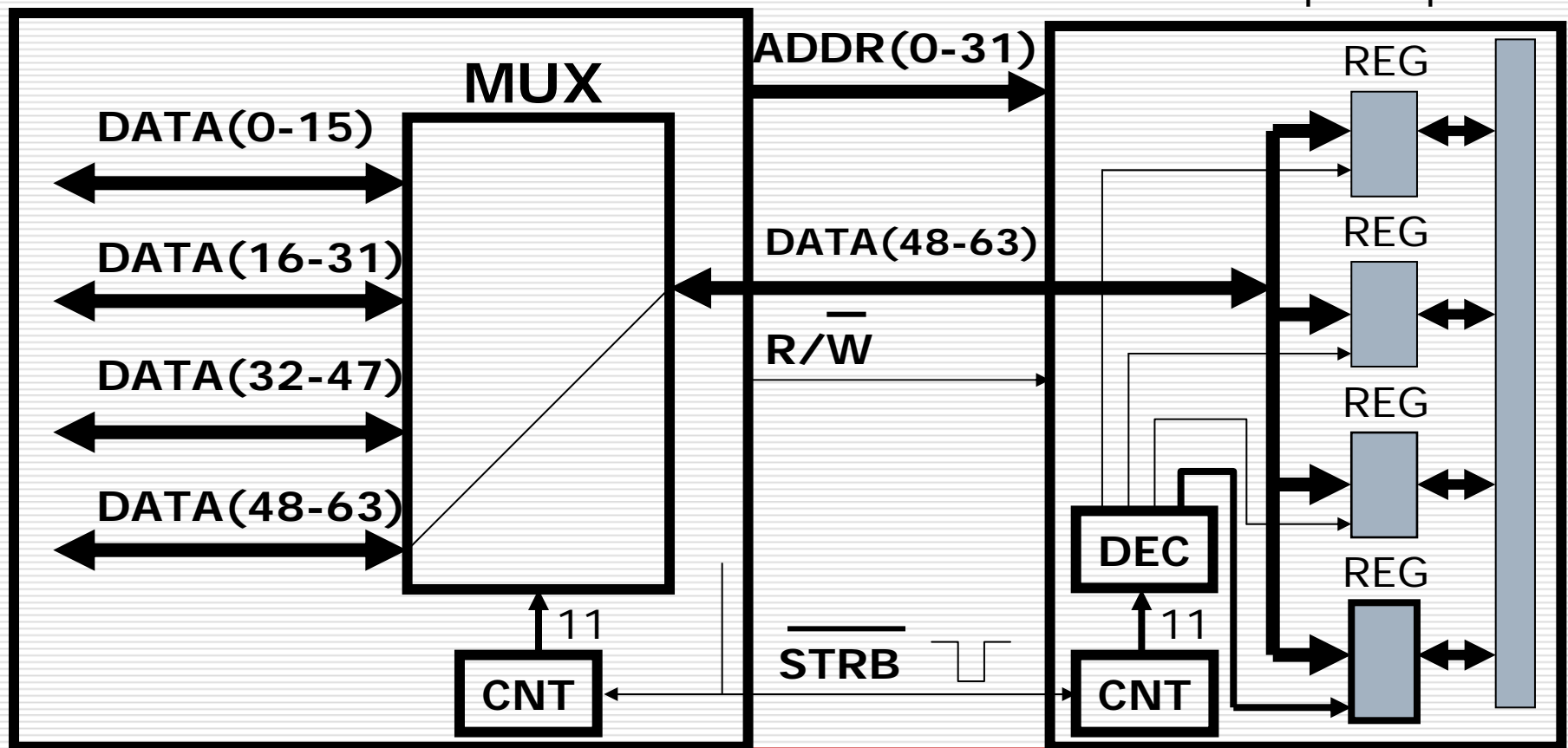


CNT: Compteur binaire
DEC: Décodeur

Lignes de données et d'adresses séparées – Données moins larges (réduire le nombre de lignes STRB) – T4

Processeur

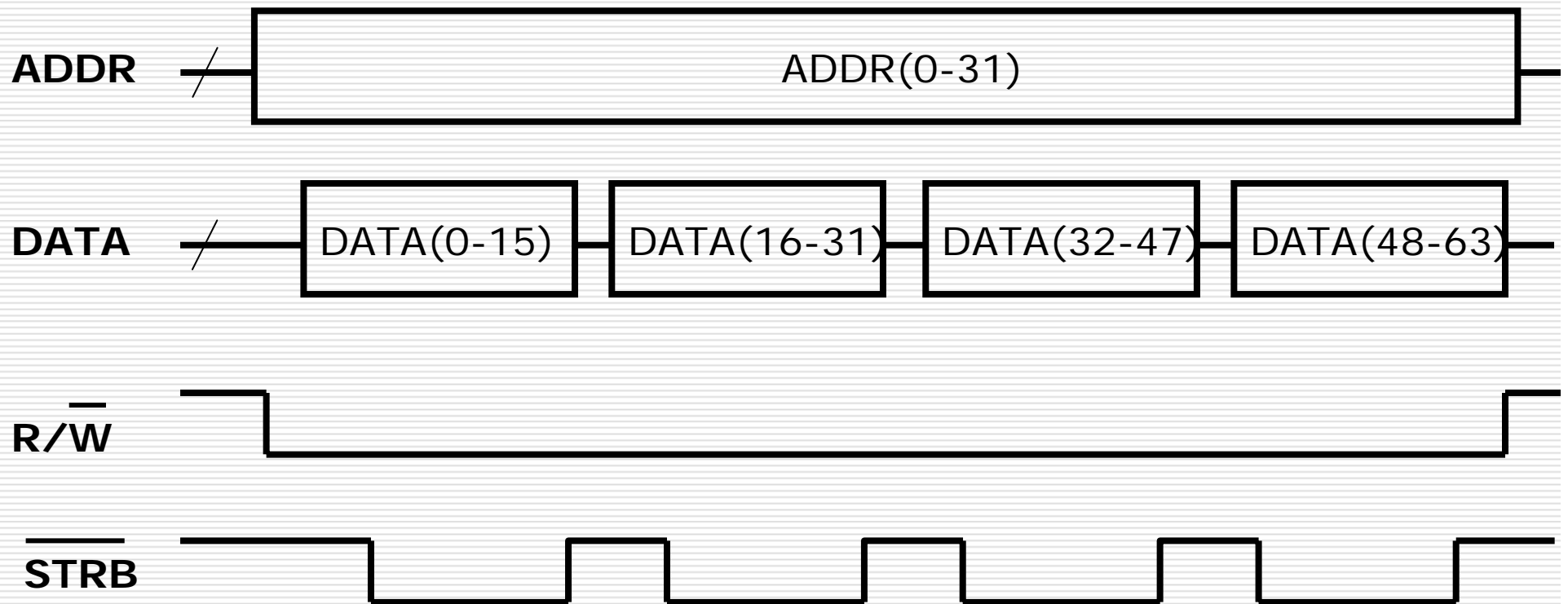
Périphérique



CNT: Compteur binaire
DEC: Décodeur

Lignes de données et d'adresses séparées

- Données moins larges (une ligne STRB) - Écriture sur le bus

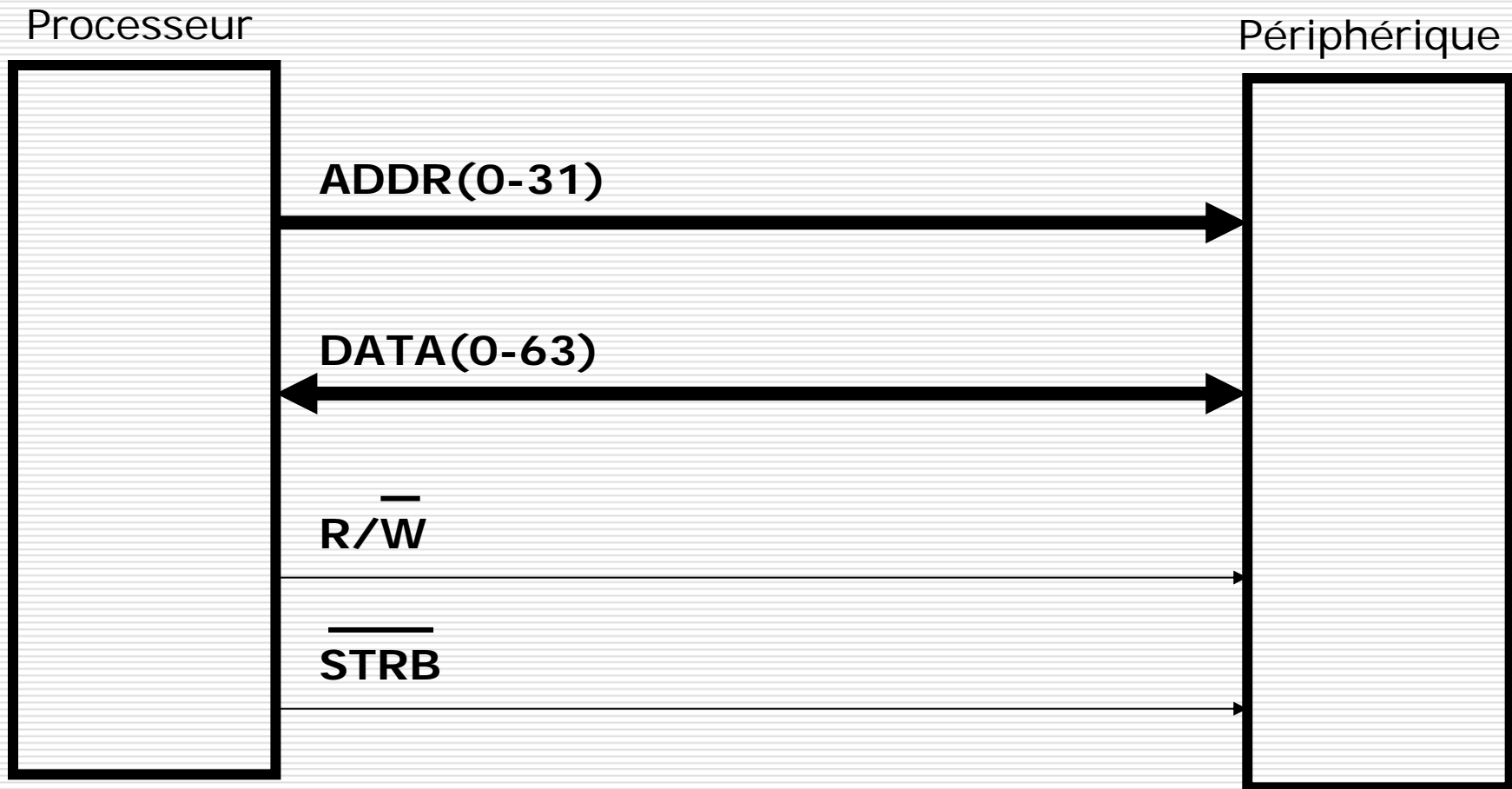


Principales options pour un bus

Option	Haute Performance	Moins Dispendieux
Largeur du bus	Lignes de données et d'adresses séparées	Lignes de données et d'adresses multiplexées
Largeur des données	Plus large est plus rapide (ex. 64 bits)	Moins large est moins dispendieux
Taille du transfert	Plusieurs mots ont moins d' « overhead »	<u>Transfert d'un mot à la fois est plus simple</u>

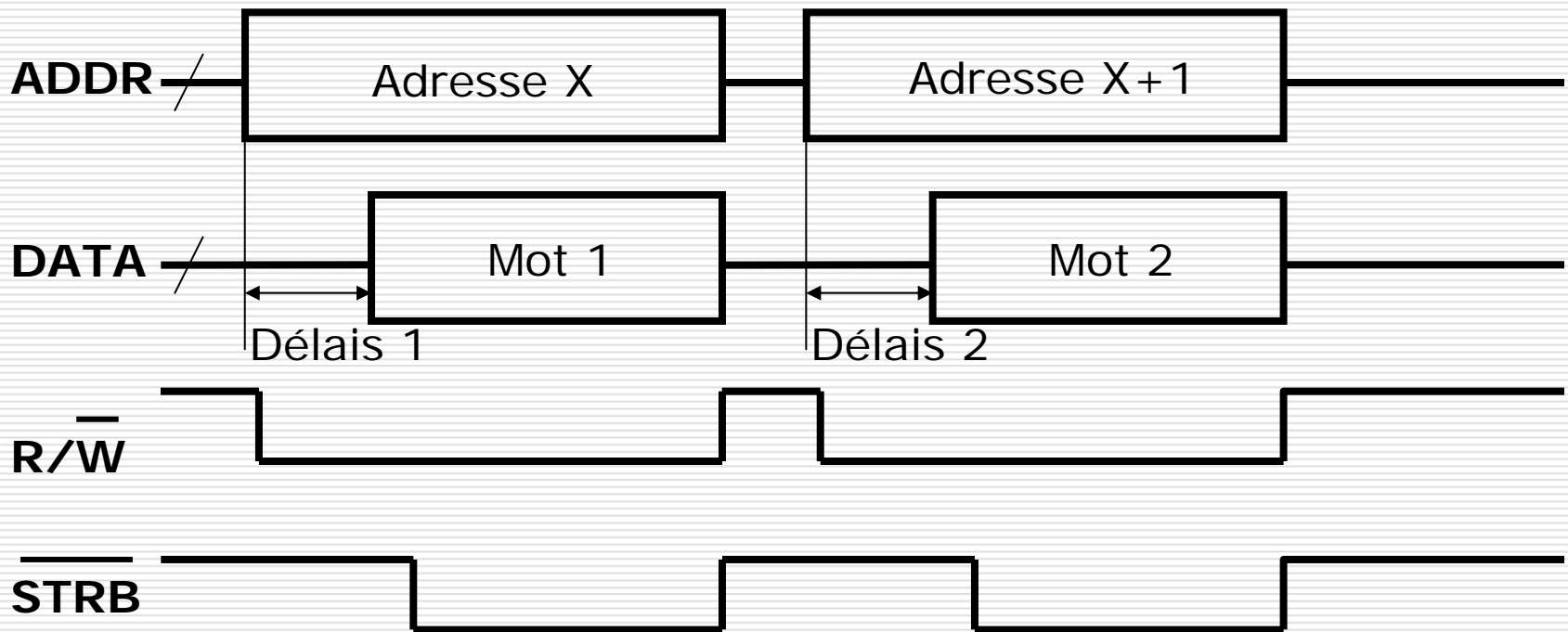
Lignes de données et d'adresses séparées

- Transfert d'un mot à la fois



Lignes de données et d'adresses séparées

- Transfert d'un mot à la fois –
- Écriture standard sur le bus



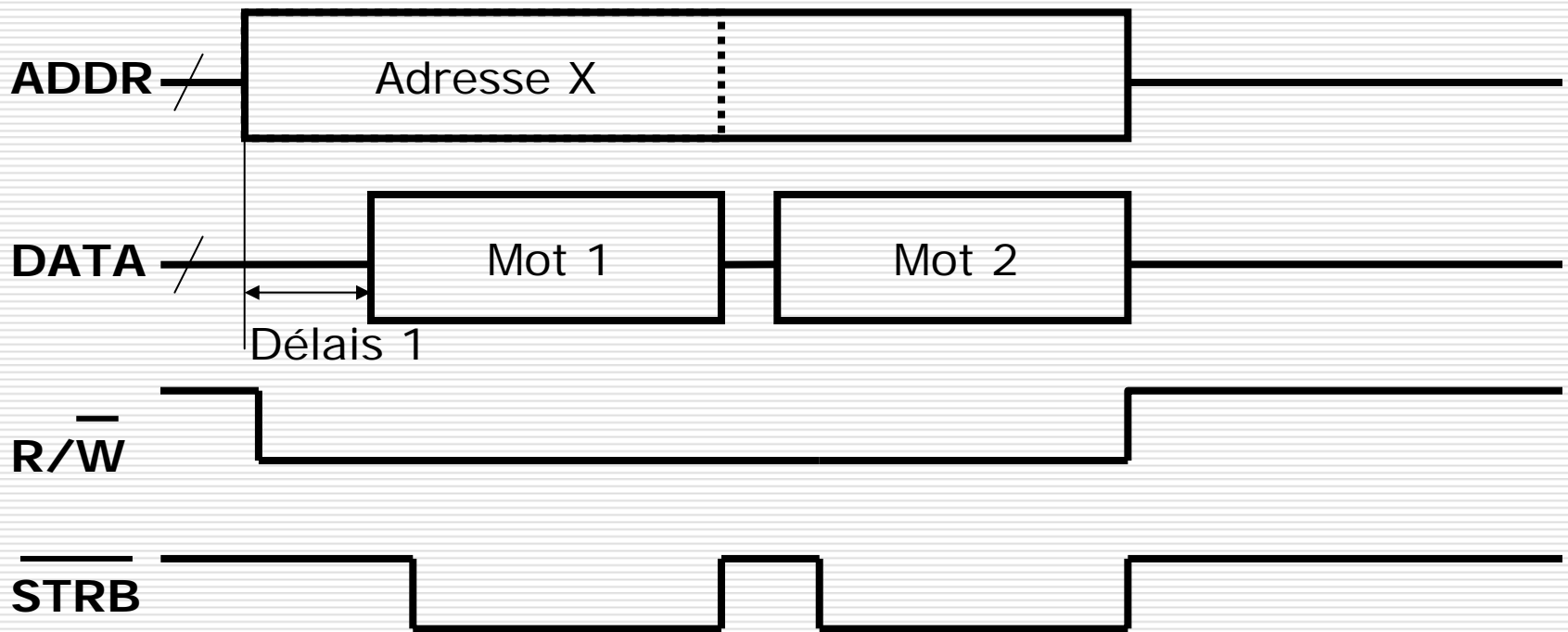
Note: Délais requis pour décodage

Principales options pour un bus

Option	Haute Performance	Moins Dispendieux
Largeur du bus	Lignes de données et d'adresses séparées	Lignes de données et d'adresses multiplexées
Largeur des données	Plus large est plus rapide (ex. 64 bits)	Moins large est moins dispendieux
Taille du transfert	<u>Plusieurs mots ont moins d' « overhead »</u>	Transfert d'un mot à la fois est plus simple

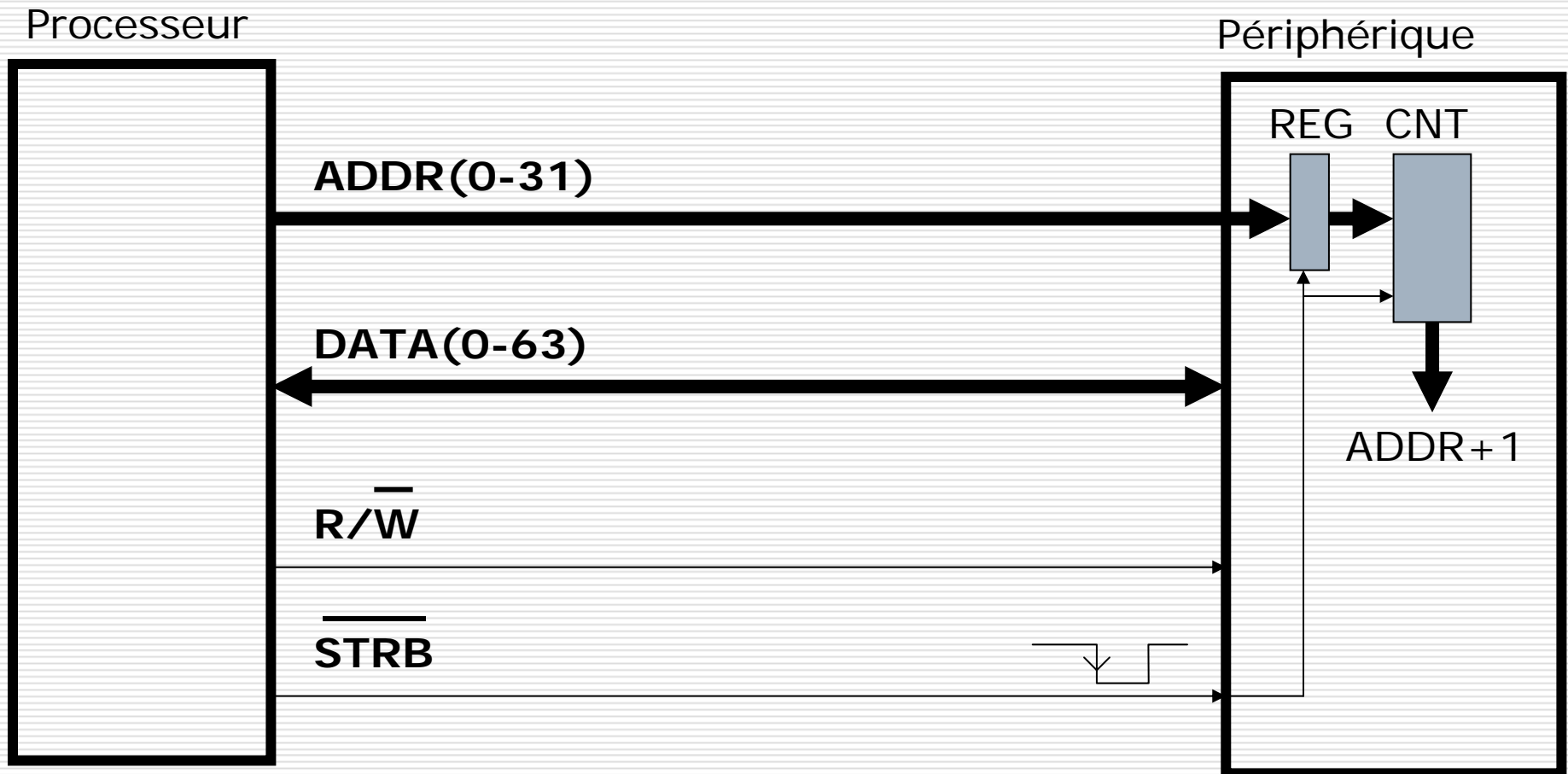
Lignes de données et d'adresses séparées

- Transfert de plusieurs mots à la fois –
- Écriture standard sur le bus

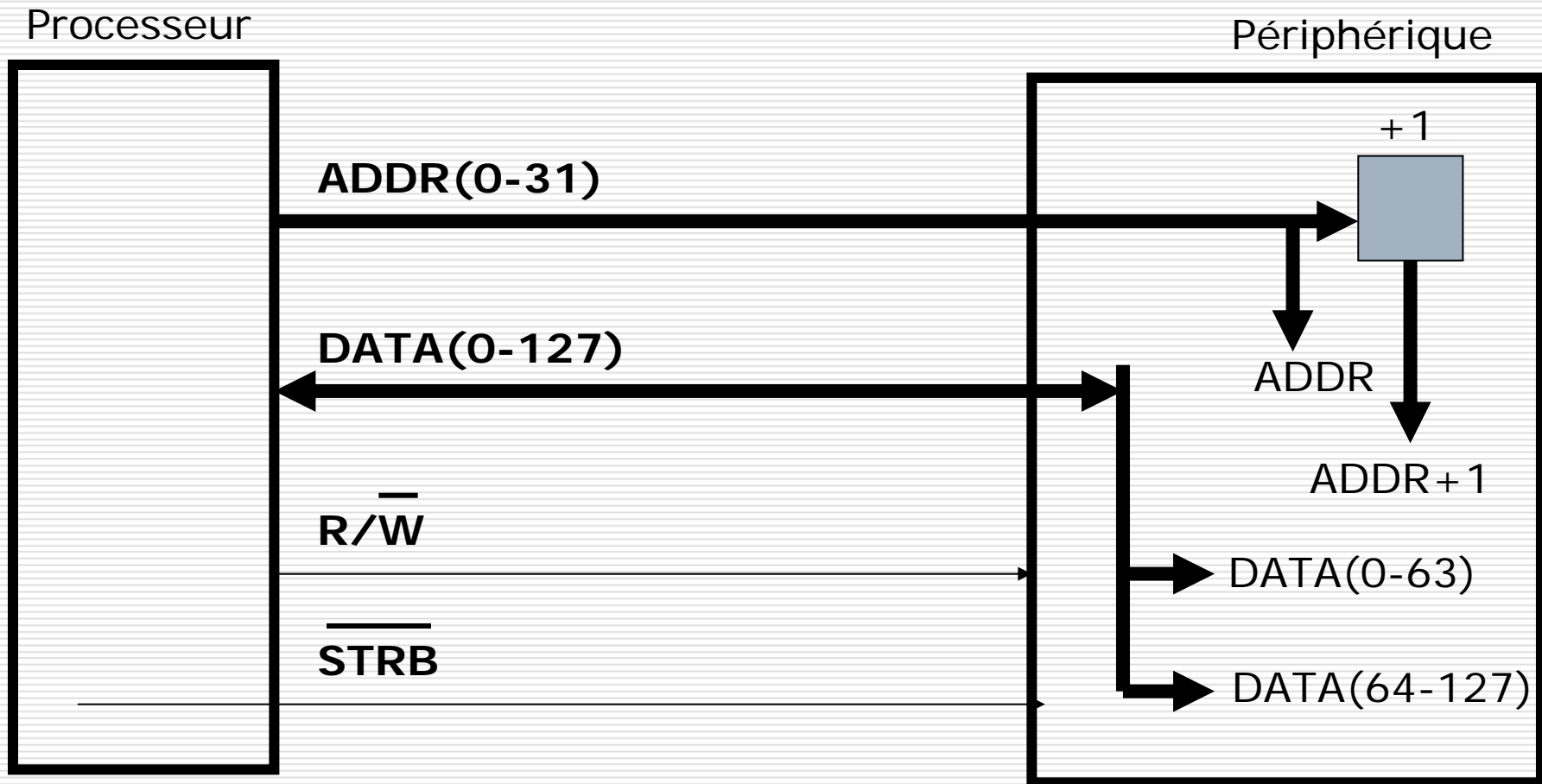


Note: 1 délais requis seulement pour décodage (adresses séquentielles)

Lignes de données et d'adresses séparées –
Transfert de plusieurs mots à la fois (typiquement
pour accès séquentiels)



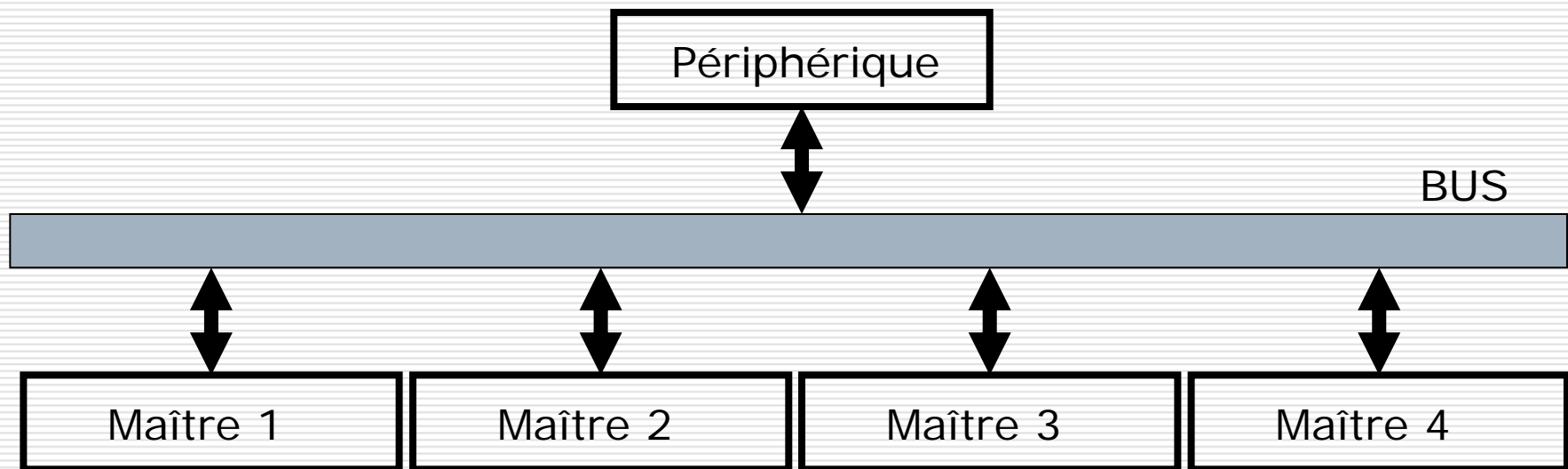
Lignes de données et d'adresses séparées – Transfert de 2 mots de 64-bit à la fois (complexe et limité)



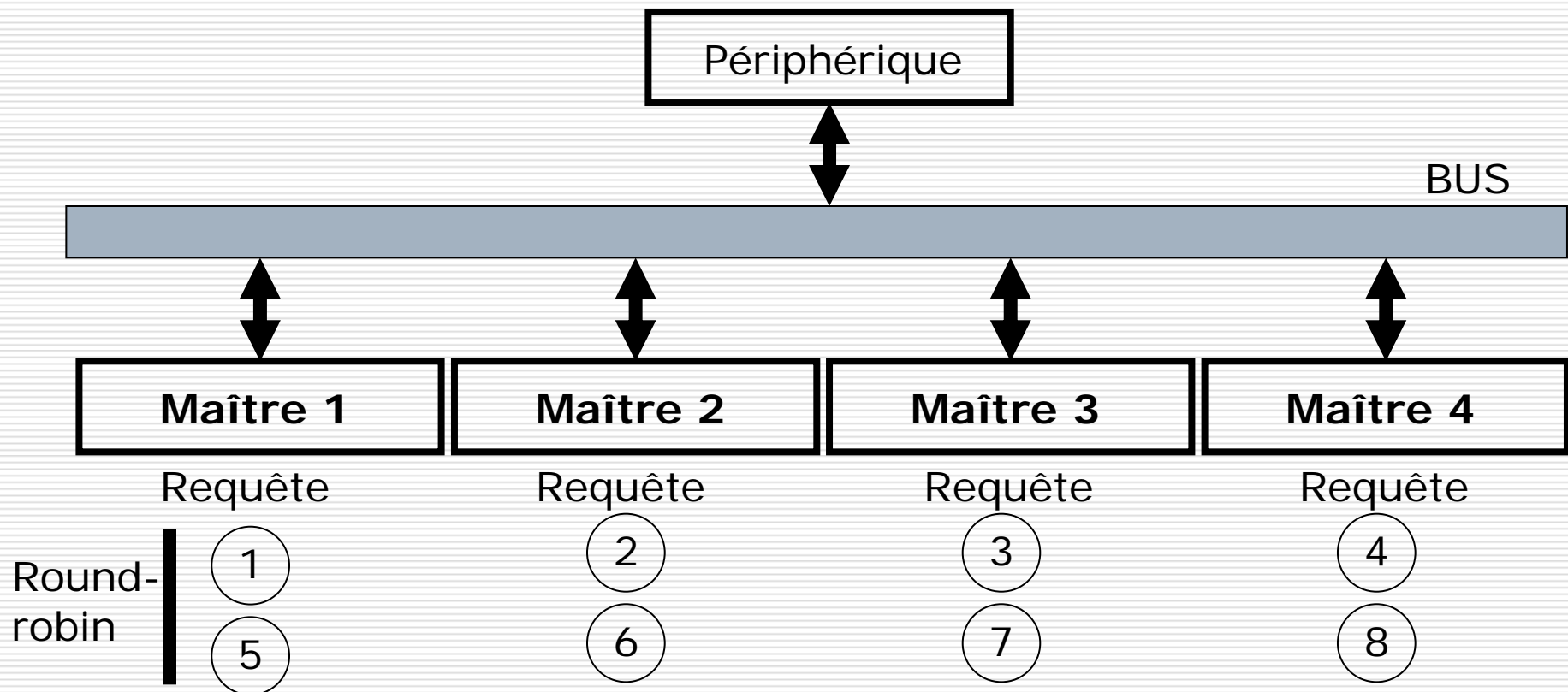
Principales options pour un bus (suite)

Option	Haute Performance	Moins Dispendieux
Maîtres du bus	<u>Plusieurs (requiert arbitration)</u>	Un seul maître (pas d'arbitration)
Transactions éclatées (split transactions)	Oui – paquets de requête and réponse (reply) séparés pour une bande passante plus grande (plusieurs maîtres)	Non – connection continue est moins dispendieuse et a moins de latence
Horloge de synchronization (Clocking)	Synchrone	Asynchrone

Plusieurs maîtres sur le bus



Plusieurs maîtres sur le bus – arbitration round-robin, aléatoire, etc.

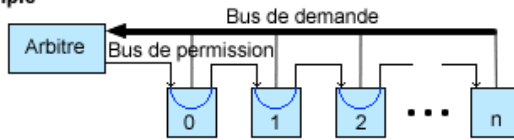


Plusieurs maîtres sur le bus (suite)

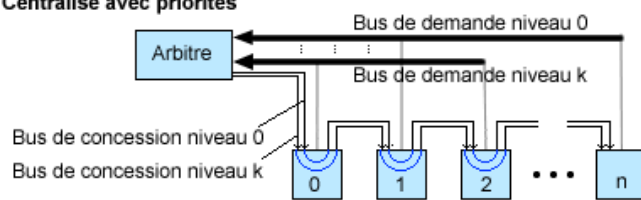
- Seules les maîtres du bus peuvent démarrer une transaction lecture ou écriture.
- Un bus a plusieurs maîtres quand il y a plusieurs UCT ou quand les composants d'E/S peuvent initialiser une transaction sur le bus.
- Un système d'arbitrage permet de gérer les différents maîtres.
- Avec plusieurs maîtres, un bus peut offrir une bande passante plus élevée en manipulant des paquets, par opposition à l'utilisation bloquée du bus pour une transaction complète. Cela se nomme les transactions éclatées.
- Un bus utilisant ce système possède une plus grande bande passante mais possède également une latence plus importante qu'un bus qui est conservé pendant la totalité de la transaction.

Bus

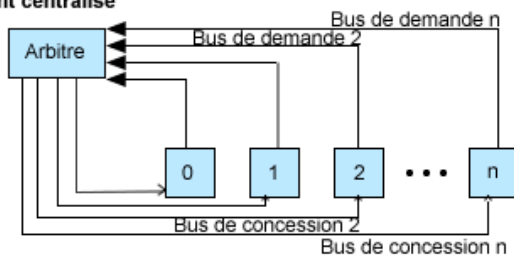
Centralisé simple



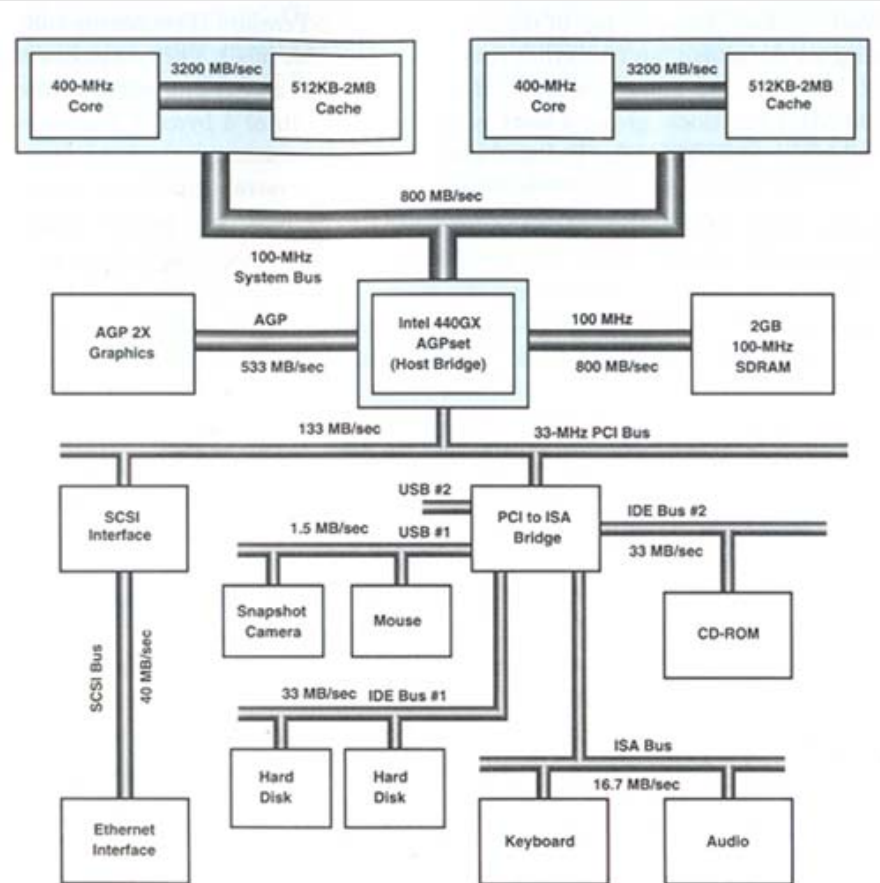
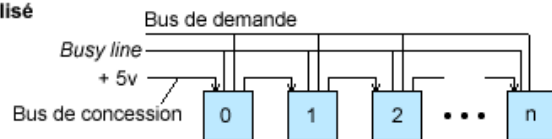
Centralisé avec priorités



Complètement centralisé



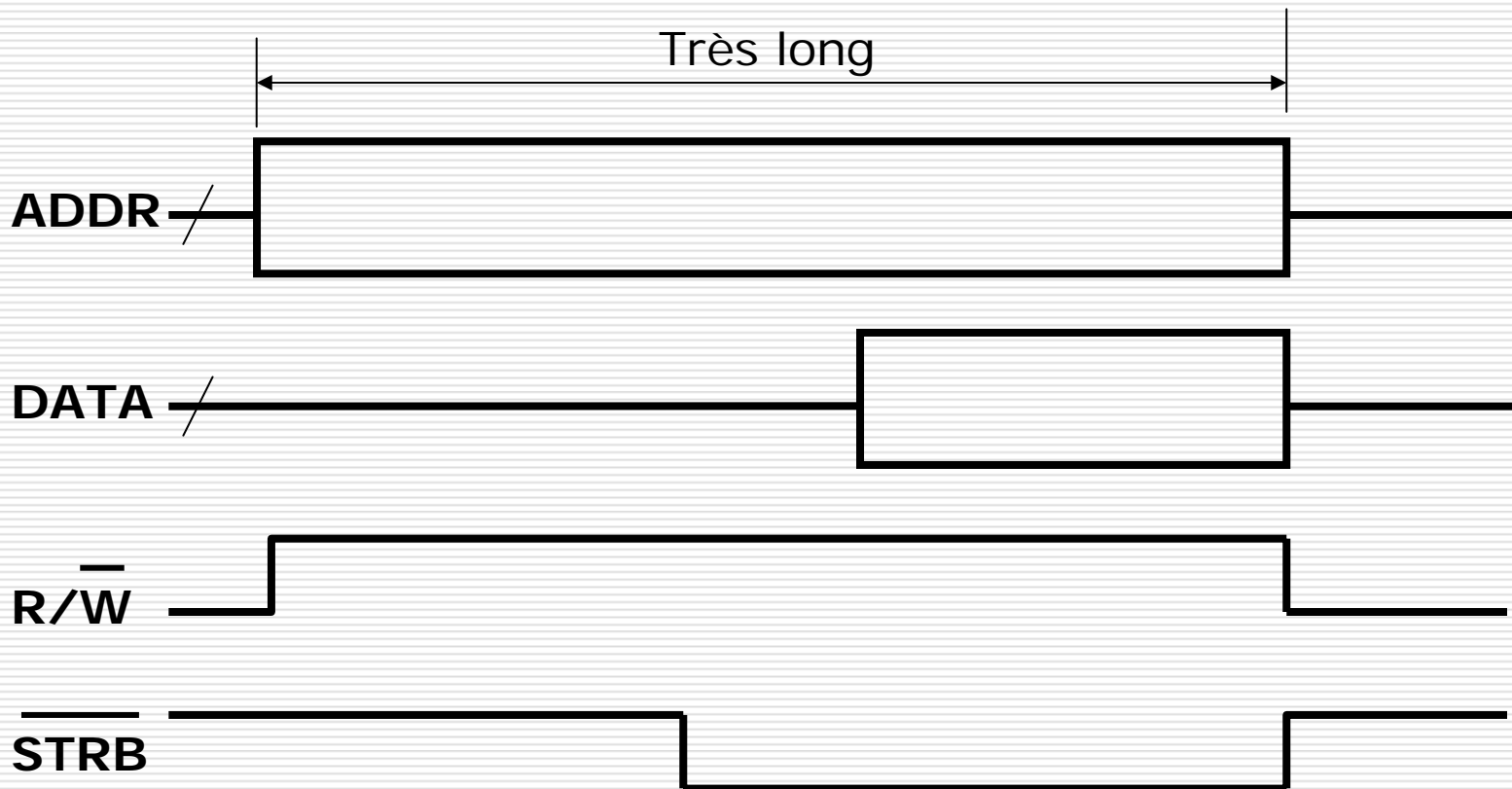
Décentralisé



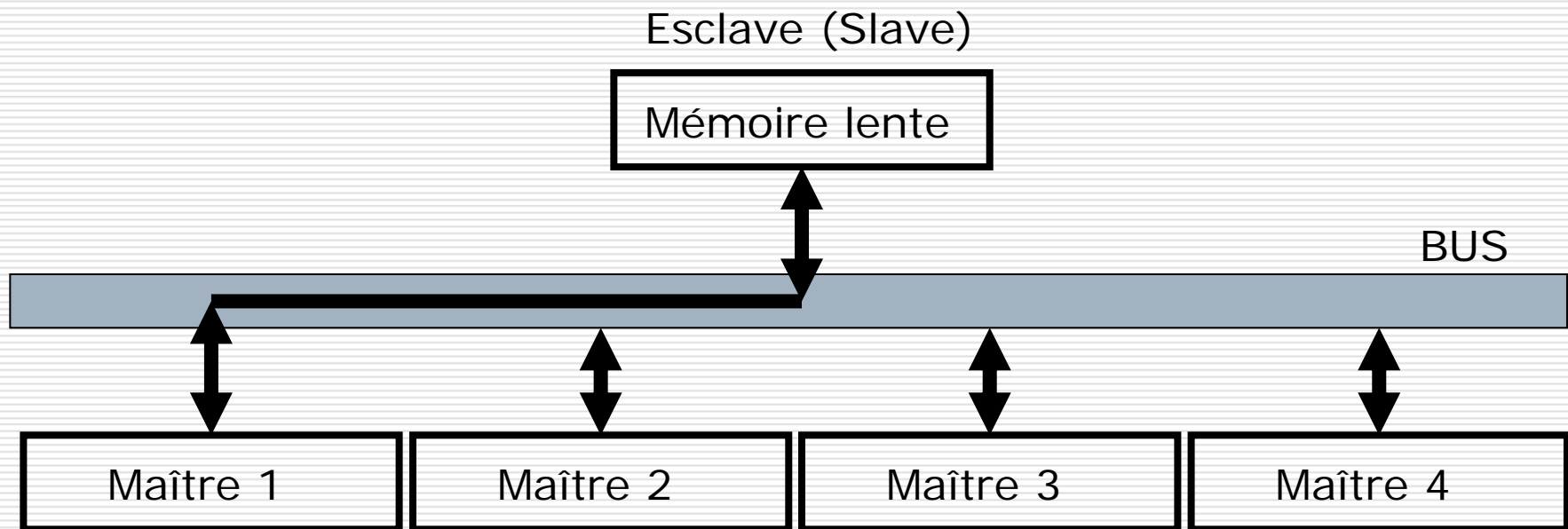
Principales options pour un bus (suite)

Option	Haute Performance	Moins Dispendieux
Maîtres du bus	Plusieurs (requiert arbitration)	Un seul maître (pas d'arbitration)
Transactions éclatées (split transactions)	<u>Oui – paquets de requête and réponse (reply) séparés pour une bande passante plus grande (plusieurs maîtres)</u>	Non – connection continue est moins dispendieuse et a moins de latence
Horloge de synchronization (Clocking)	Synchrone	Asynchrone

Split transactions – Le cycle de lecture standard sur le bus peut-être très long si la le temps d'accès à la mémoire est très long

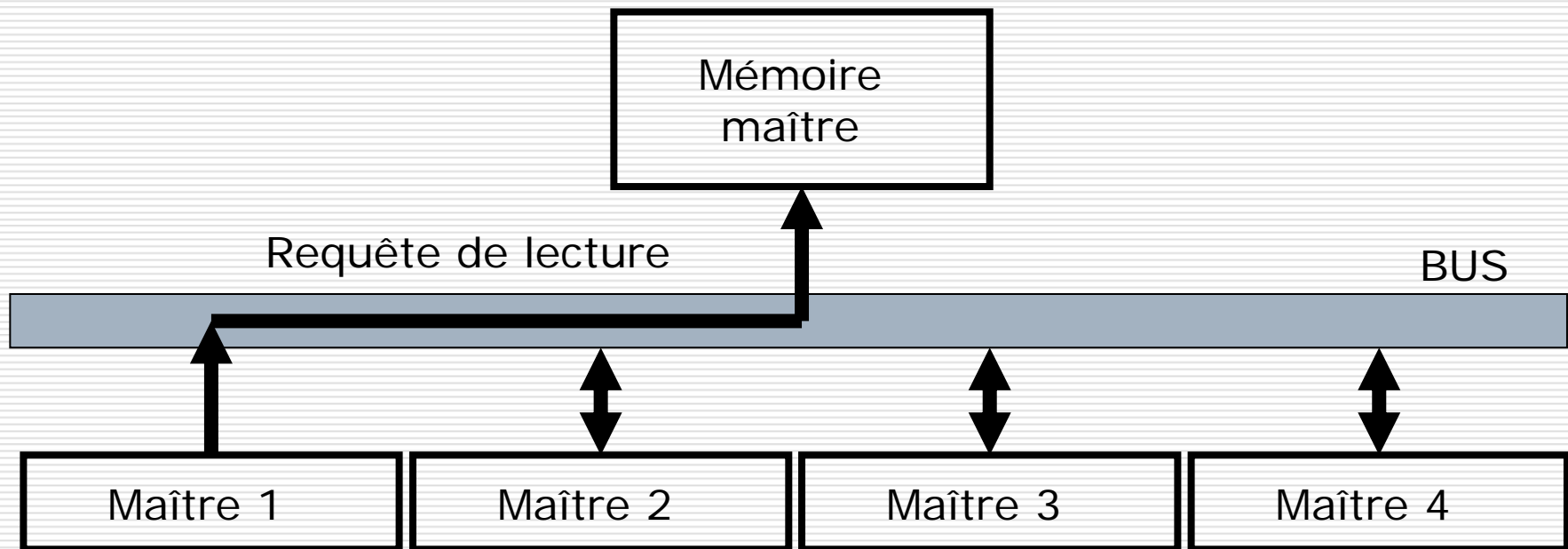


Transactions éclatées (Split Transactions) – Transaction standard



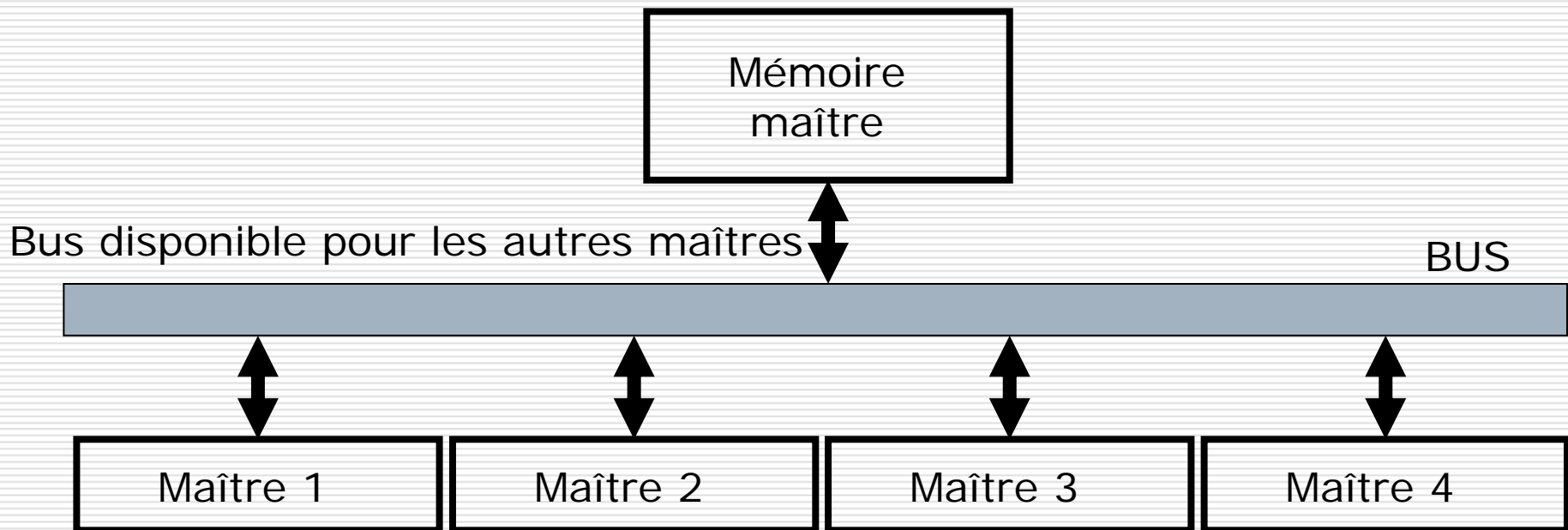
Pendant un long accès à la mémoire lente par le maître 1, la transaction standard occupe le bus ce qui empêche les autres maîtres d'utiliser le bus.

Transactions éclatées (Split Transactions): Simple exemple – T1



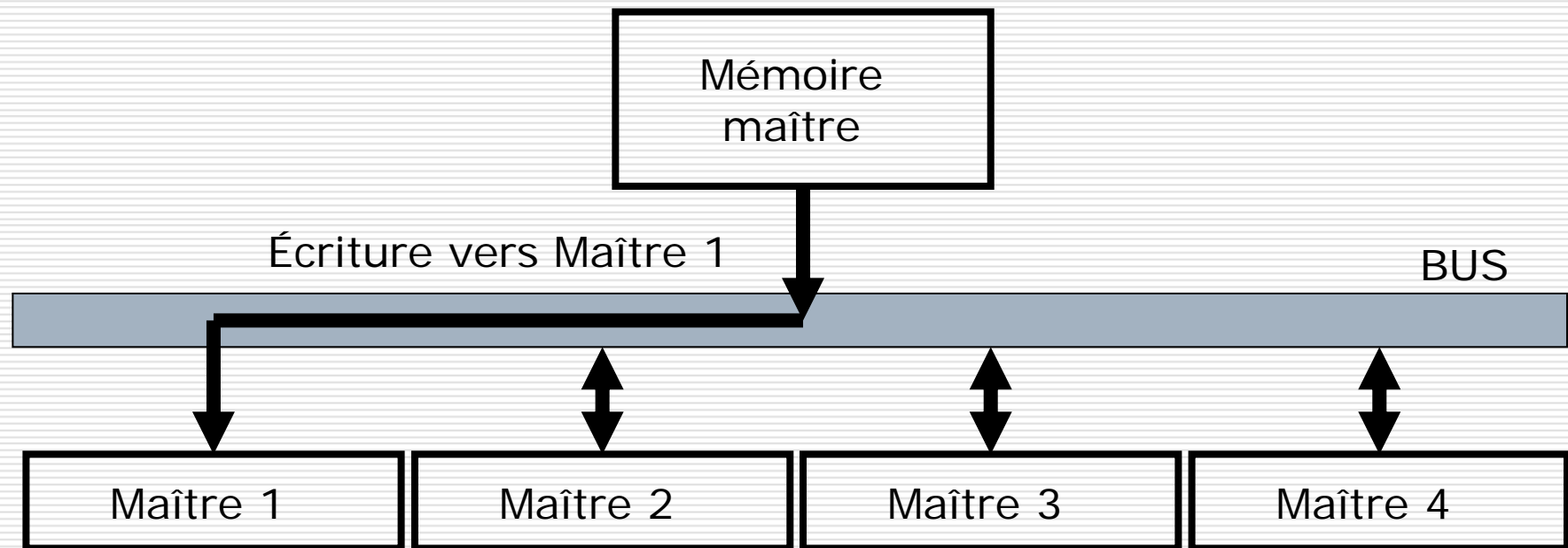
Le maître 1 envoie une requête de lecture à la mémoire maître. L'adresse de destination est enregistrée (latched) dans l'interface de la mémoire maître.

Transactions éclatées (Split Transactions): Simple exemple – T2



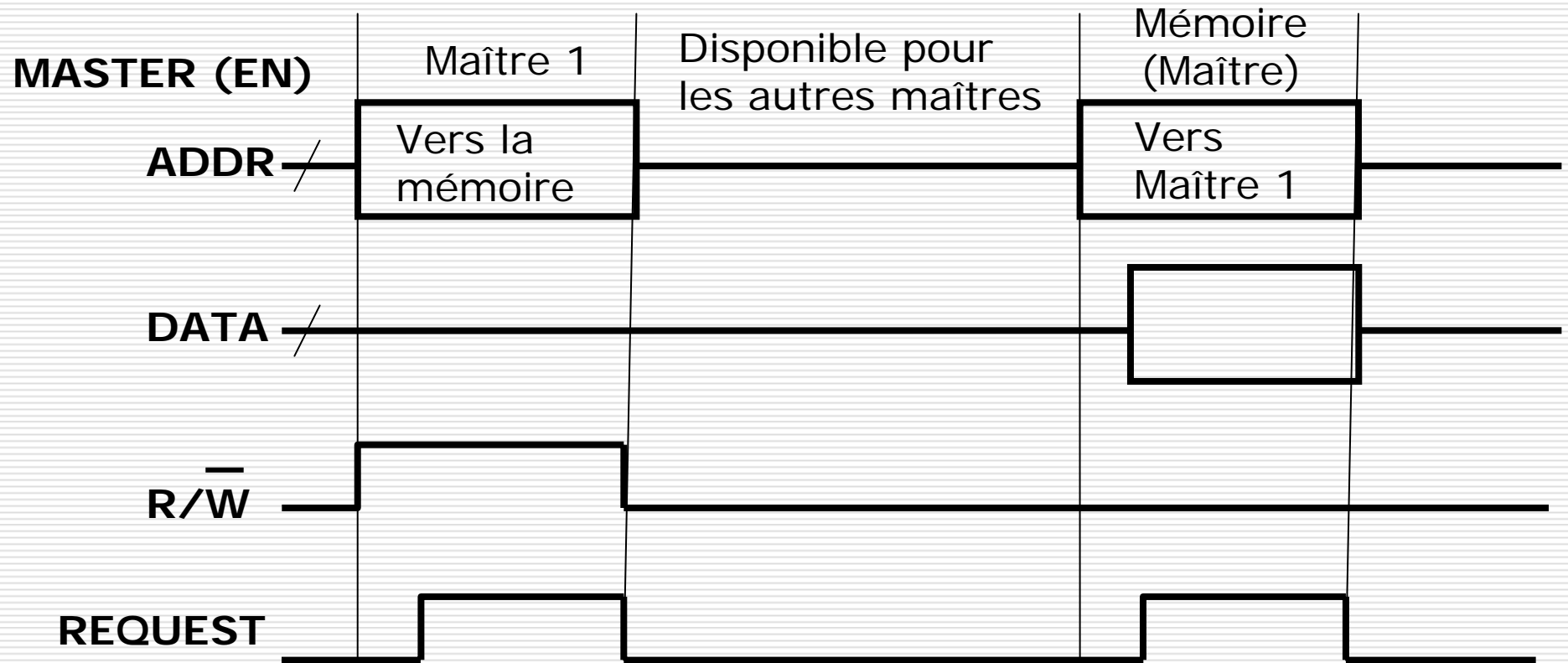
Pendant que l'adresse est décodée et le relativement long cycle d'accès dans la mémoire maître soit complété, le bus est libéré par le Maître 1 et disponible pour transactions par les autres maîtres.

Transactions éclatées (Split Transactions): Simple exemple – T3



Le maître 1 envoie une requête de lecture à la mémoire maître. L'adresse de destination est enregistrée (latched) dans l'interface de la mémoire maître.

Split transactions (Simple exemple)



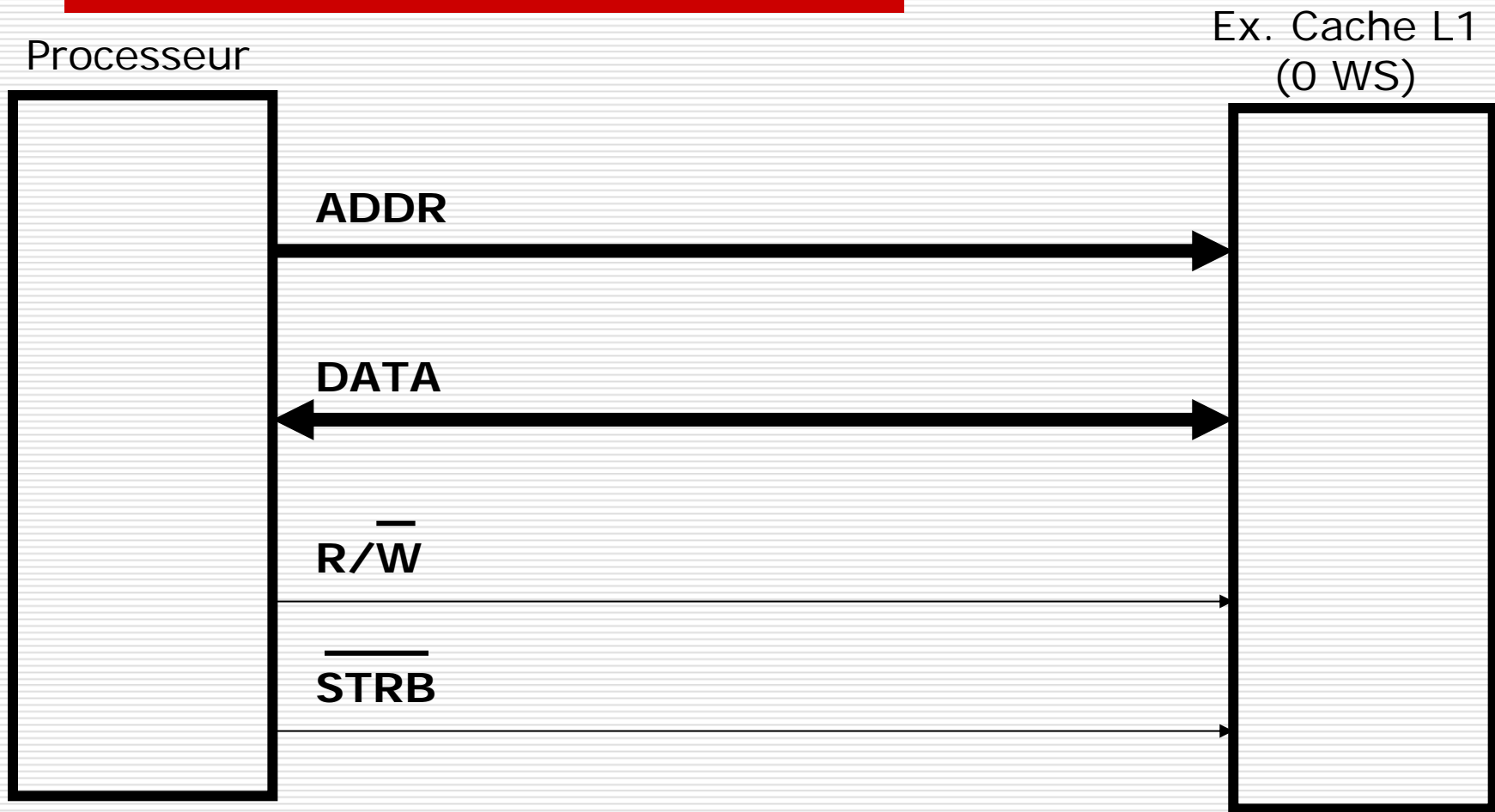
Principales options pour un bus (suite)

Option	Haute Performance	Moins Dispendieux
Maîtres du bus	Plusieurs (requiert arbitration)	Un seul maître (pas d'arbitration)
Transactions éclatées (split transactions)	Oui – paquets de requête and réponse (reply) séparés pour une bande passante plus grande (plusieurs maîtres)	Non – connection continue est moins dispendieuse et a moins de latence
<u>Horloge de synchronization (Clocking)</u>	<u>Synchrone</u>	<u>Asynchrone</u>

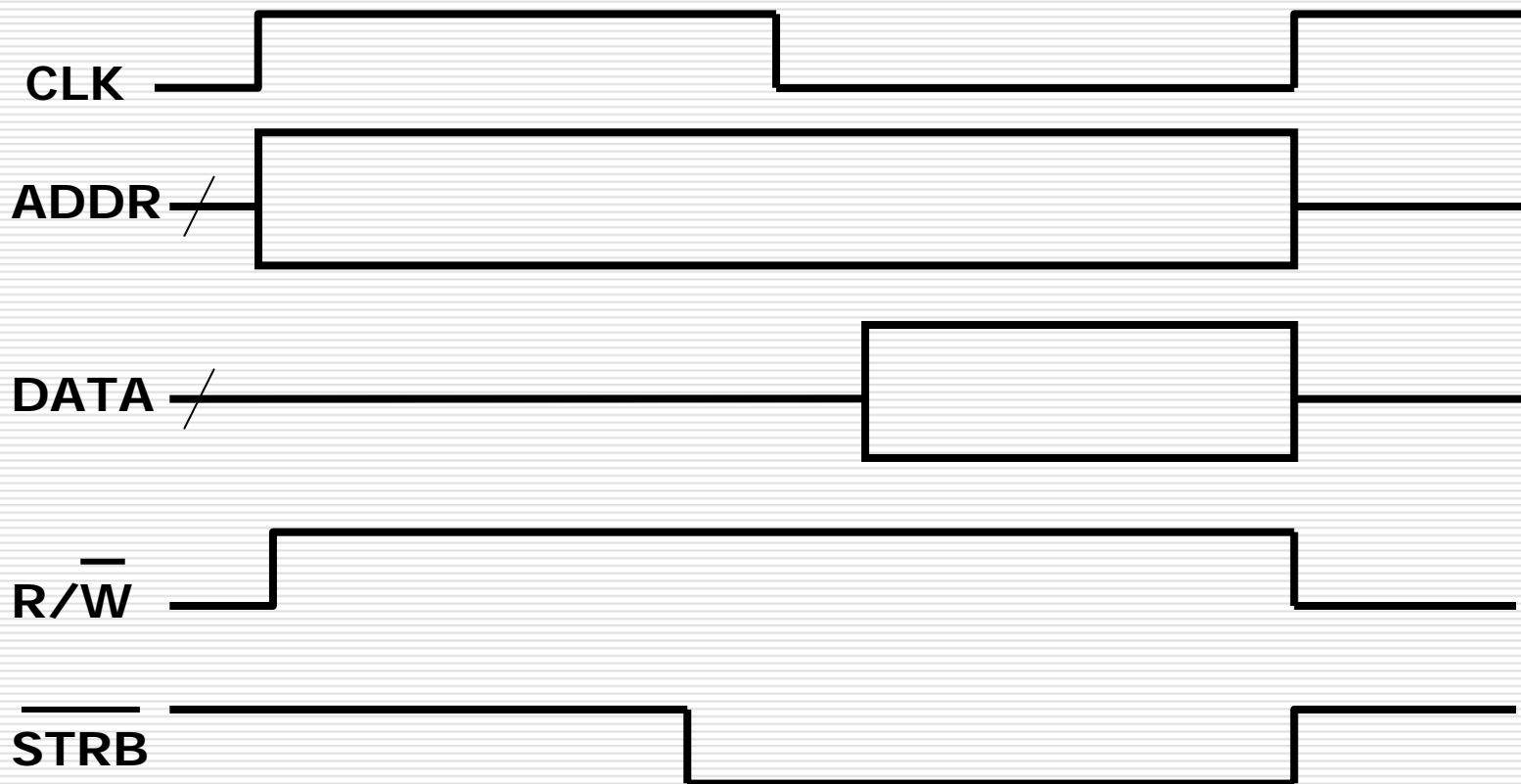
Synchrone ou asynchrone

- Un bus peut être synchrone (possède une horloge et un protocole pour les données et les adresses par rapport à l'horloge) ou asynchrone.

Accès standard sur un bus synchrone – 0 Wait-state (1 cycle horloge)



Lecture standard sur un bus synchrone – 0 Wait-state (1 cycle horloge)

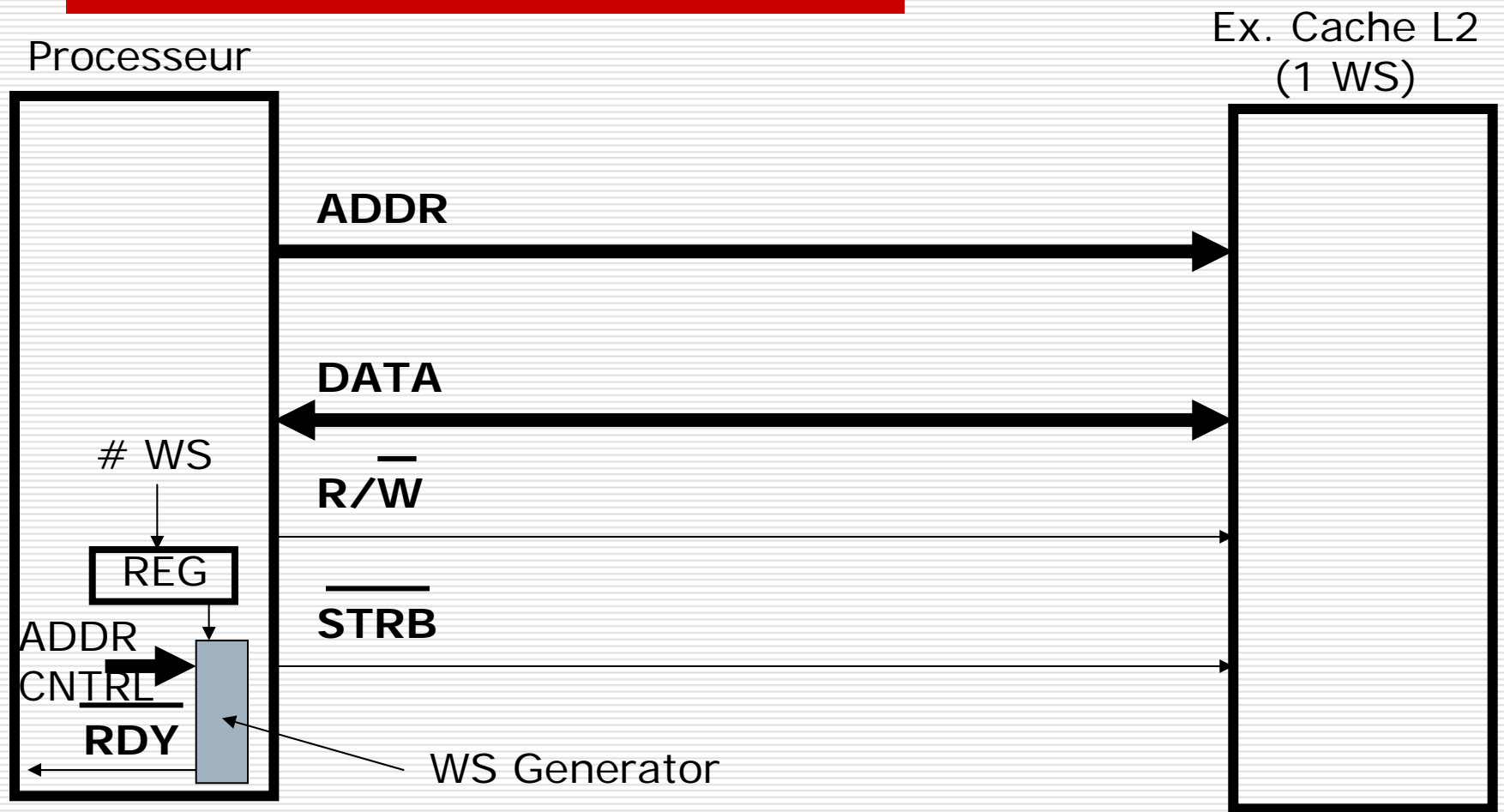


Note: CLK: Clock ou Horloge

Écriture standard sur un bus synchrone – 0 Wait-state (1 cycle horloge)

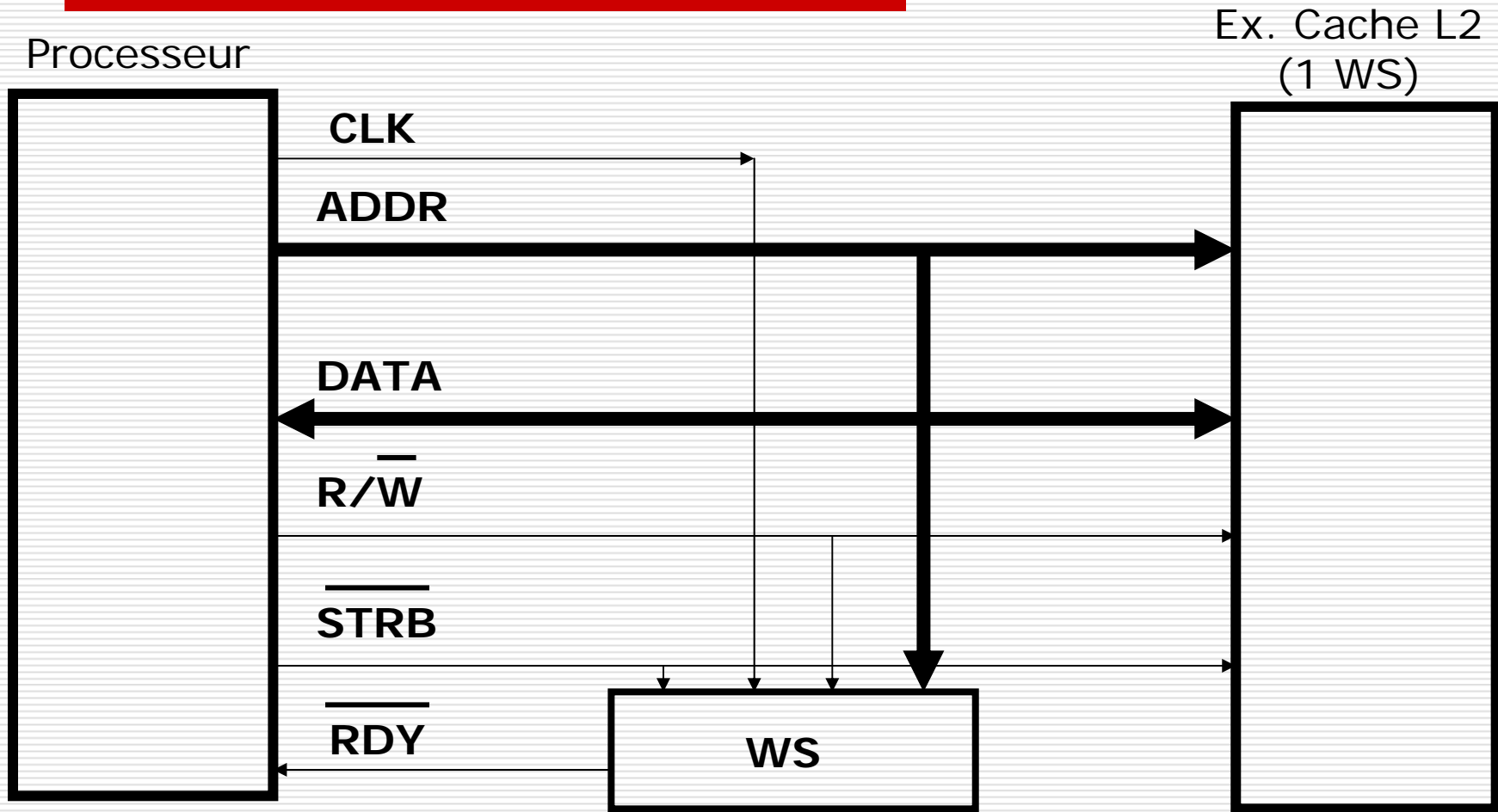


Accès standard sur un bus synchrone –
 1 Wait-state (2 cycles horloge) –
 Ready interne généré par WS générateur du processeur (software programmable par registre)
 (nombre de WS connu)



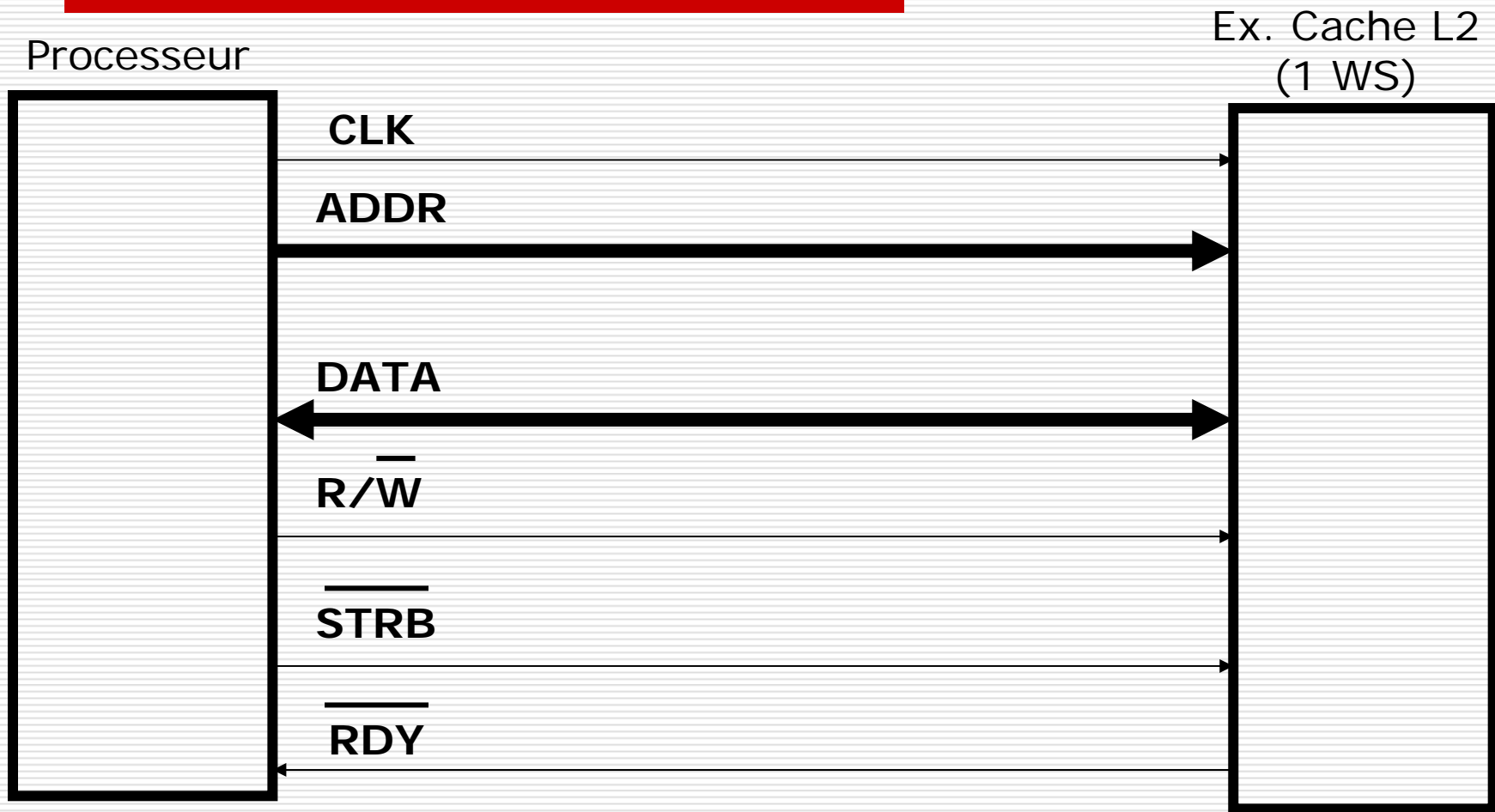
Note: RDY: Ready (Prêt) Sylvain Martel - INF6500

Accès standard sur un bus synchrone –
1 Wait-state (2 cycles horloge) –
Ready généré par WS générateur externe
(nombre de WS connu)



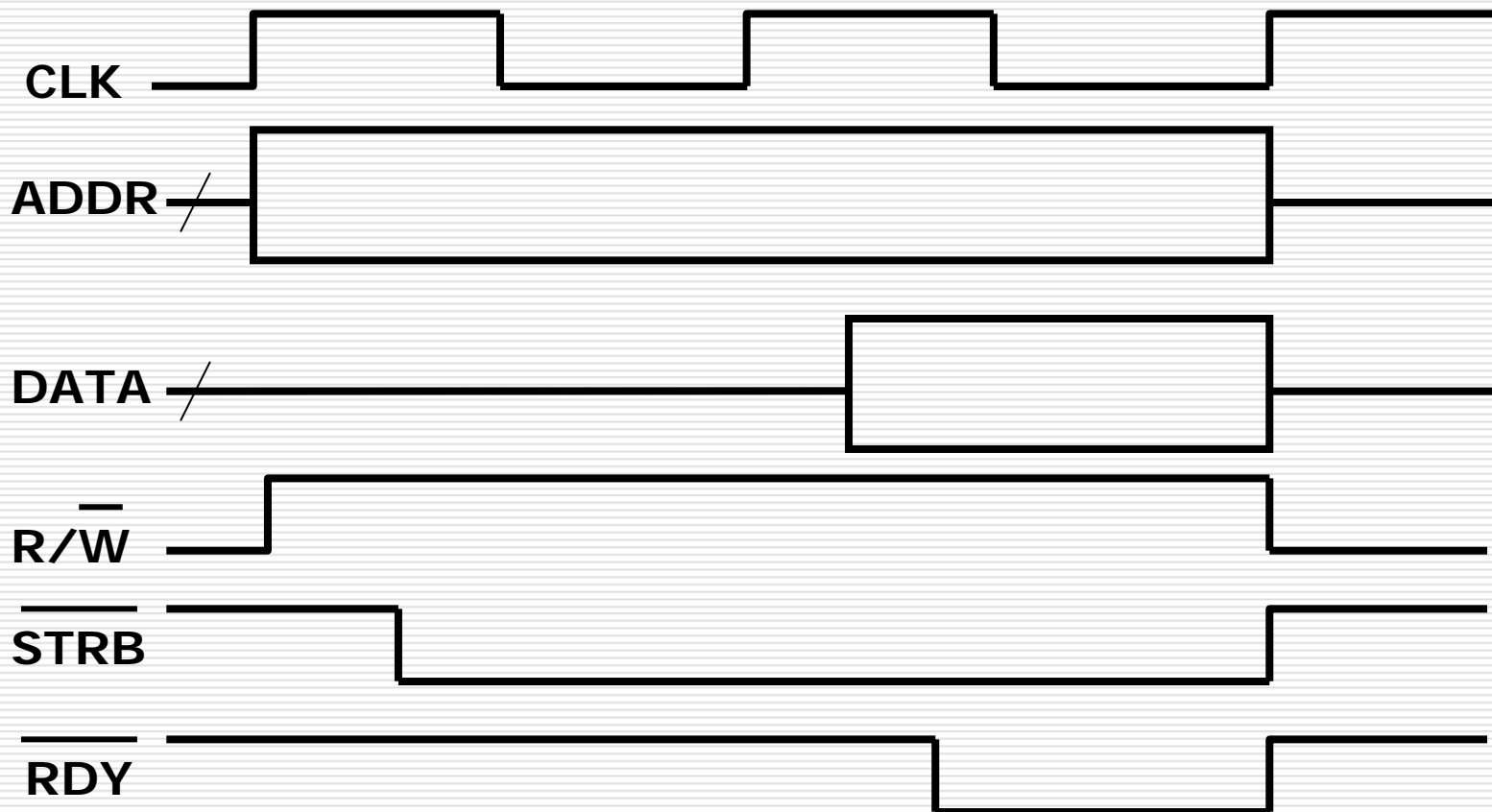
Note: RDY: Ready (Prêt) Sylvain Martel - INF6500

Accès standard sur un bus synchrone –
1 Wait-state (2 cycles horloge) –
Ready externe généré par WS générateur du
périphérique (nombre de WS connu ou inconnu)



Note: RDY: Ready (Prêt) Sylvain Martel - INF6500

Lecture standard sur un bus synchrone – 1 Wait-state (2 cycles horloge)

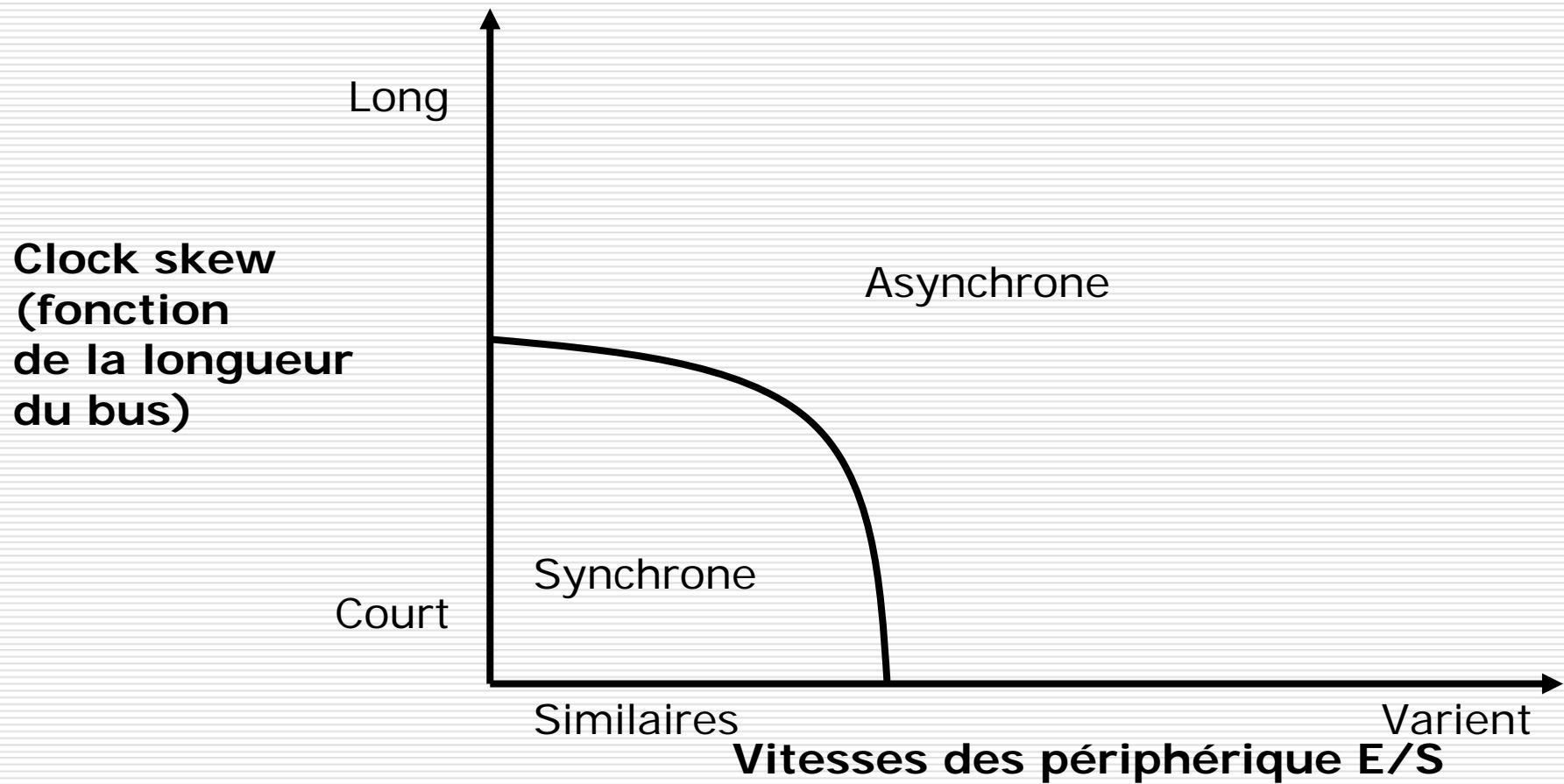


Synchrone versus asynchrone

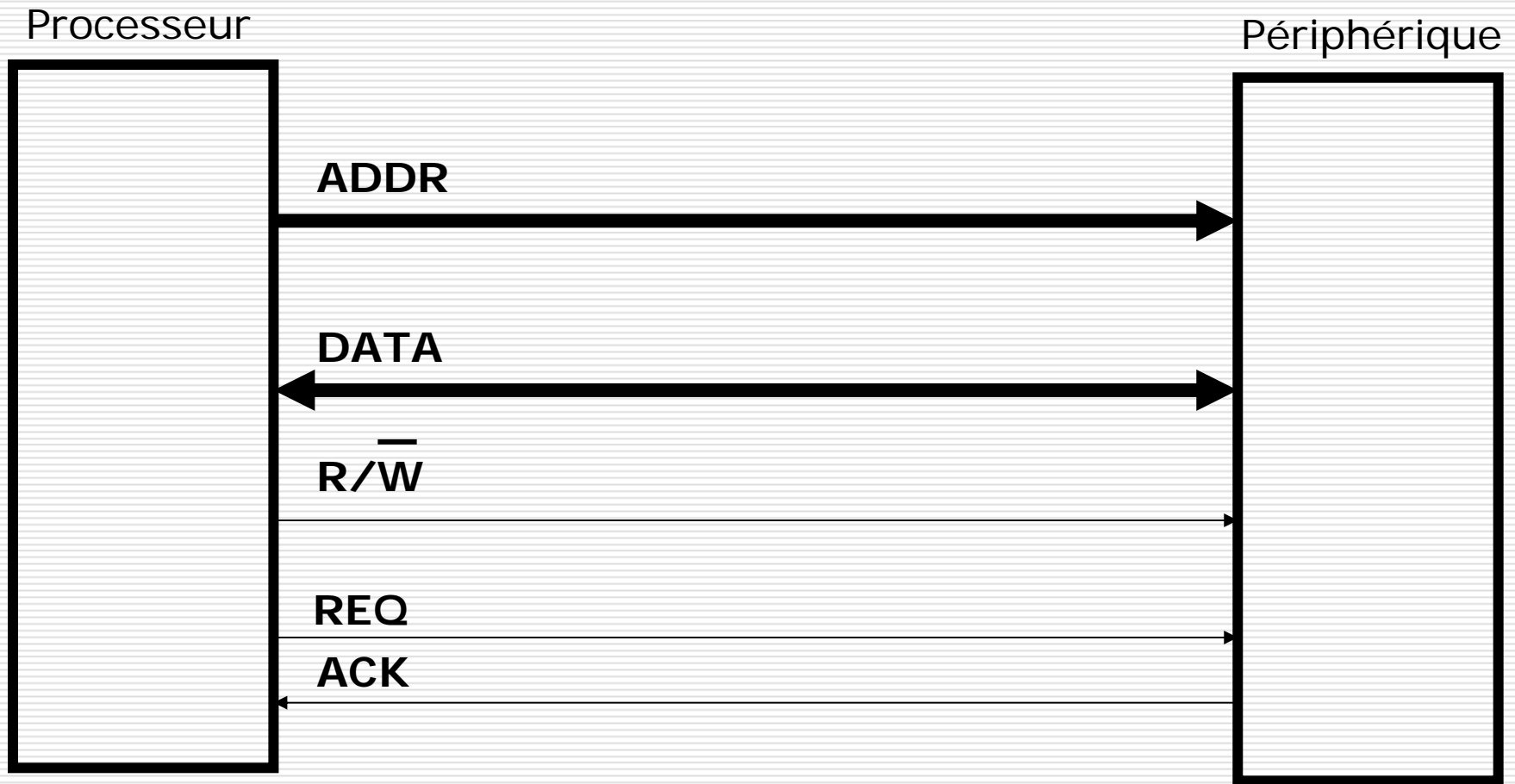
Les bus synchrones sont généralement rapides et peu chers mais en contrepartie, tout le bus doit fonctionner à la même fréquence d'horloge.

Un bus asynchrone n'a pas d'horloge, il possède des protocoles de poignée de main (*hand shaking*). Ce type de bus permet l'allongement du bus sans se préoccuper de dispersions d'horloge (clock skew).

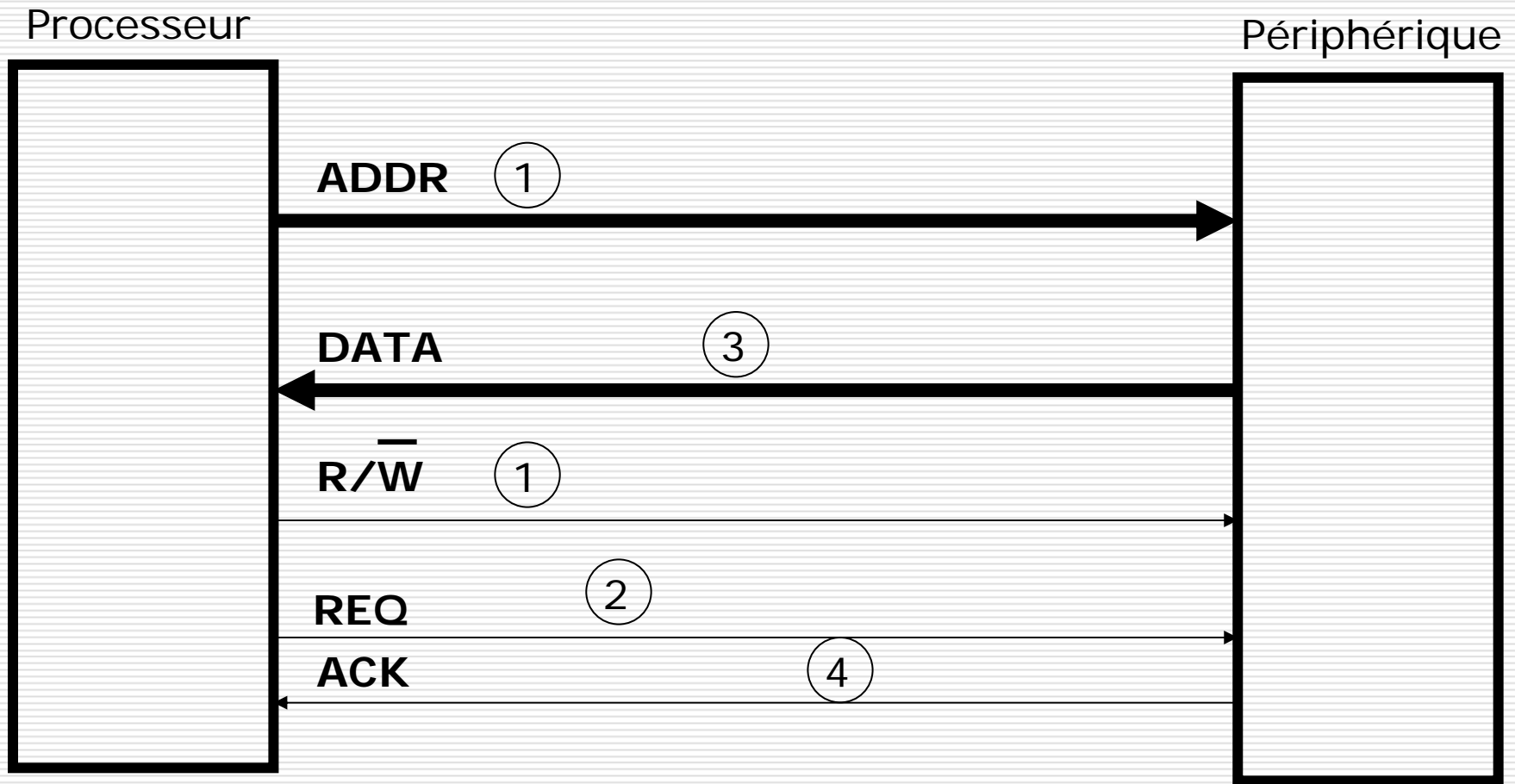
Synchrone versus asynchrone



Accès standard sur un bus asynchrone

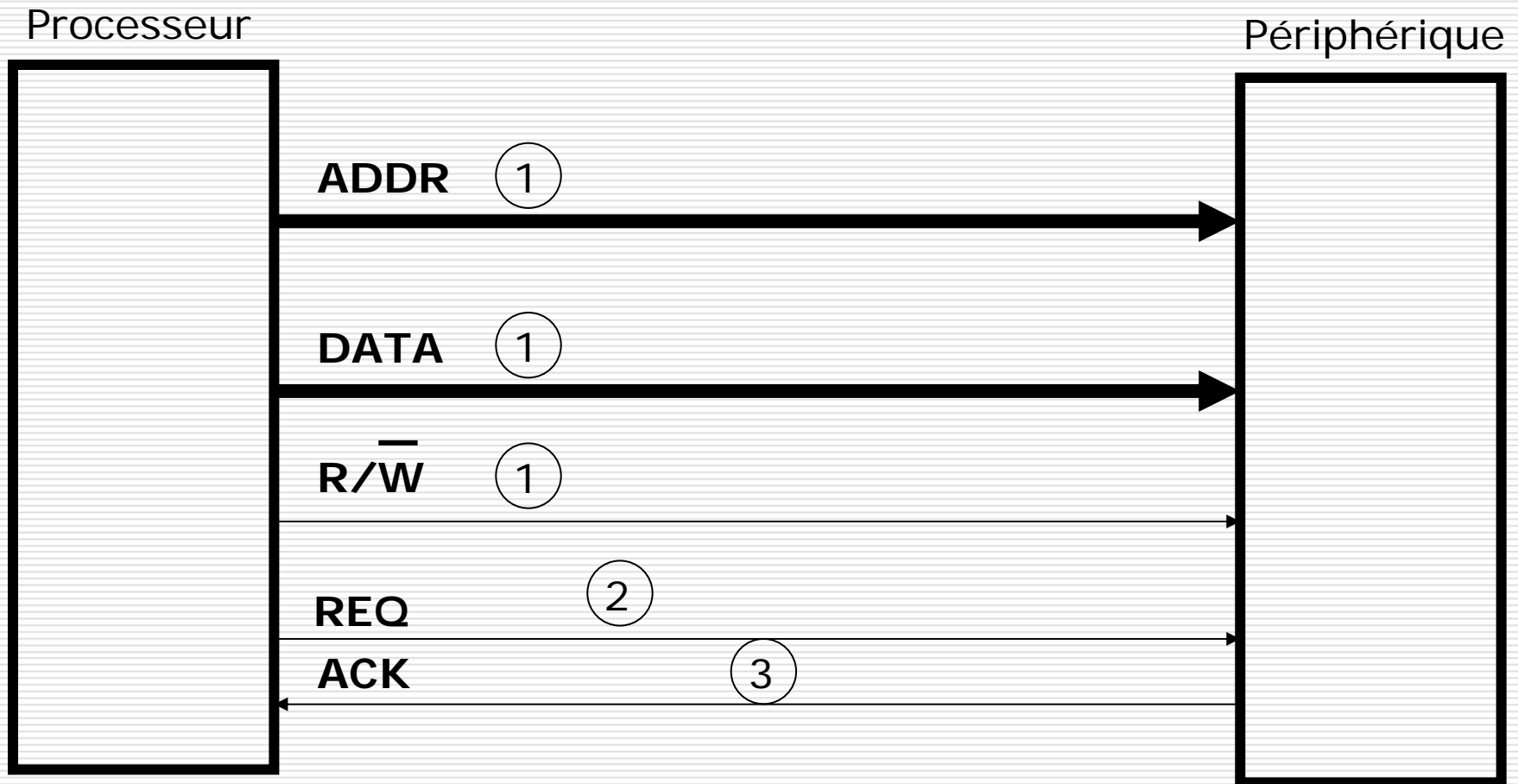


Lecture sur un bus asynchrone



Note: REQ: Request, ACK: Acknowledge

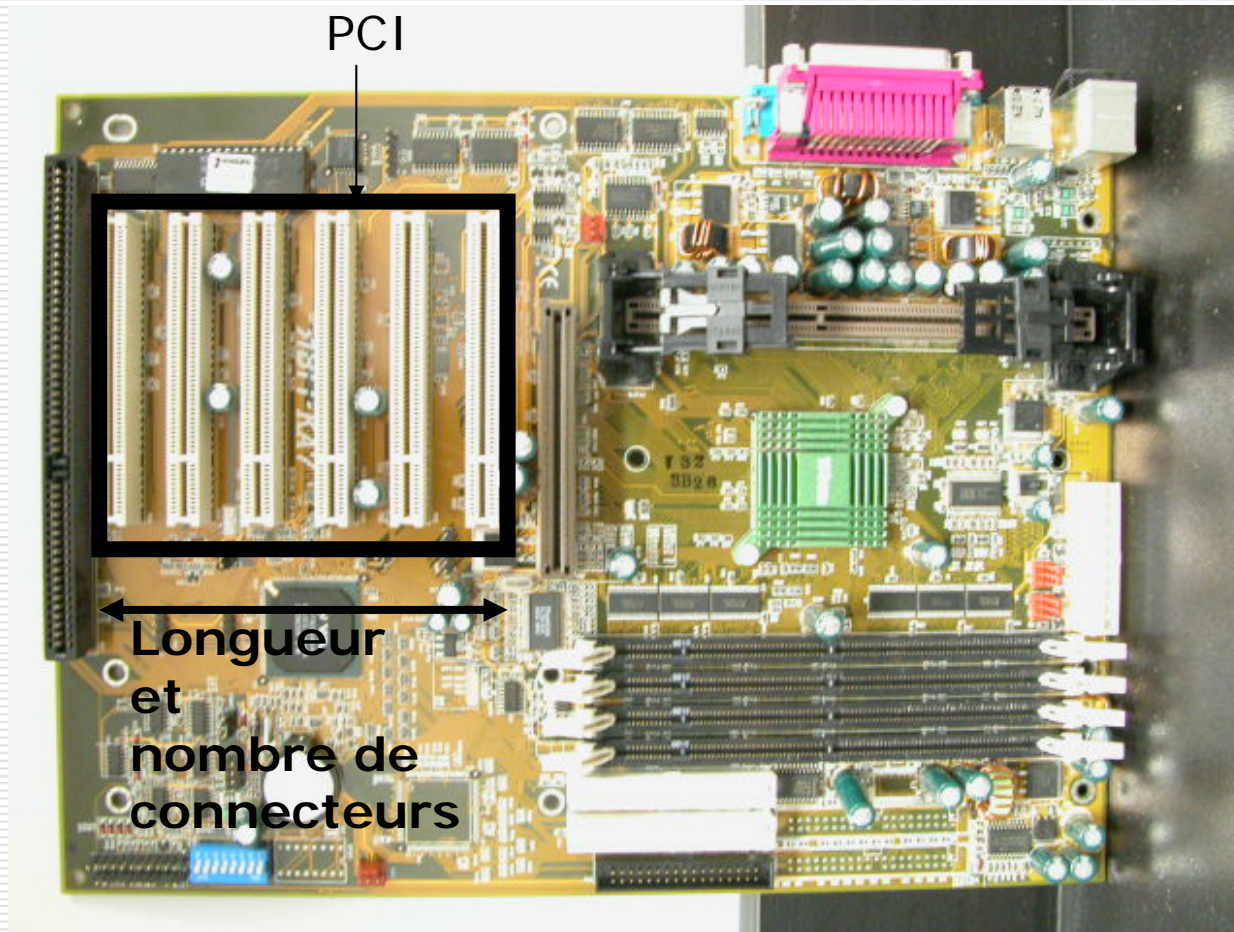
Écriture sur un bus asynchrone



Exemples de bus parallèles

	IDE	SCSI	PCI	PCI-X (eXtended)
Largeur DATA	16 bits	8 ou 16 bits	32 ou 64 bits	32 ou 64 bits
Nbre de maîtres	1	Plusieurs	Plusieurs	Plusieurs
Clocking	Async.	Async.	Sync.	Sync.

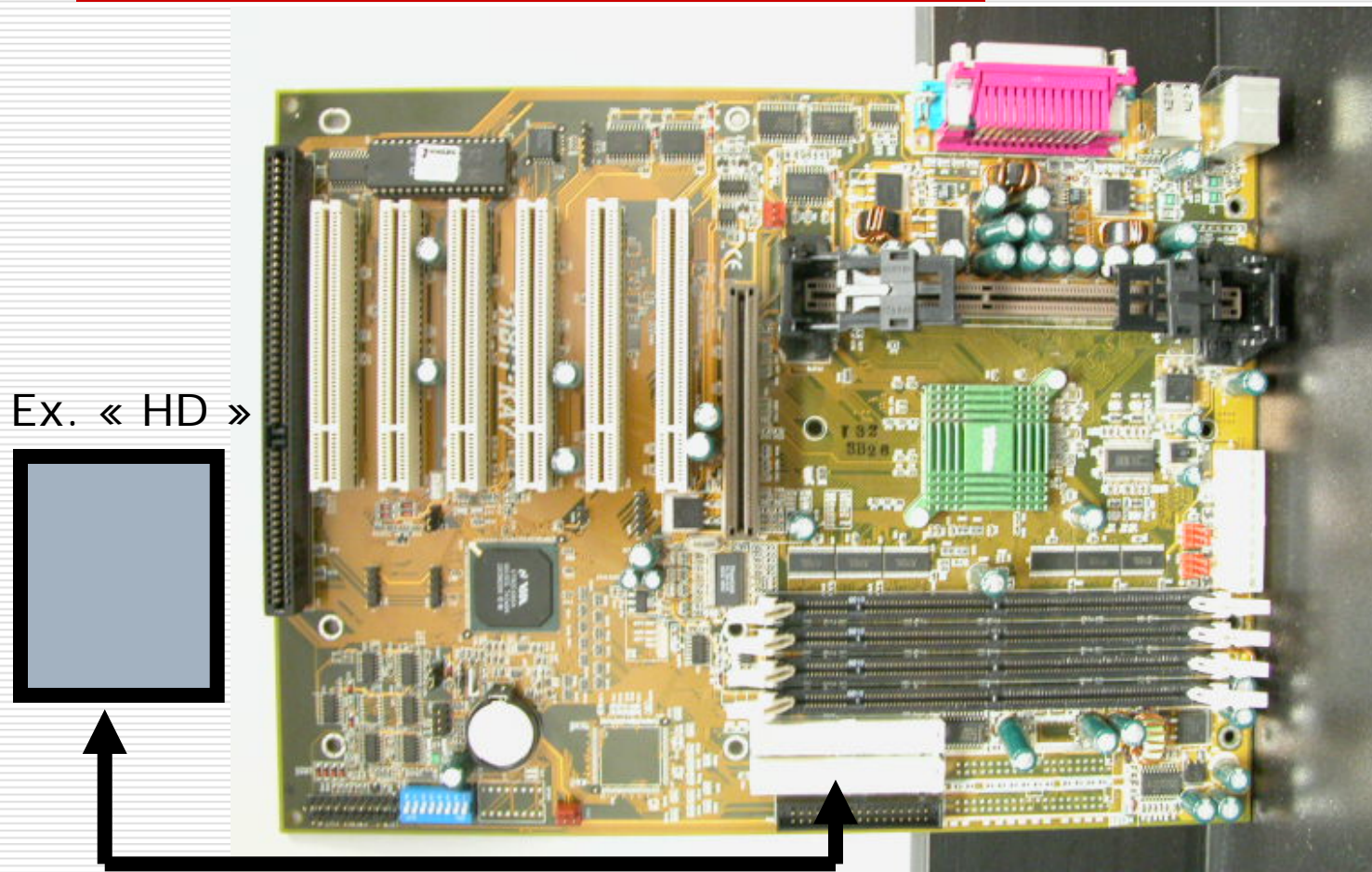
Bus PCI - Synchrone



Exemples de bus parallèles

	IDE/ ATA	SCSI	PCI	PCI-X (eXtended)
Largeur DATA	16 bits	8 ou 16 bits	32 ou 64 bits	32 ou 64 bits
Nbre de maîtres	1	Plusieurs	Plusieurs	Plusieurs
Clocking	Async.	Async.	Sync.	Sync.

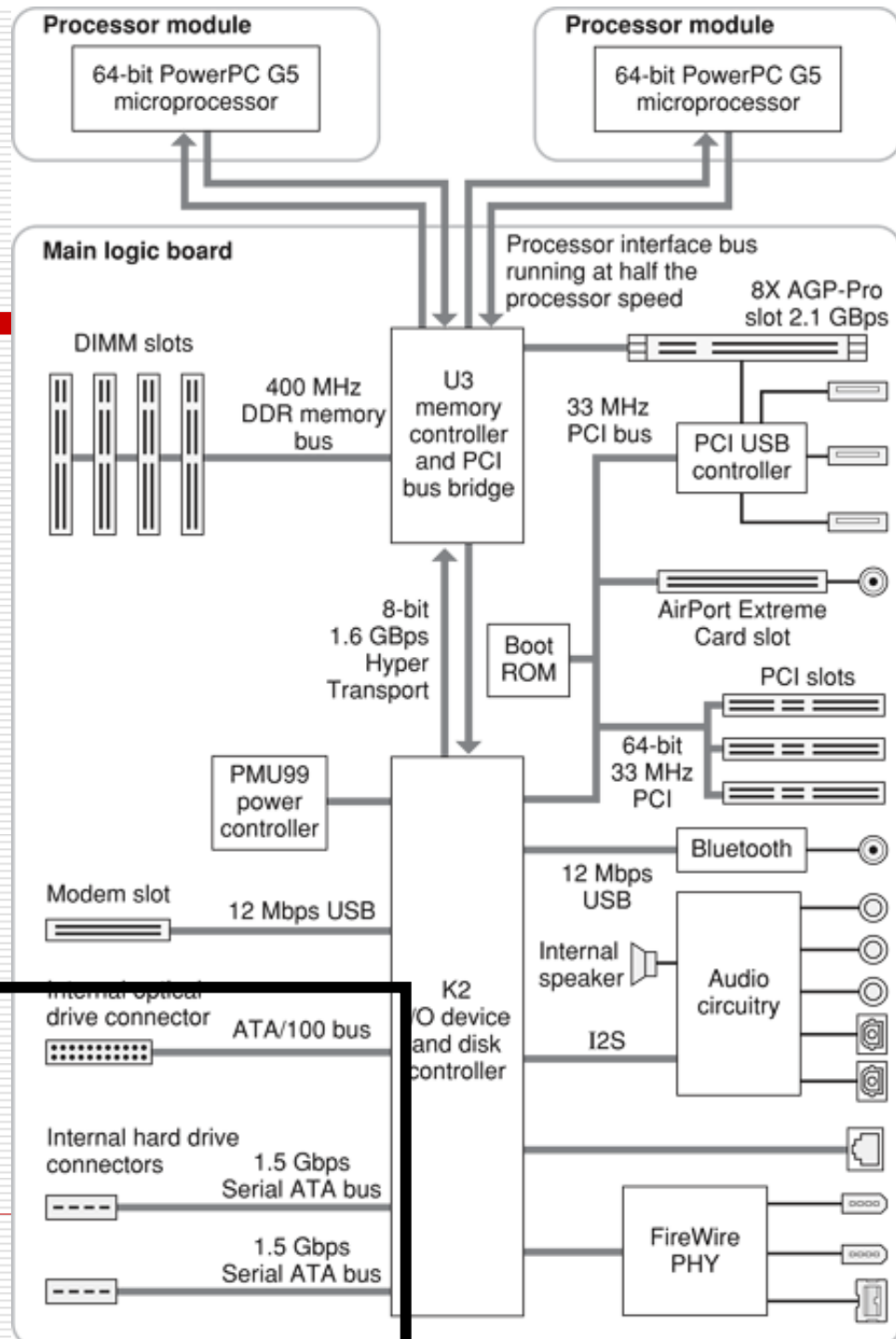
IDE/SCSI – asynchrone (ex. disque magnétique)



Ex. « HD »

Sylvain Martel - INF6500

Relativement longue distance



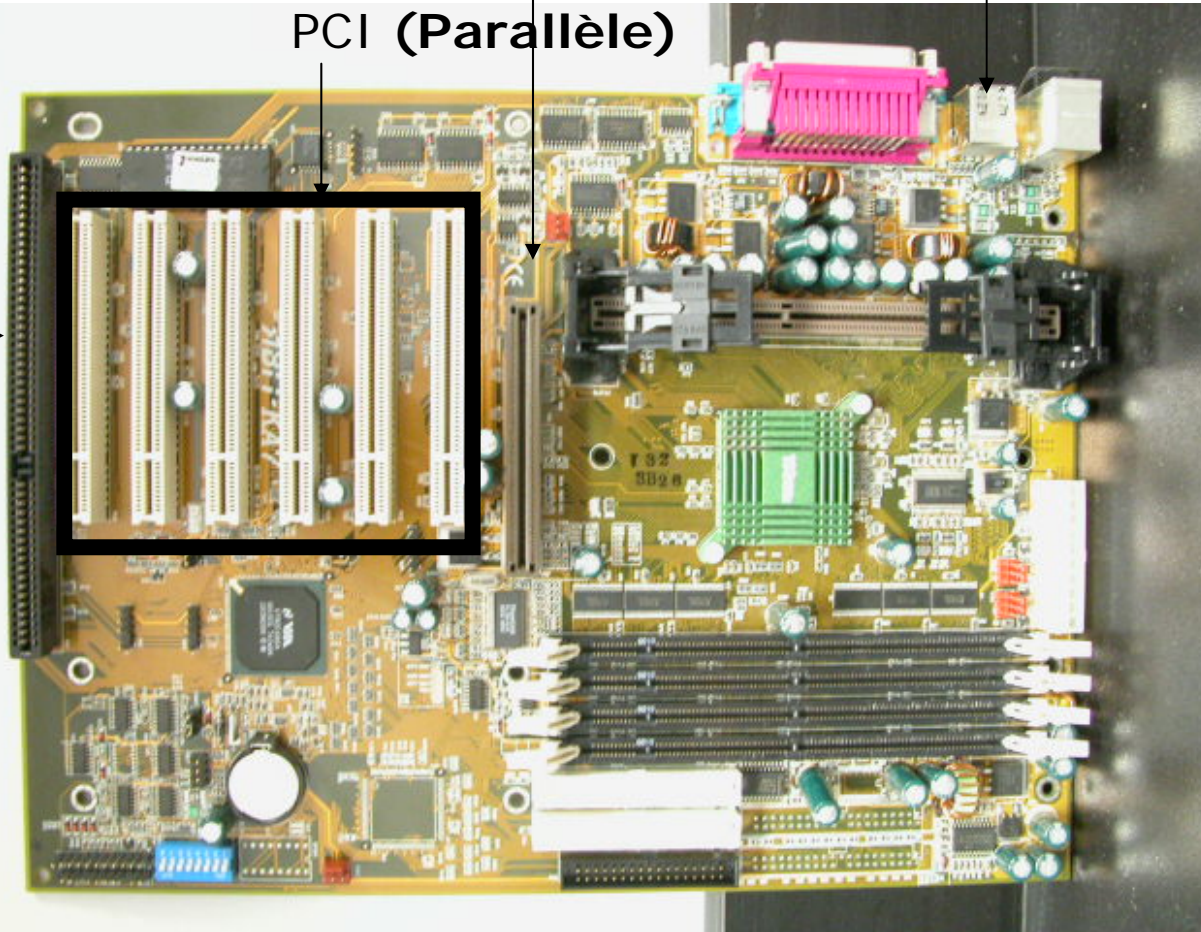
Bus série

AGP (Parallèle)

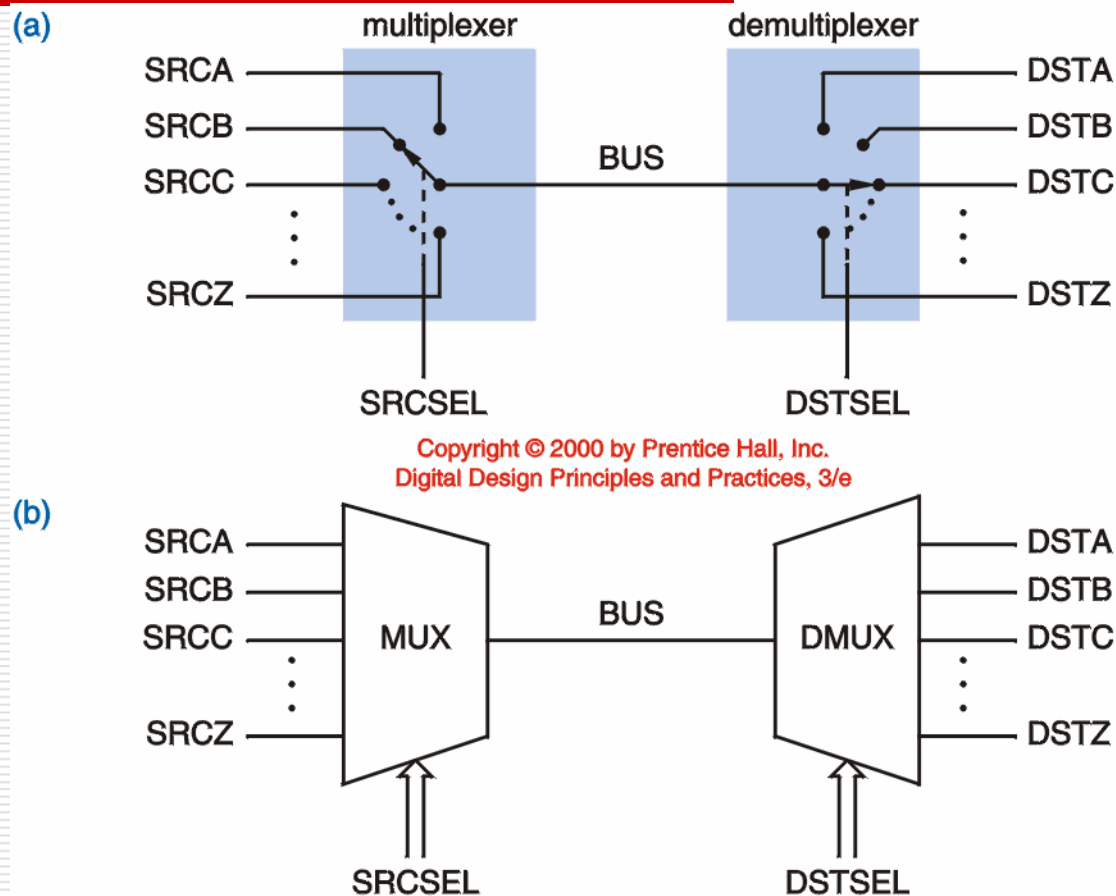
USB (Universal Serial Bus)
(Série)

PCI (Parallèle)

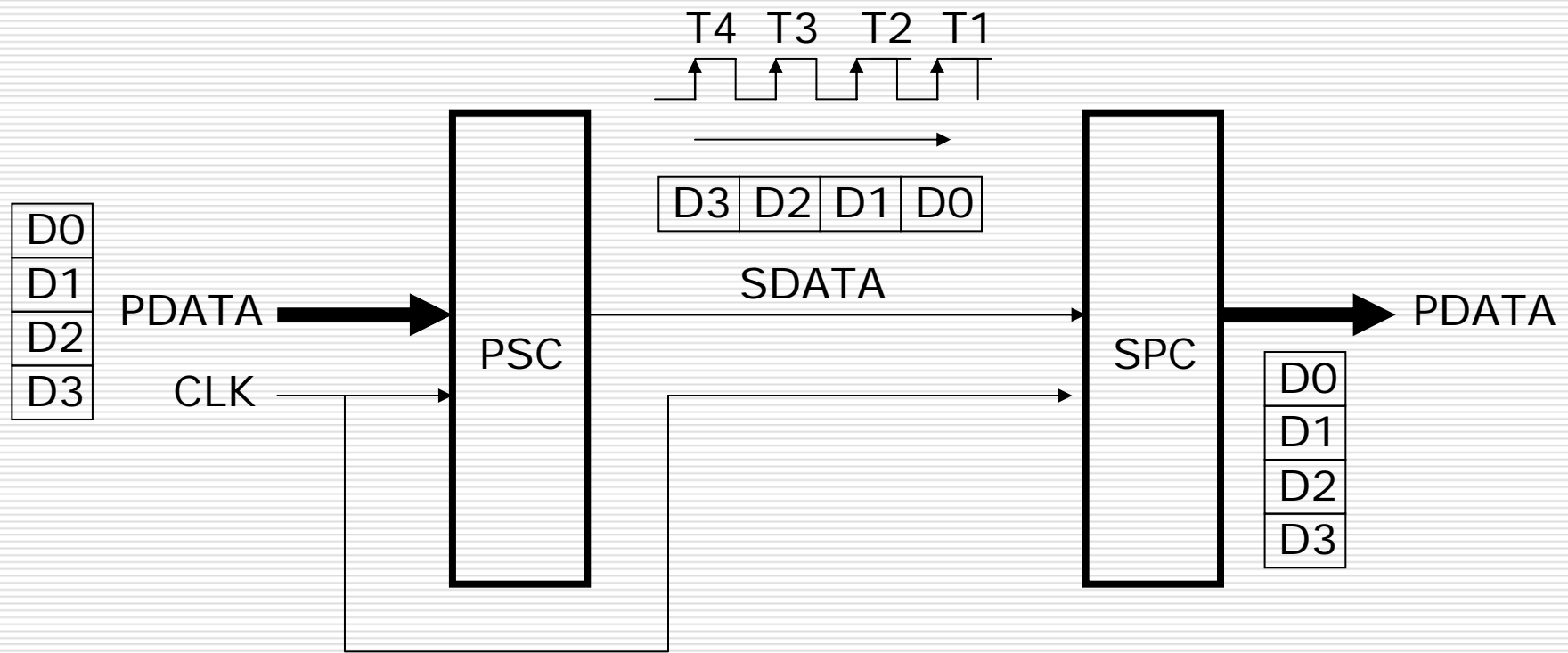
ISA (Parallèle)



Multiplexeur (MUX/DMUX) – Exemple d'application – Bus série

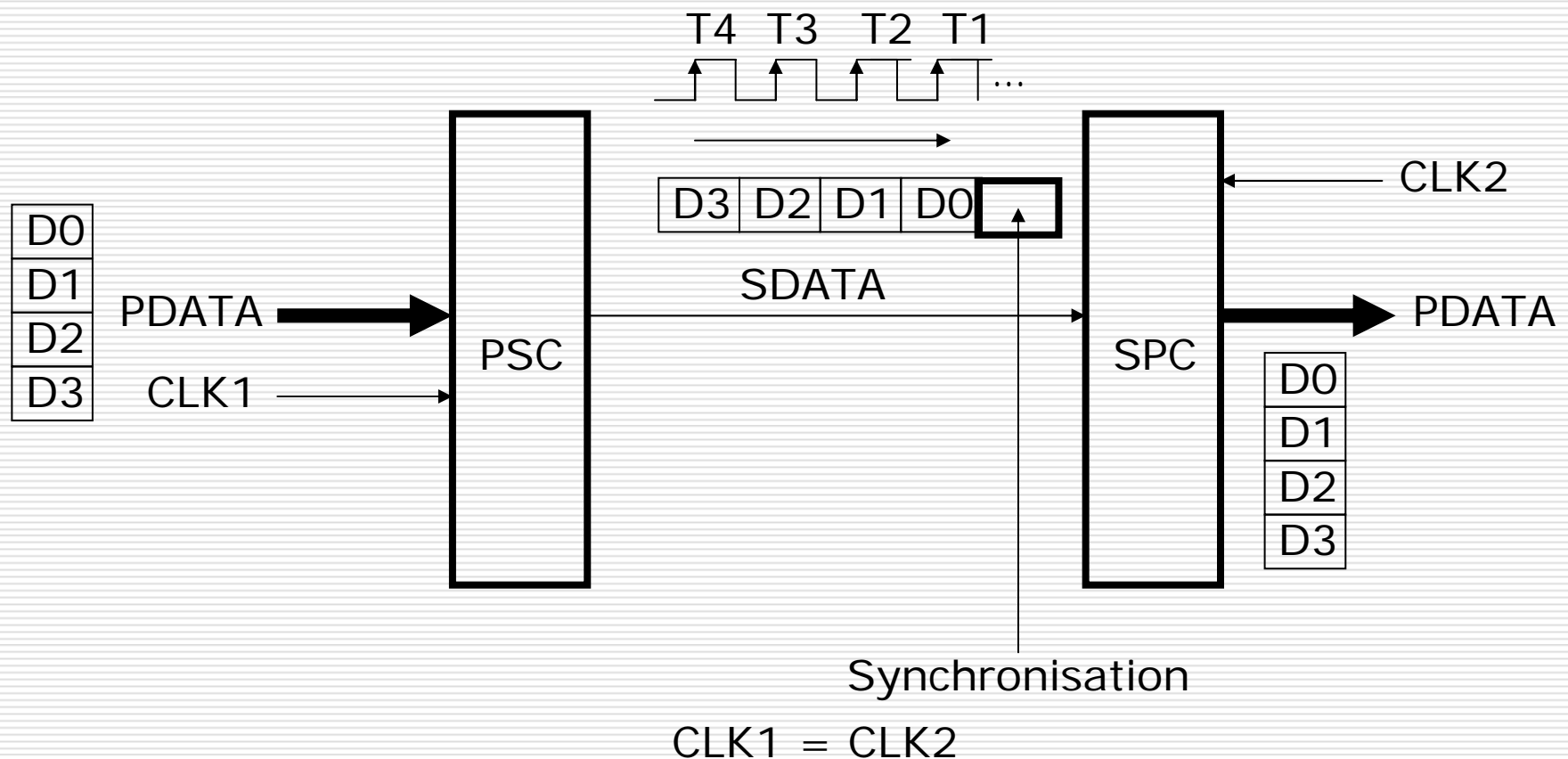


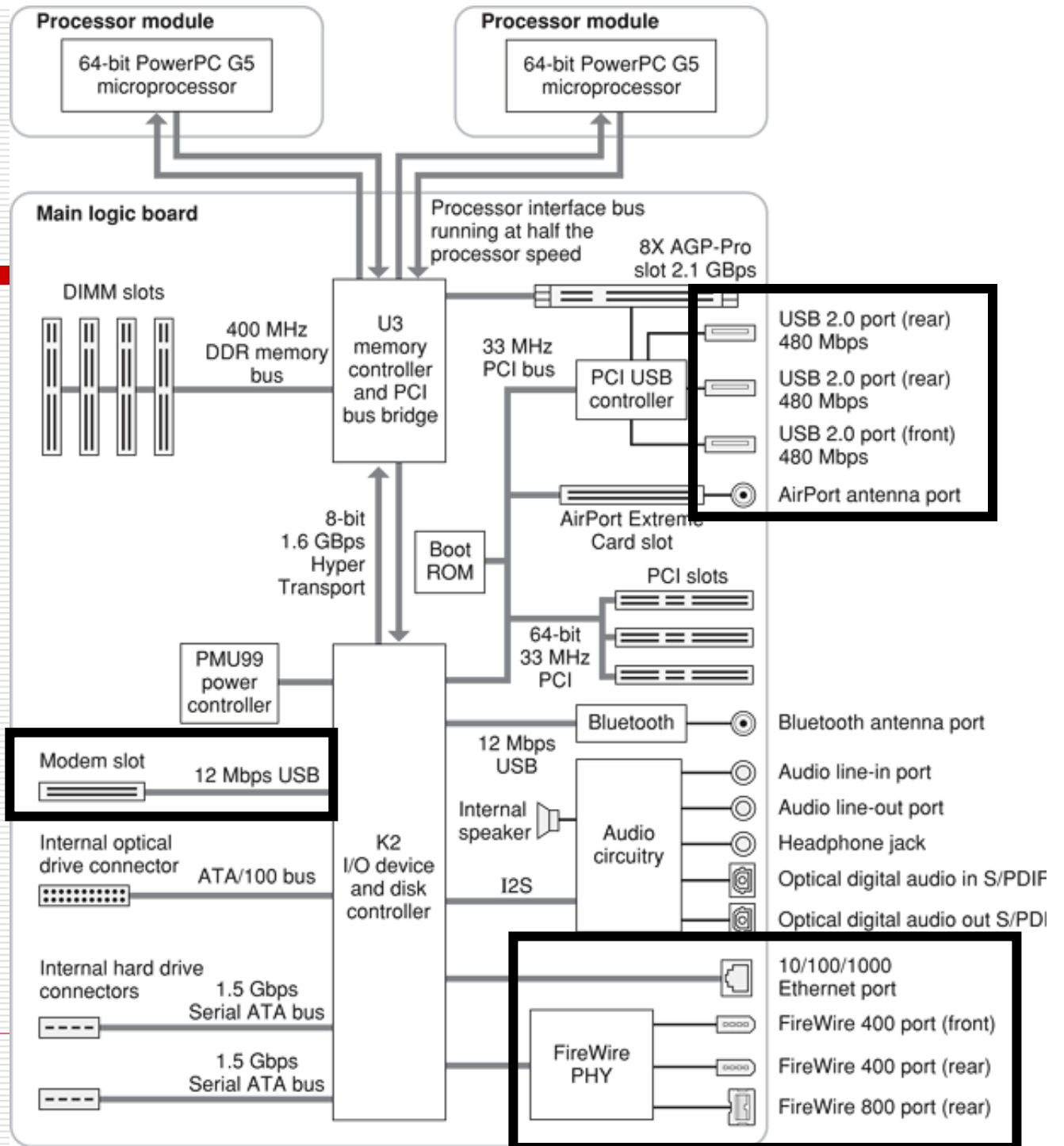
Multiplexeur (MUX/DMUX) – Exemple d'application – Bus série (2-wires)



Note: PDATA: Parallèle DATA, SDATA: Serial DATA,
PSC: Parallel-Serial Converter, SPC: Serial-Parallel Converter

Multiplexeur (MUX/DMUX) – Exemple d'application – Bus série (1-wire)





Exemples de bus séries

	I2C (2 signal wires)	1-wire (1 signal wire)	RS-232 (9 ou 25 signal wires)	SPI (3 signal wires)
Largeur DATA	1 bit	1 bit	2 bits	1 bit
Nbre de maîtres	Plusieurs	Plusieurs	Plusieurs	Plusieurs
Clocking	Async.	Async.	Async.	Async.

Bus séries ou parallèles

Série: moins de lignes	I2C	1-wire	RS-232	SPI
Bande passante (MAX)	0.4-3.4 Mbits/s (Mb/s)	0.014 Mb/s	0.192 Mb/s	1 Mb/s
Parallèle: plus de lignes	IDE/ ATA	SCSI	PCI	PCI-X
Bande passante (MAX)	200 MBytes/s (MB/s)	320 MB/s	533 MB/s	1066 MB/s

Bus séries **rapides** ou parallèles

□ Exemple:

- USB 2.0 = 480 Mb/s ou Mbps est équivalent à $480/8 = 60$ MB/s ou MBps << 200 MB/s IDE (un des plus lents bus parallèles)

Bus mémoire et bus E/S

- L'interface des composants de stockage avec l'UCT.
 - Le bus E/S peut être connecté à la mémoire ou au cache
 - Dans le cas où il est connecté à la mémoire:
 - Une commande E/S sur le bus peut interférer avec une autre lecture d'instruction par l'UCT.
 - La méthode la plus courante d'adressage d'un composant d'E/S par l'UCT est nommée 'Les E/S mappées en mémoire'
 - Des parties de l'espace adressable sont affectées aux composants d'E/S

Bus mémoire et bus E/S (suite)

- Une autre méthode consiste à utiliser des codes opérations dédiés aux E/S dans l'UCT.
 - L'UCT envoie un signal indiquant qu'une adresse est pour des composants E/S.
- Quelle que soit la technique d'adressage choisie, chaque composant E/S a des registres pour fournir des informations d'état et de contrôle.
- Ces registres d'état sont continuellement testés par l'UC pour savoir si la prochaine opération d'E/S est prête. C'est l'interrogation (*polling*).
- Cette interrogation gaspille une grande quantité de temps UC.
- Pour cette raison, les architectures modernes utilisent un système de d'interruptions qui permet de décharger l'UC, lui donnant plus de temps pour l'exécution des processus. Cette technique est la clé des systèmes d'exploitation multitâches.

Choix - Bus

- ❑ La vitesse maximale d'un bus est limitée surtout par des facteurs physiques: la longueur du bus et le nombre de composants
- ❑ Ces limites physiques empêchent d'accélérer à volonté le bus.
- ❑ Dans le pire scénario, N composants voulant communiquer avec tous les autres, nécessiteront $N^2/2$ liens.

Bus Type	Bus Width	Bus Speed	MB/sec			
ISA	16 bits	8 MHz	16 MBps			
EISA	32 bits	8 MHz	32 MBps			
VL-bus	32 bits	25 MHz	100 MBps			
VL-bus	32 bits	33 MHz	132 MBps			
PCI	32 bits	33 MHz	132 MBps			
PCI	64 bits	33 MHz	264 MBps			
PCI	64 bits	66 MHz	512 MBps			
PCI	64 bits	133 MHz	1GBps			

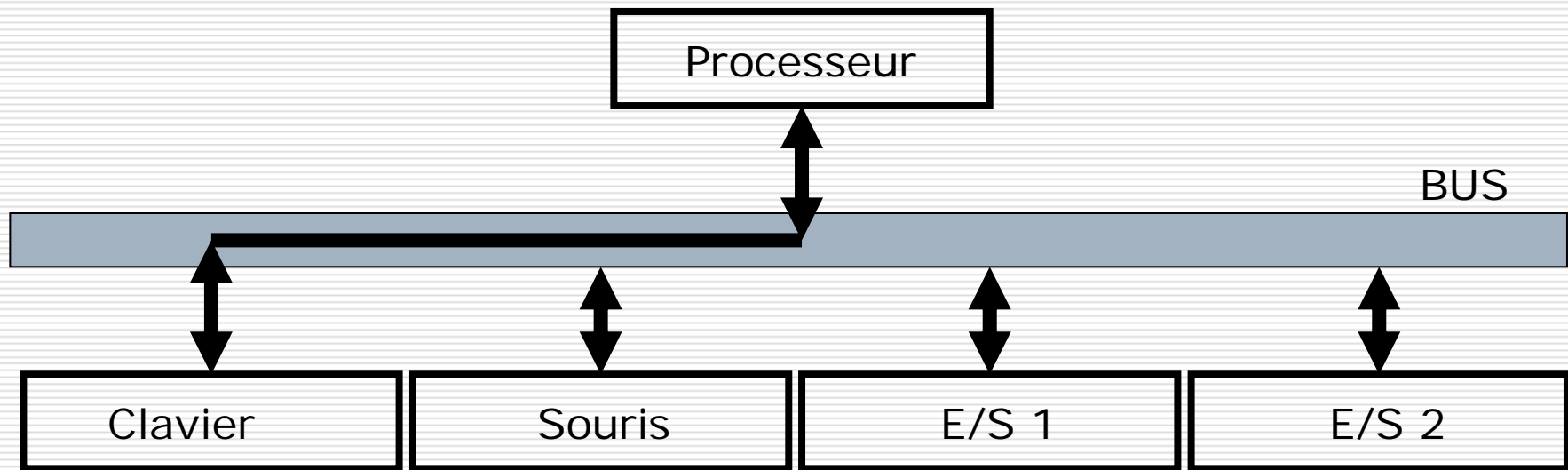
Choix – Bus

Nom	Specification	# de composants	Largeur du bus	Vitesse du bus	Mo/s
Asynchrone SCSI	SCSI-1	8	8 bits	5 MHz	4 Mo/s
Synchrone SCSI	SCSI-1	8	8 bits	5 MHz	5 Mo/s
Wide SCSI	SCSI-2	16	16 bits	5 MHz	10 Mo/s
Fast SCSI	SCSI-2	8	8 bits	10 MHz	10 Mo/s
Fast/Wide SCSI	SCSI-2	16	16 bits	10 MHz	20 Mo/s
Ultra SCSI	SCSI-3 SPI	8	8 bits	20 MHz	20 Mo/s
Ultra/Wide SCSI	SCSI-3 SPI	8	16 bits	20 MHz	40 Mo/s
Ultra2 SCSI	SCSI-3 SPI-2	8	8 bits	40 MHz	40 Mo/s
Ultra2/Wide SCSI	SCSI-3 SPI-2	16	16 bits	40 MHz	80 Mo/s
Ultra3 SCSI	SCSI-3 SPI-3	16	16 bits	40 MHz	160 Mo/s

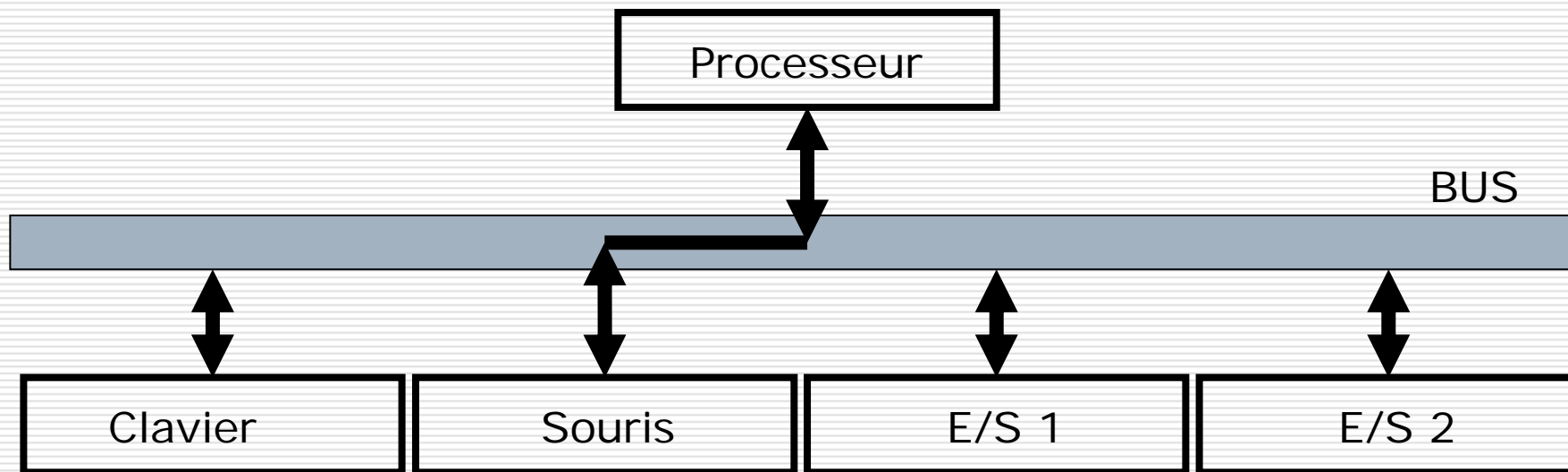
Polling ou interruption

- Quelle que soit la technique d'adressage choisie, chaque composant E/S a des registres pour fournir des informations d'état et de contrôle.
- Ces registres d'état sont continuellement testés par l'UCT pour savoir si la prochaine opération d'E/S est prête. C'est l'interrogation (*polling*).
- Cette interrogation gaspille une grande quantité de temps UCT.
- Pour cette raison, les architectures modernes utilisent un système de d'interruptions qui permet de décharger l'UCT, lui donnant plus de temps pour l'exécution des processus. Cette technique est la clé des systèmes d'exploitation multitâches.
- Cette technique provoque cependant un surcoût pour les OS. Dans certains cas où il y a des centaines d'opérations d'E/S par seconde (temps réel), cela devient insupportable pour le processeur (et OS)

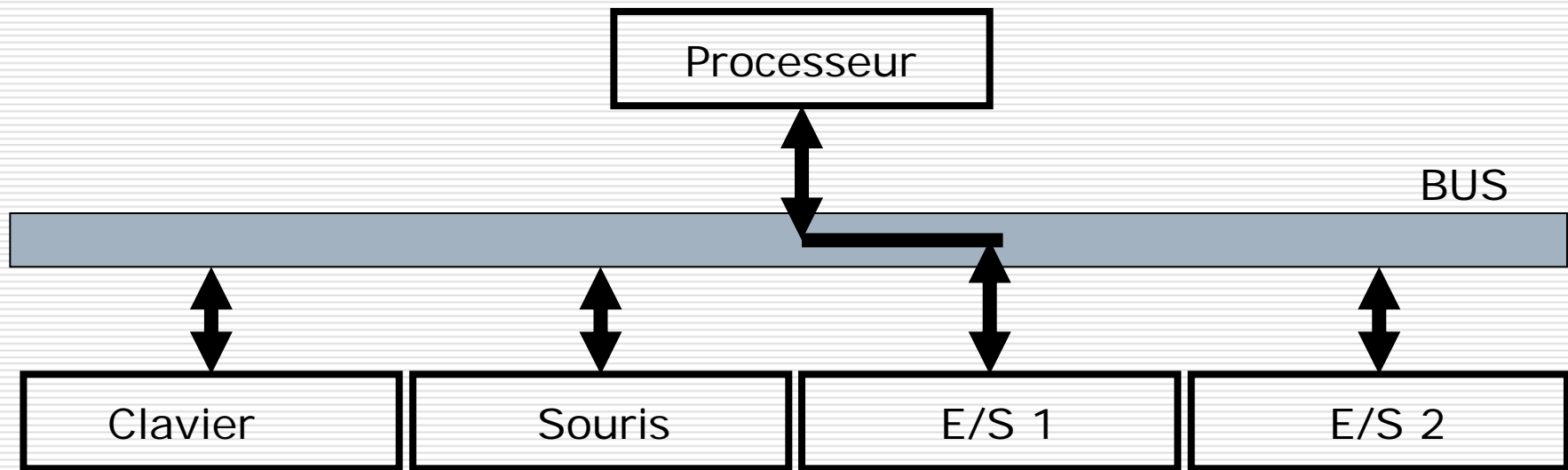
Polling - T1



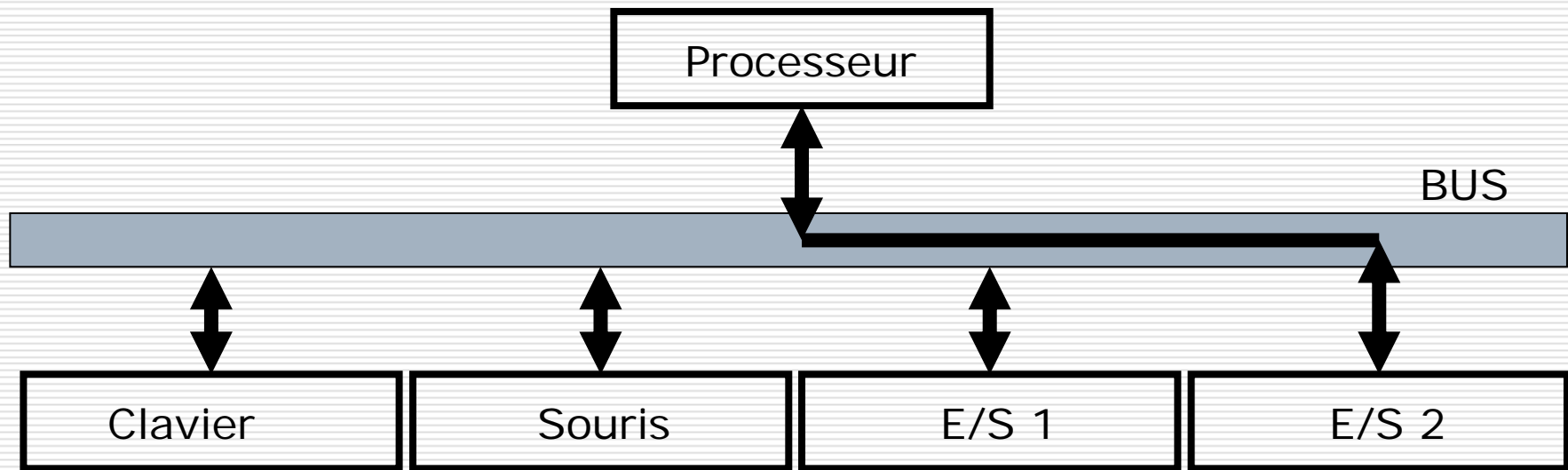
Polling – T2



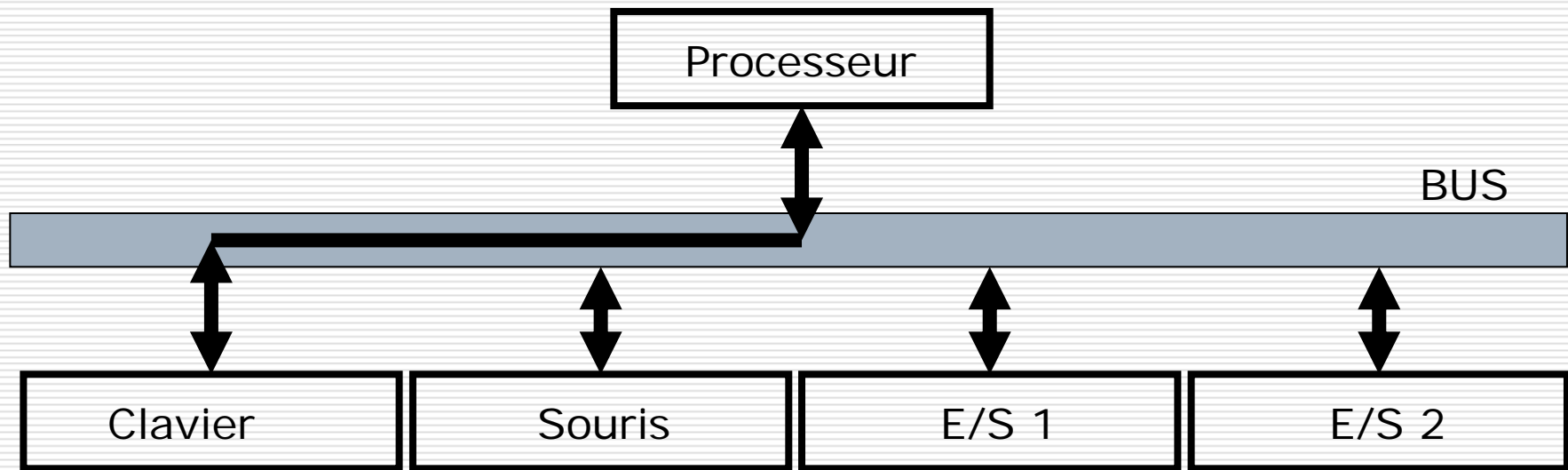
Polling – T3



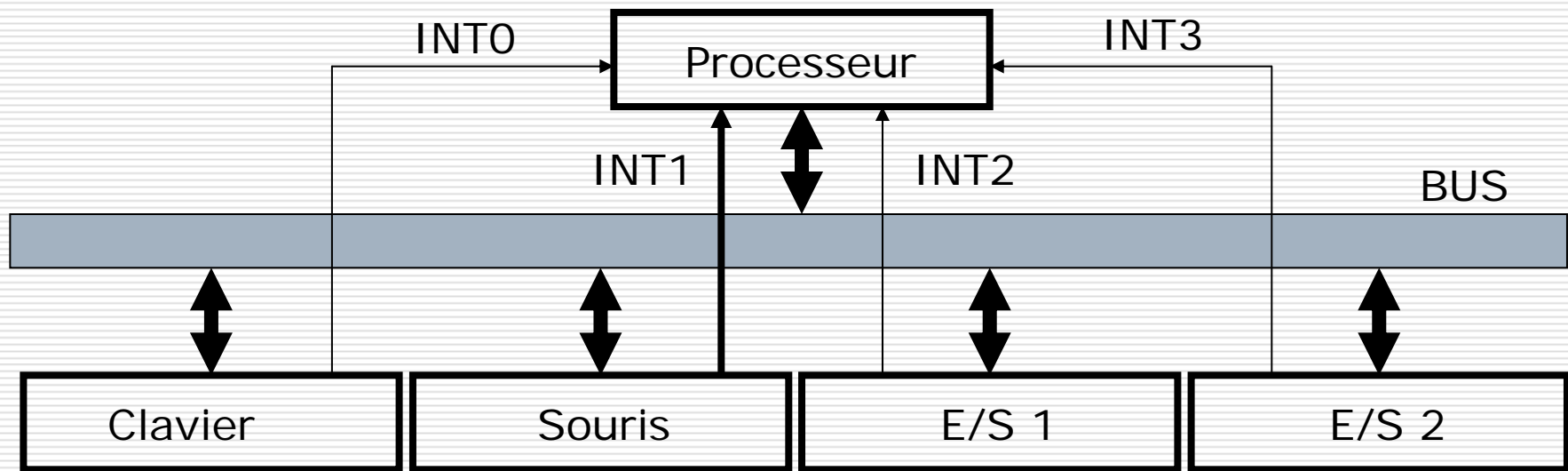
Polling – T4



Polling – T5, ...



Interruption



Polling vs. Interruption

□ Polling:

- Latence (temps de réponse) courte avec faible nombre de périphériques
- Beaucoup de transactions et overhead pour obtenir une latence relativement courte

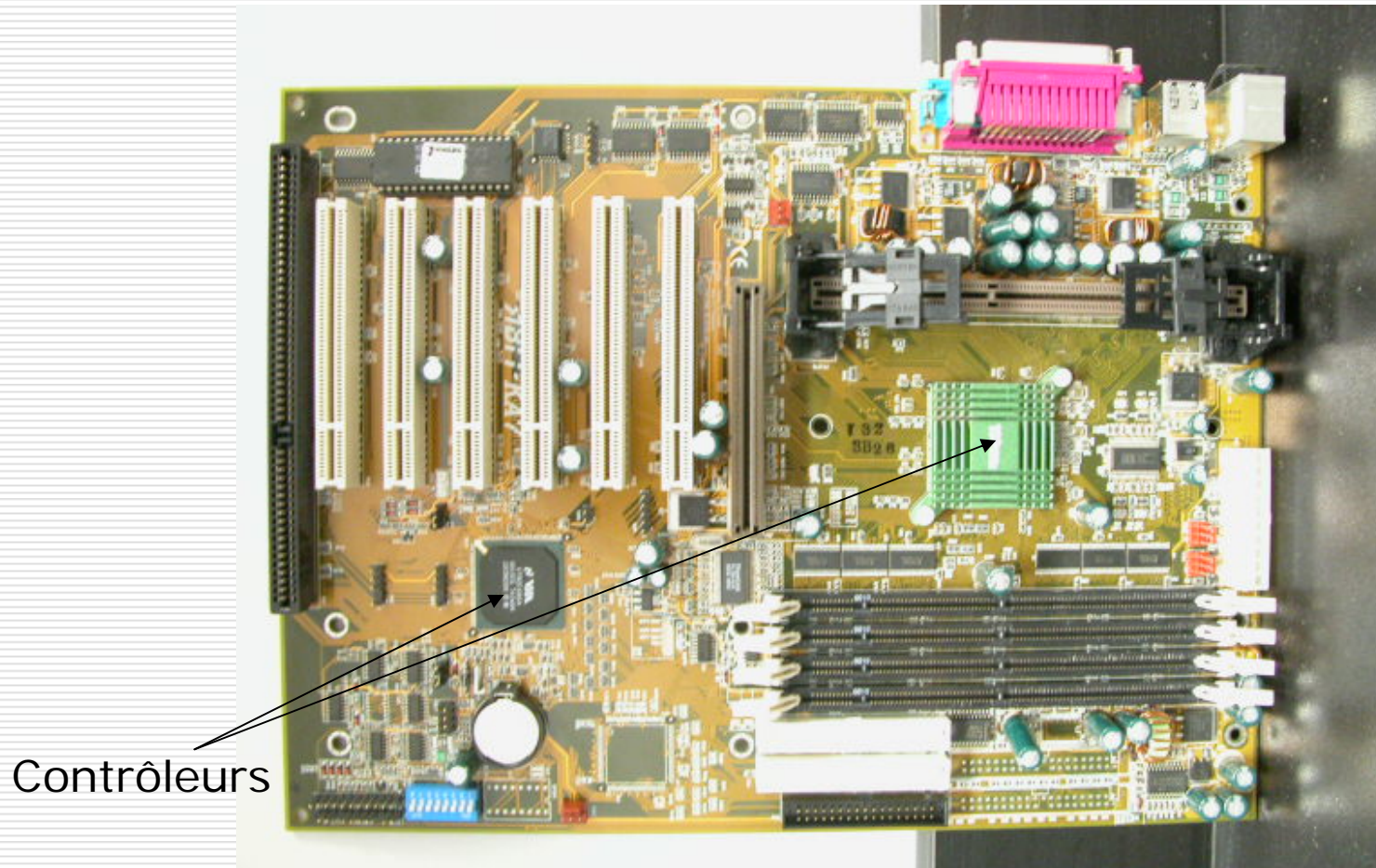
□ Interruption:

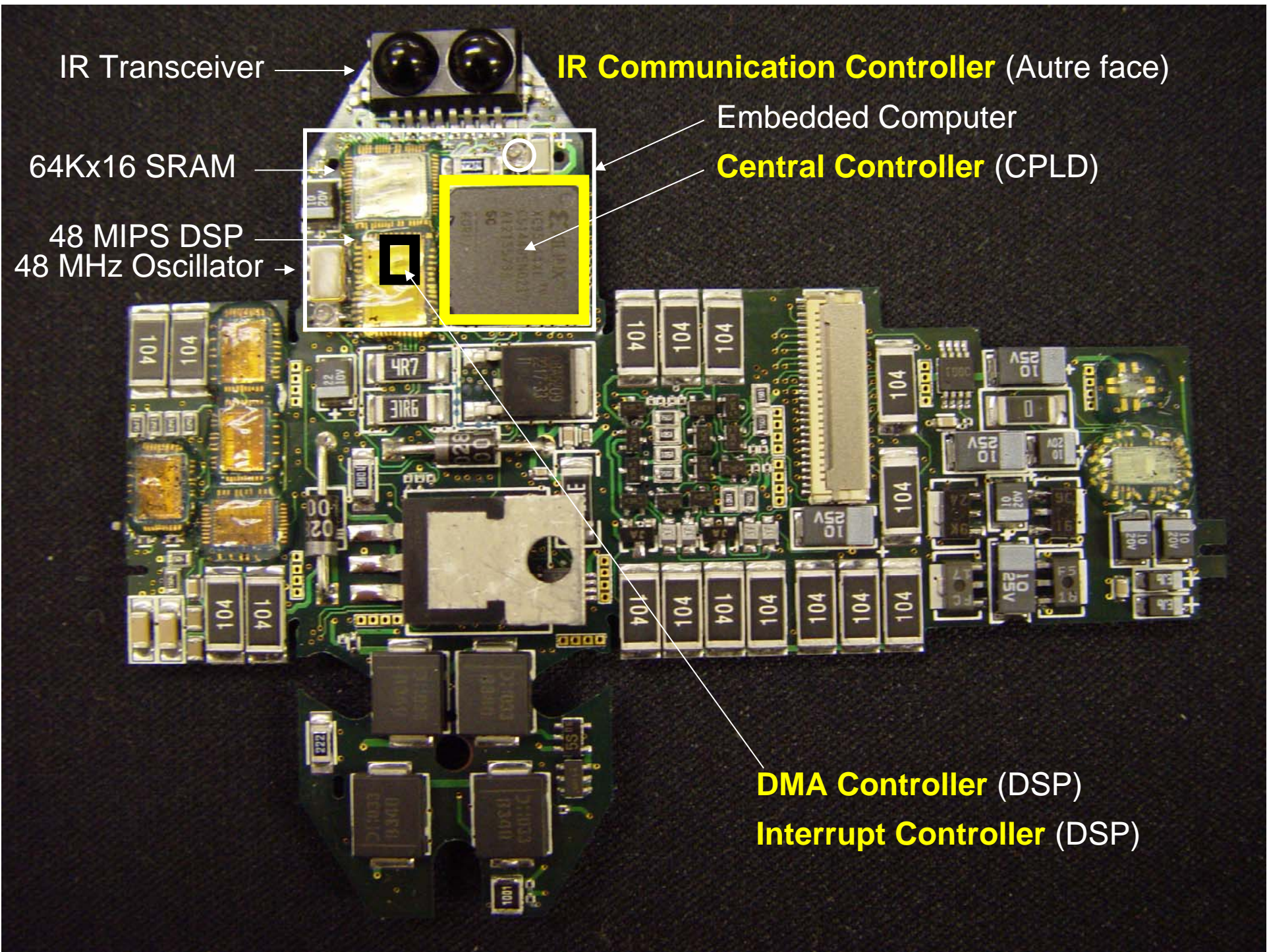
- Latence (temps de réponse) plus longue (overhead du context switch, avant d'exécuter l'ISR (Interrupt Service Routine))
- Latence plus courte que polling si nombre de périphériques est relativement élevé

Contrôleurs.

- Retirer à l'UCT la responsabilité des E/S.
 - Les interruptions libèrent l'UC de l'attente de chaque événement d'E/S mais il y a encore beaucoup de cycles UCT passés en transferts de données.

Contrôleurs





IR Transceiver →

IR Communication Controller (Autre face)

Embedded Computer

64Kx16 SRAM →

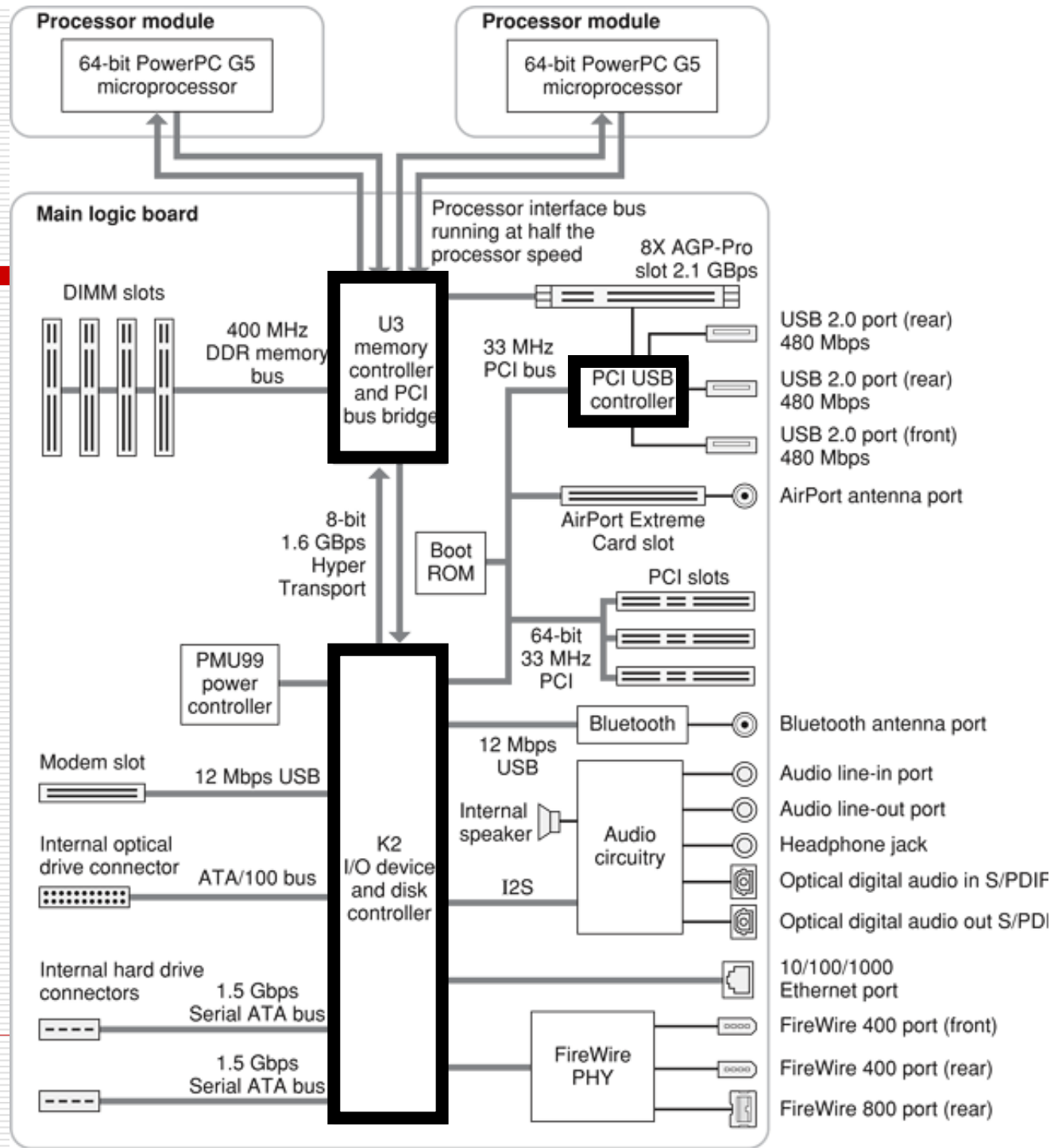
Central Controller (CPLD)

48 MIPS DSP →

48 MHz Oscillator →

DMA Controller (DSP)

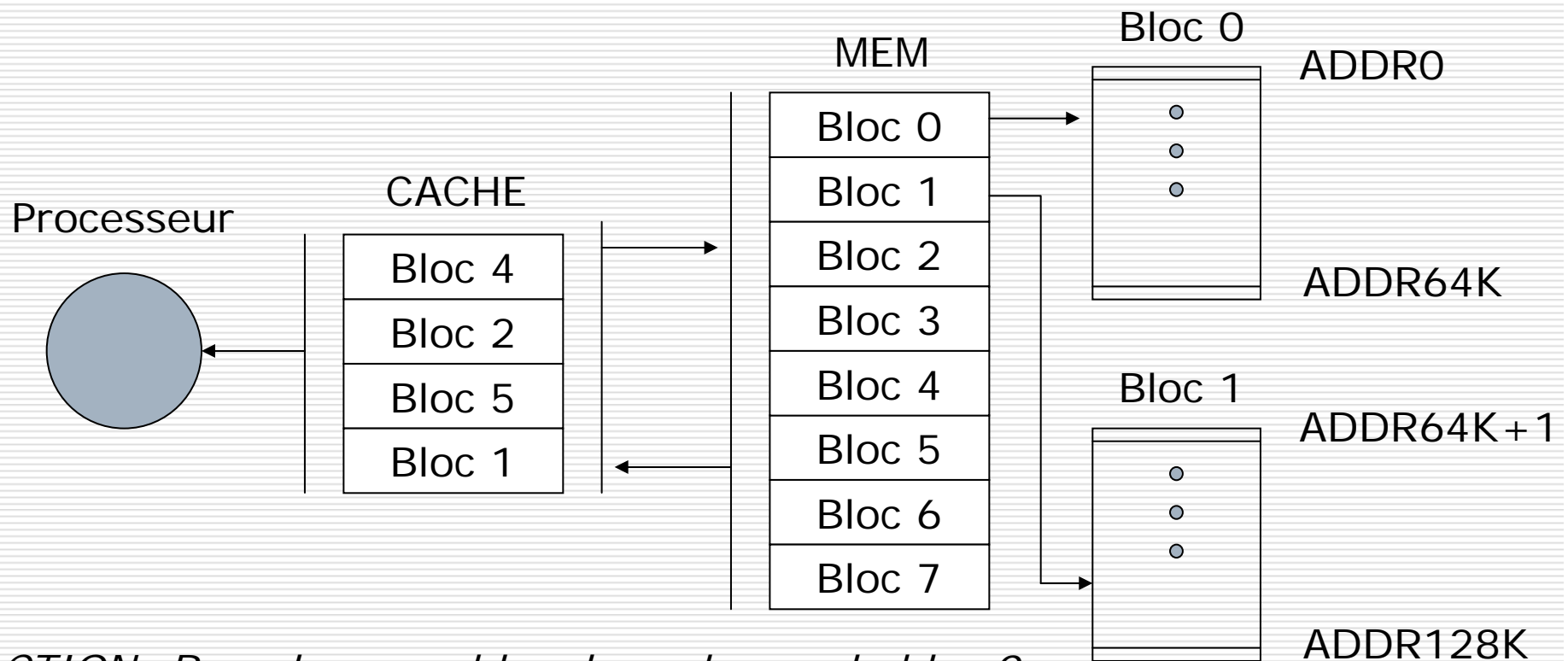
Interrupt Controller (DSP)



Contrôleur DMA

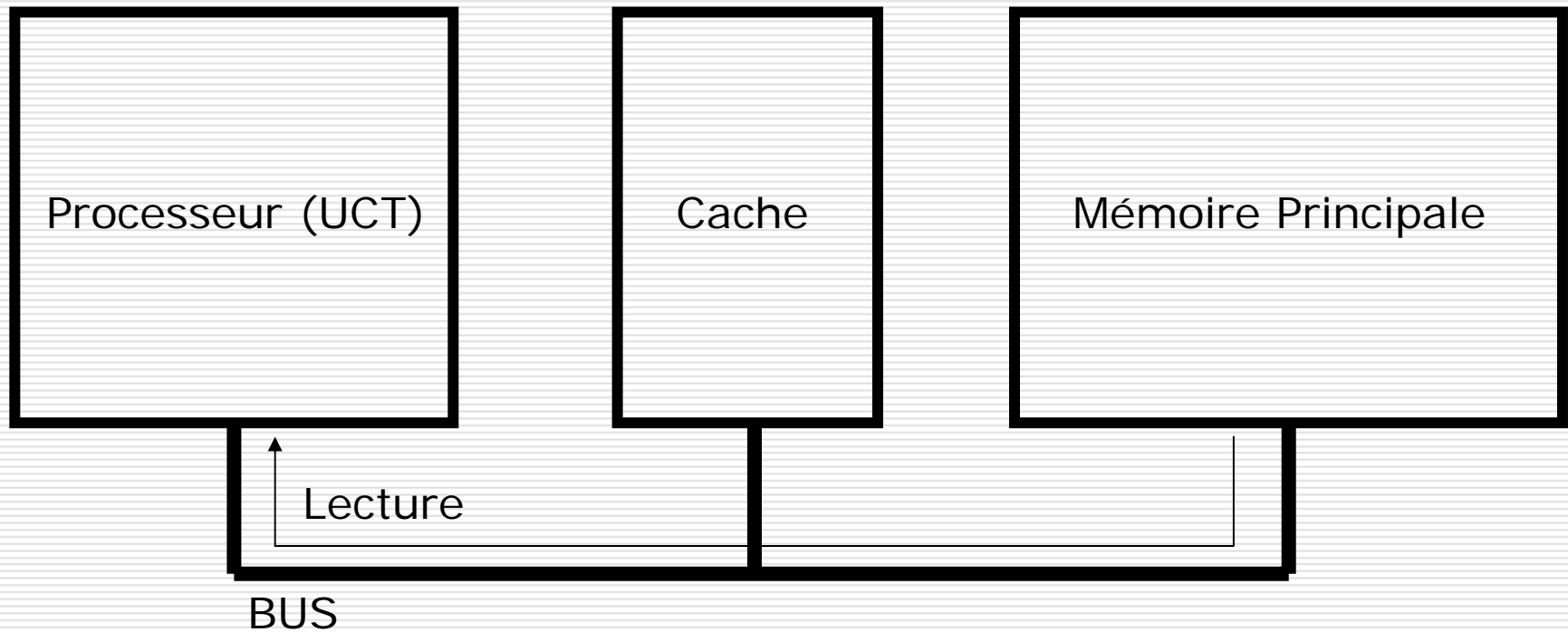
- Retirer à l'UCT la responsabilité des E/S.
 - Des dispositifs d'accès direct mémoire (*DMA*) sont ajoutés à beaucoup d'ordinateurs pour permettre des transferts d'un grand nombre de mots sans intervention de l'UCT.
 - Le DMA est un contrôleur spécialisé qui transfère des données entre la mémoire et/ou un composant E/S pendant que l'UCT s'occupe avec d'autres tâches.
 - S'il est à l'extérieur de l'UCT, doit agir comme un maître du bus.

Exemple: Transfert du bloc 0 entre la mémoire principale et la mémoire cache suite à un échec au cache

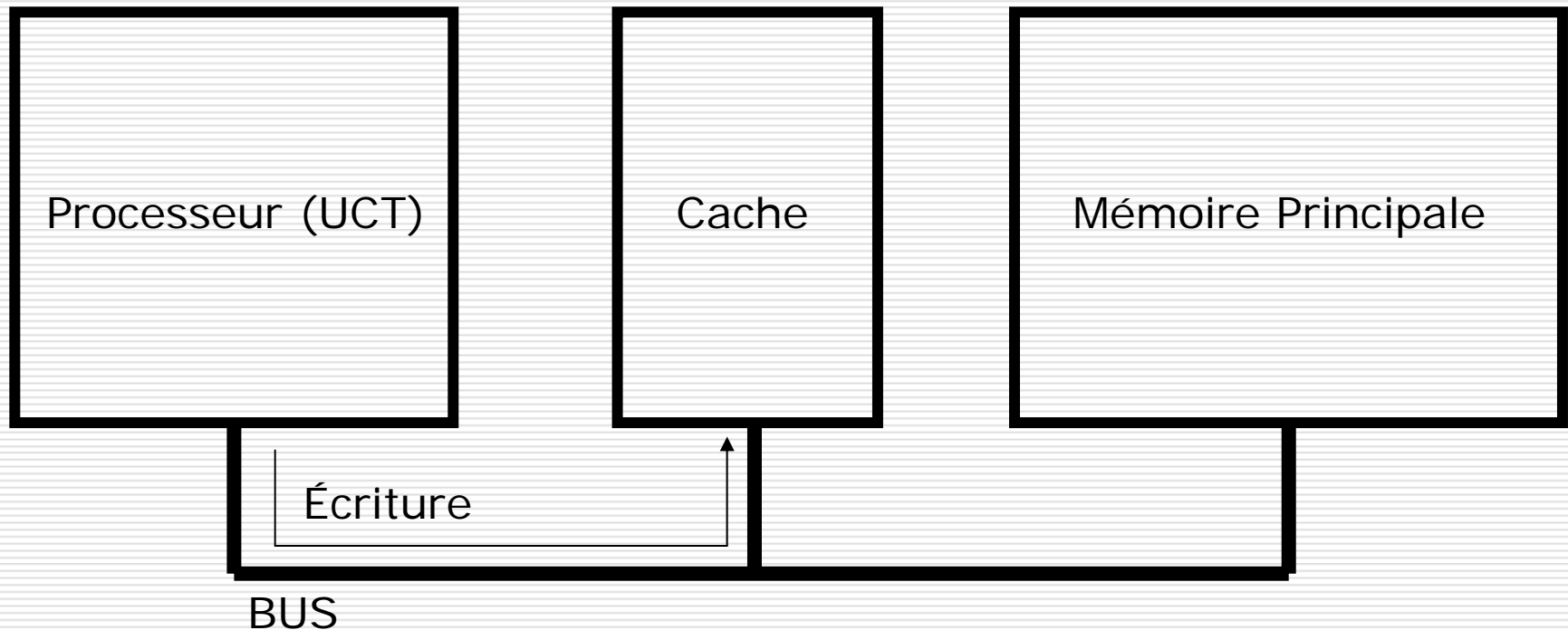


ACTION: Remplacer un bloc du cache par le bloc 0
Lequel: algorithme de remplacement LRU, Random, FIFO, etc...

Sans contrôleur DMA – T1

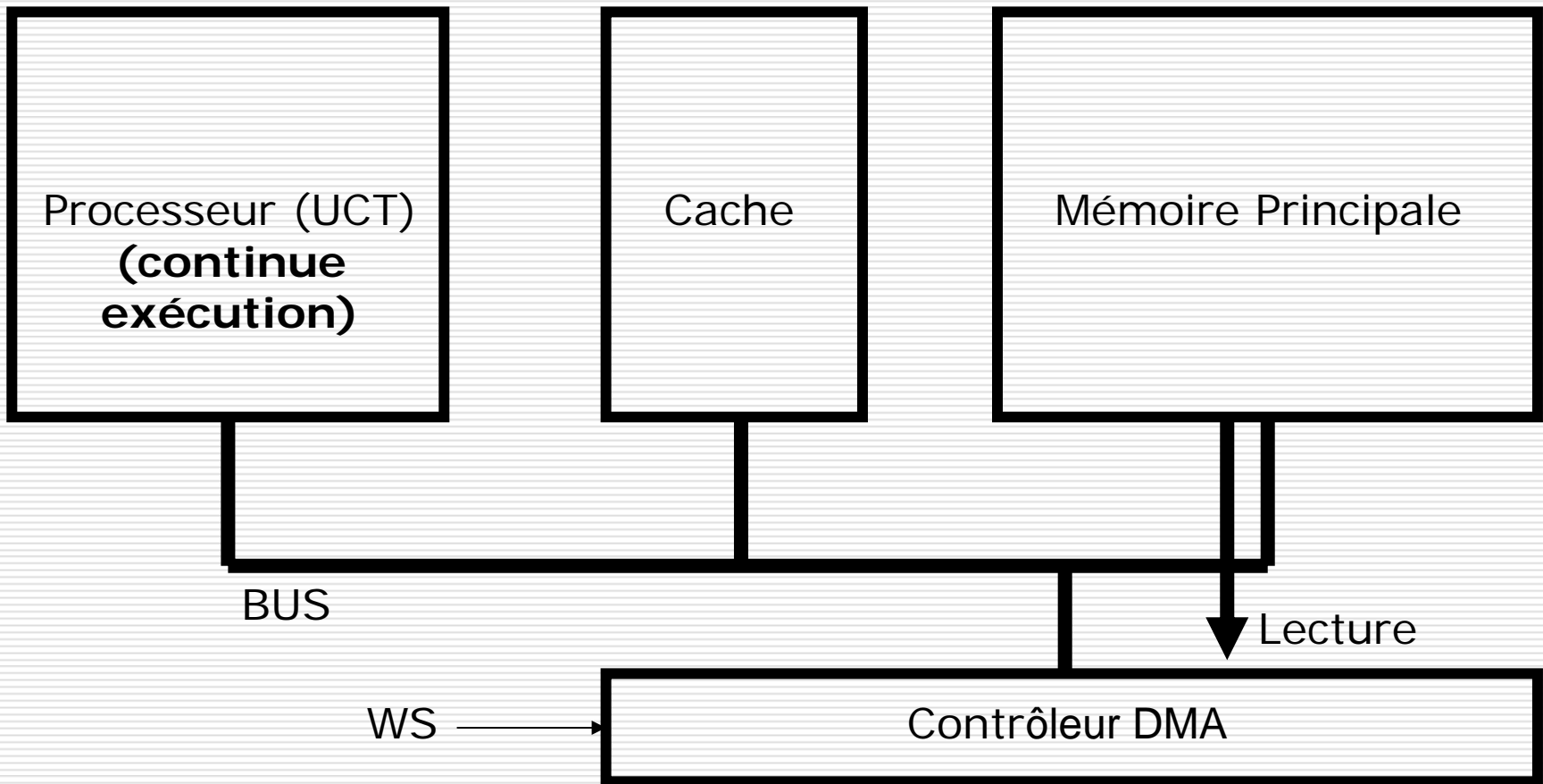


Sans contrôleur DMA – T2



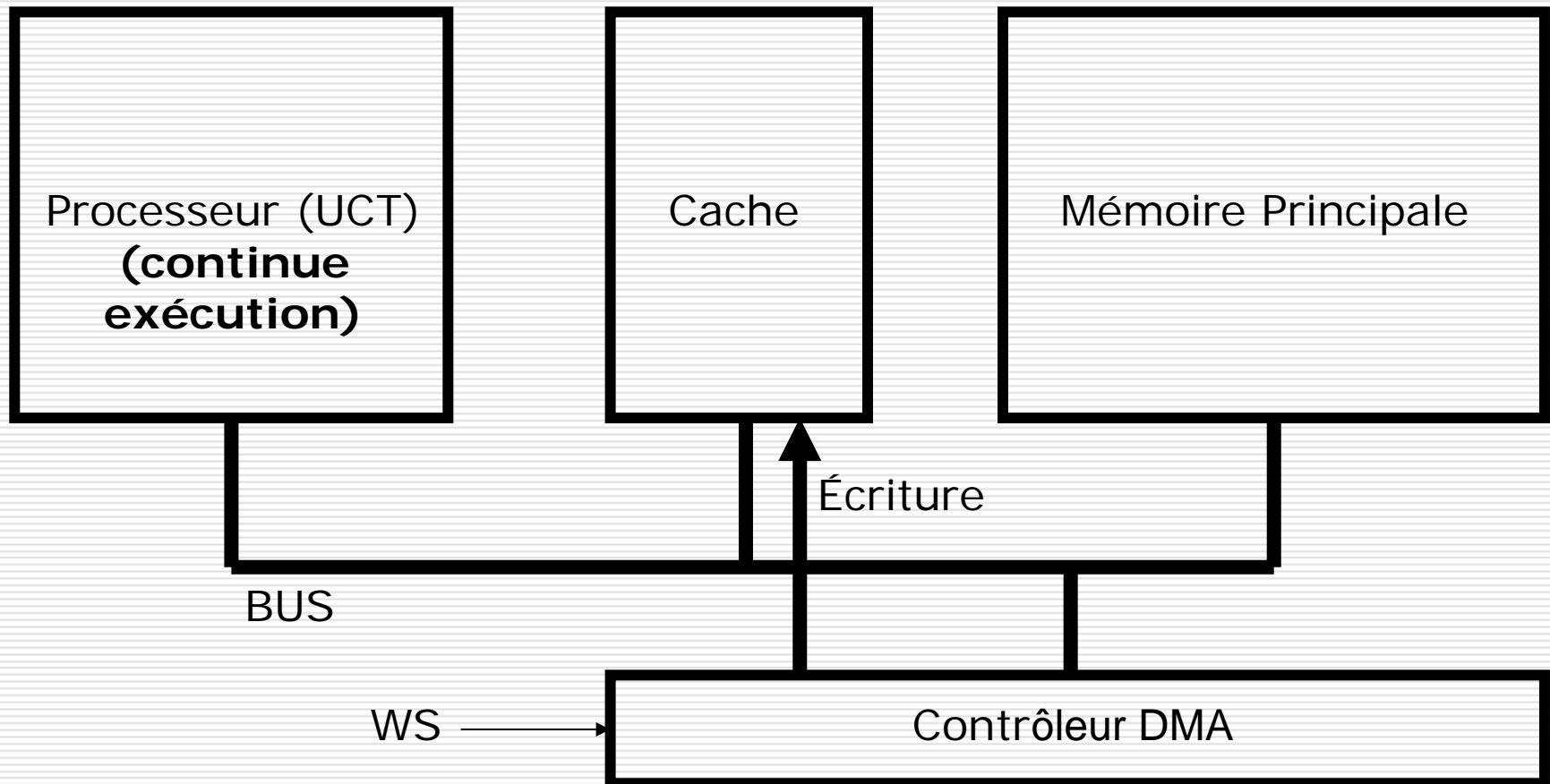
Note: Cycle T1 et T 2 (Lecture/Écriture) exécuté 65,536 fois par le processeur pour un bloc de 64K, une grande quantité de cycles non disponibles pour effectuer des tâches plus utiles)

Avec contrôleur DMA – T1



Note: Cycle T1 et T2 (Lecture/Écriture) exécuté 65,536 fois par le Contrôleur DMA pour un bloc de 64K, une grande quantité de cycles disponibles à l'UCT pour effectuer des tâches plus utiles)

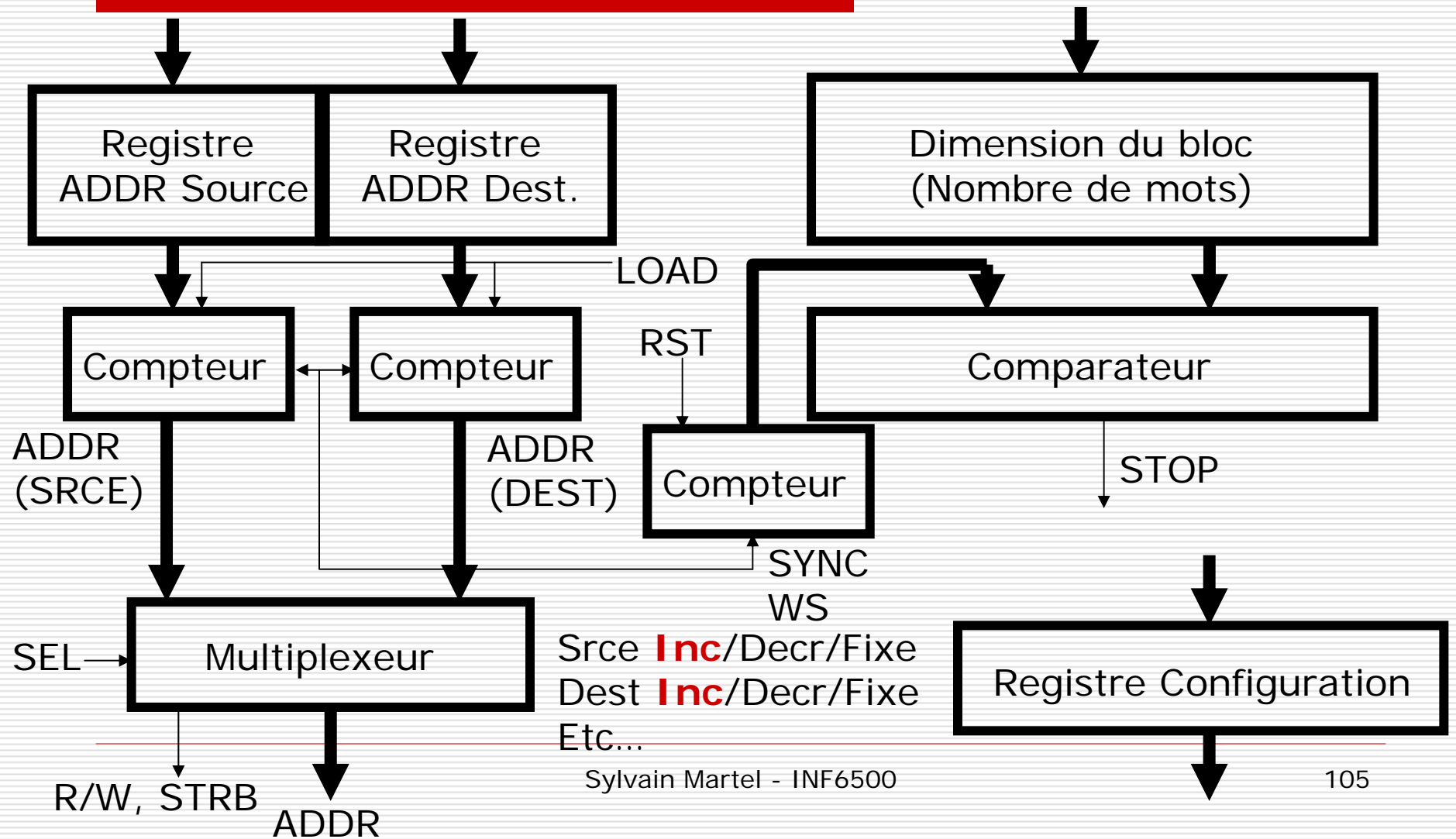
Avec contrôleur DMA – T2



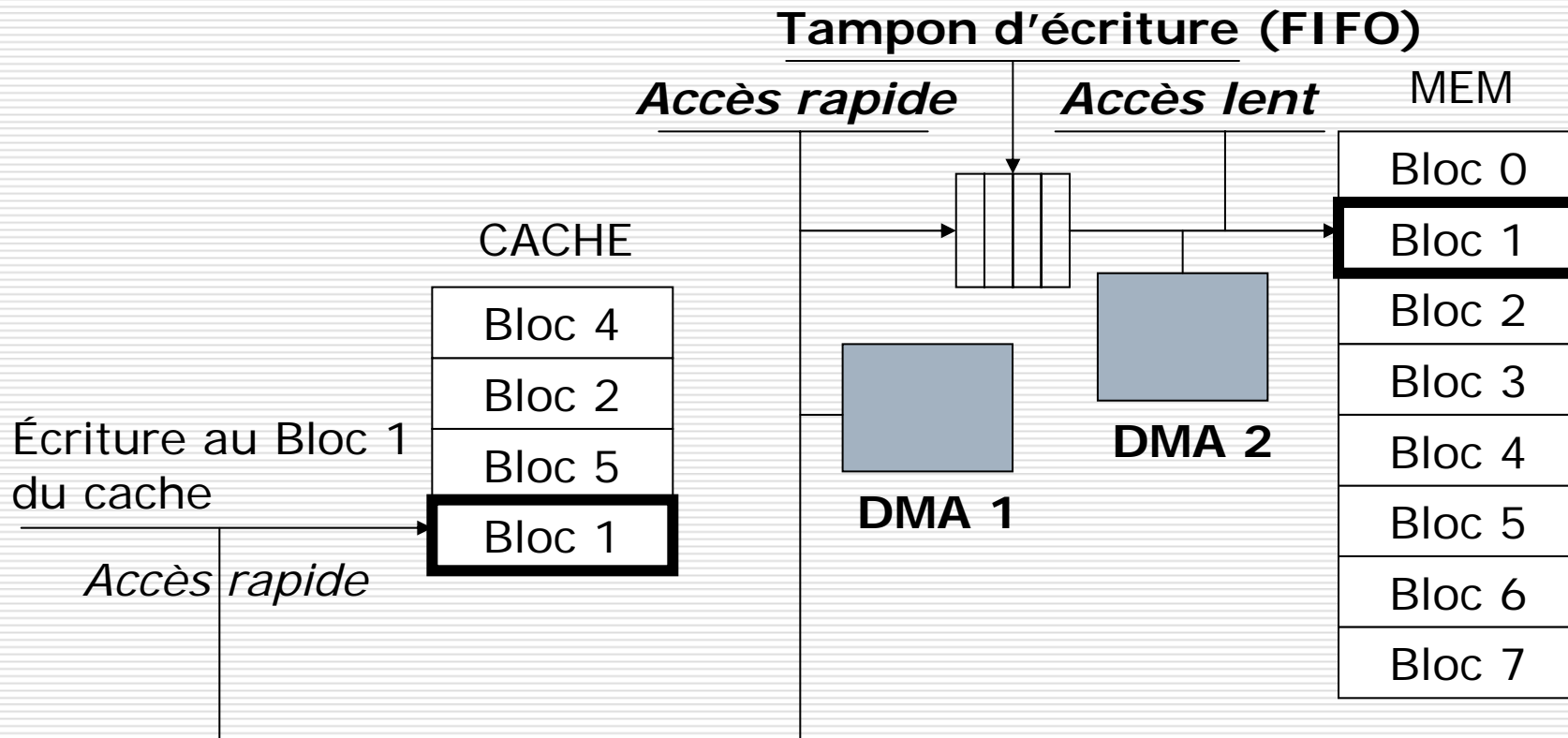
Contrôleur DMA – Fonctionnement interne

- L'UC initialise d'abord les registres du DMA, qui contiennent une adresse mémoire et le nombre d'octets à transférer. Une fois le transfert DMA terminé, le contrôleur interrompt l'UCT.
- Il peut y avoir plusieurs composants DMA dans un ordinateur
- Il est possible d'accroître la performance DMA et donc décharger d'avantage l'UCT: des composants appelés processeurs d'E/S (*I/O processor controller*) opèrent à partir de programmes fixes ou de programmes chargés par l'OS. Celui-ci établit généralement une file d'attente de blocs de contrôle E/S qui contiennent des informations telles que la localisation de données et la taille des données. Le processeur d'E/S prend alors les informations dans la file, fait tout ce qui est demandé et envoie une interruption seulement lorsque la tâche spécifiée dans les blocs de contrôle des E/S est terminée.

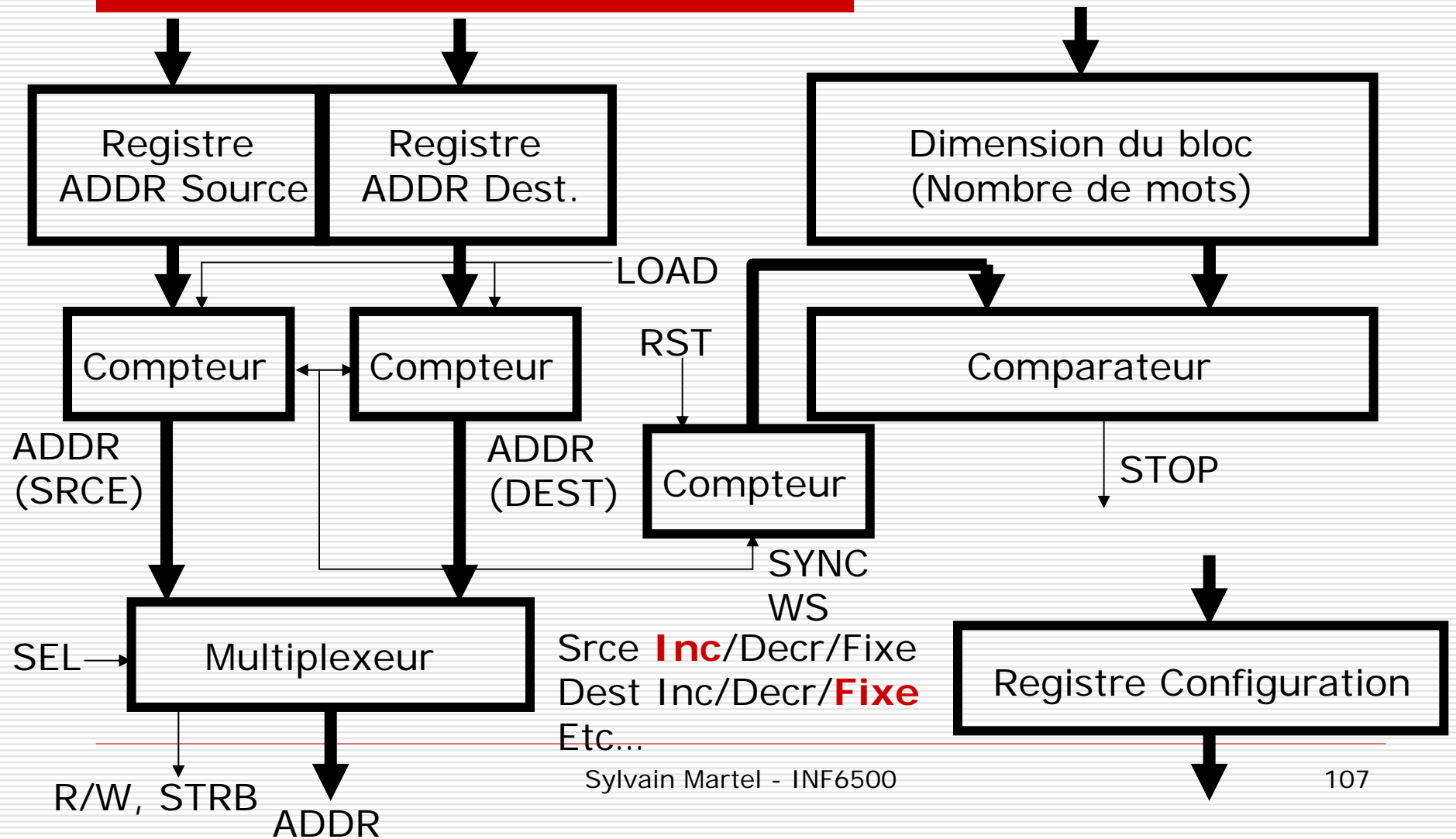
Contrôleur DMA – Fonctionnement interne (exemple)



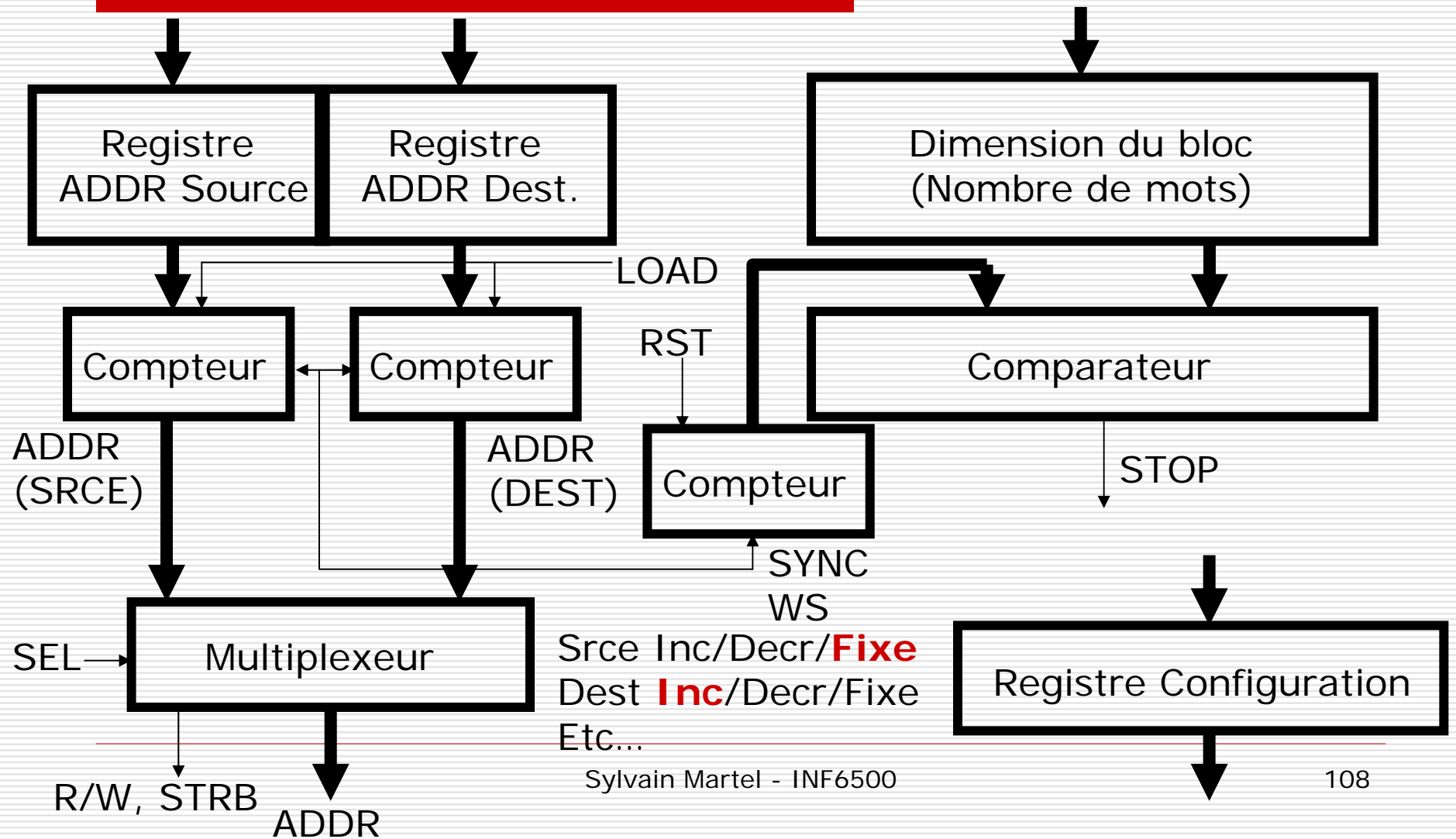
Transfert **avec** tampon d'écriture de type FIFO du bloc 1 de la mémoire cache à la mémoire principale



Configuration: Contrôleur DMA 1



Configuration: Contrôleur DMA 2



Autres types de contrôleurs

- Réseaux;
- Interruptions;
- Graphiques;
- Bus;
- DRAM (refresh, ...)
- Disk;
- Mémoire (cache);
- Etc.

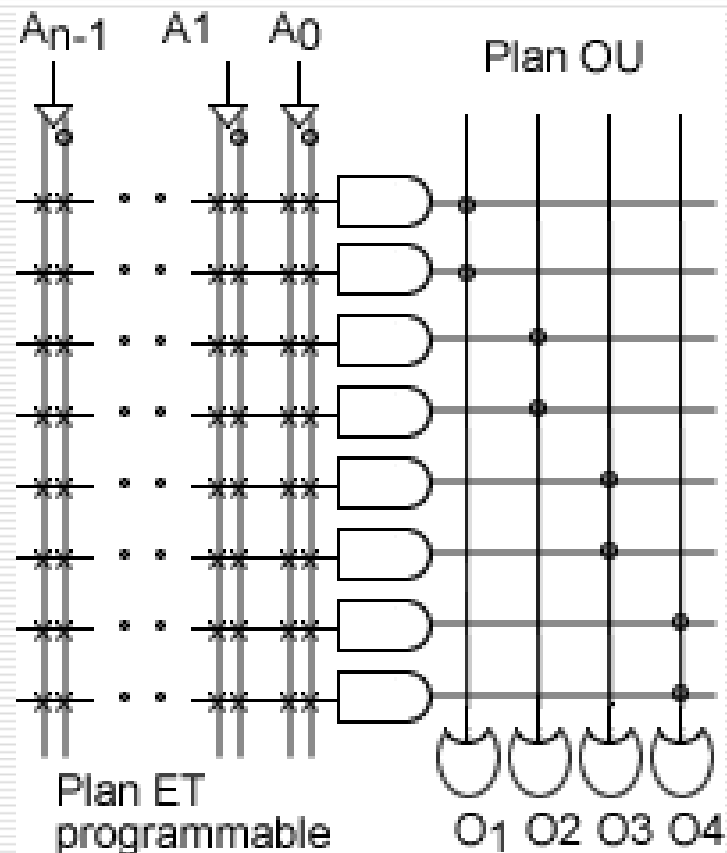
Implémentation : Electronique Programmable - Choix

Choix – Electronique programmable

- PAL
- GAL
- CPLD
- FPGA

Choix – Electronique programmable

- ❑ PAL (*Programmable Array Logic*)
 - Logique restreinte (par rapport aux CPLDs et FPGAs).
 - C'est le plus populaire des SPLDs.
 - Programmable une seule fois par survoltage.
 - Plan ET programmable.
 - Plan OU fixe.

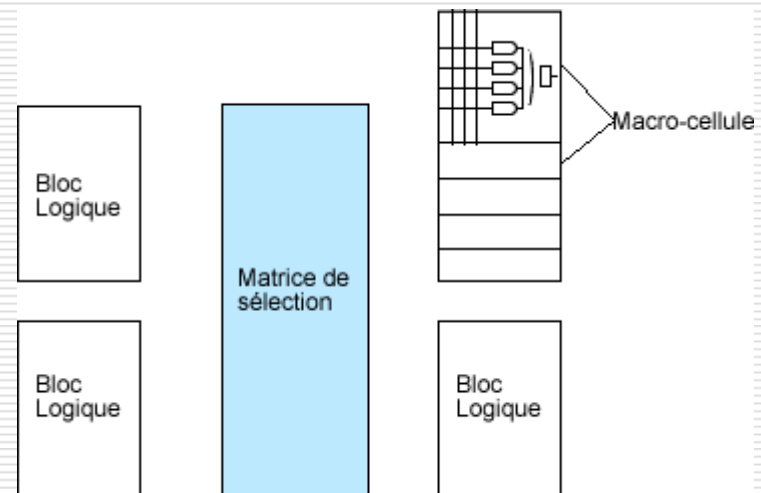


Choix – Electronique programmable

- GAL (*Generic Array Logic*)
 - C'est un PAL reprogrammable
 - Il est constitué de plusieurs macro-cellules que l'utilisateur peut configurer en un bloc combinatoire simple, complexe ou bien en sortie avec registres.

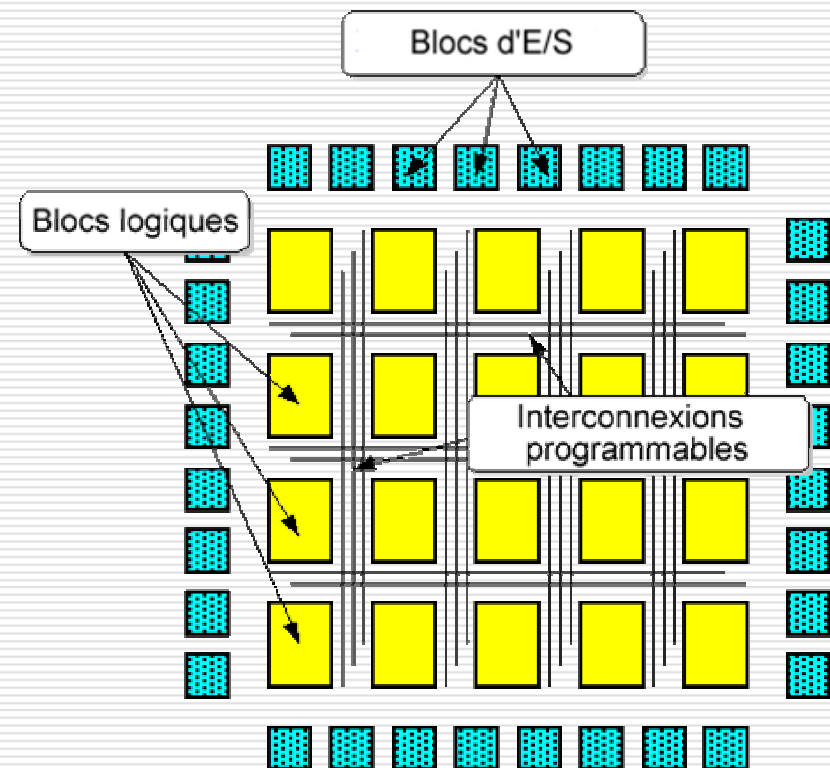
Choix – Electronique programmable

- CPLD (*Complex Programmable Logic Devices*)
 - D'un point de vue conceptuel, un CPLD consiste en plusieurs macro-cellules de PAL interconnectées par l'entremise d'une matrice de sélection.
 - Contient de quelques dizaines à quelques centaines de macro-cellules.
 - En général, les groupes de 8 à 16 macro-cellules se regroupés dans un bloc de fonction.
 - Selon le distributeur et la famille du CPLD, les blocs de fonction sont également interconnectés.

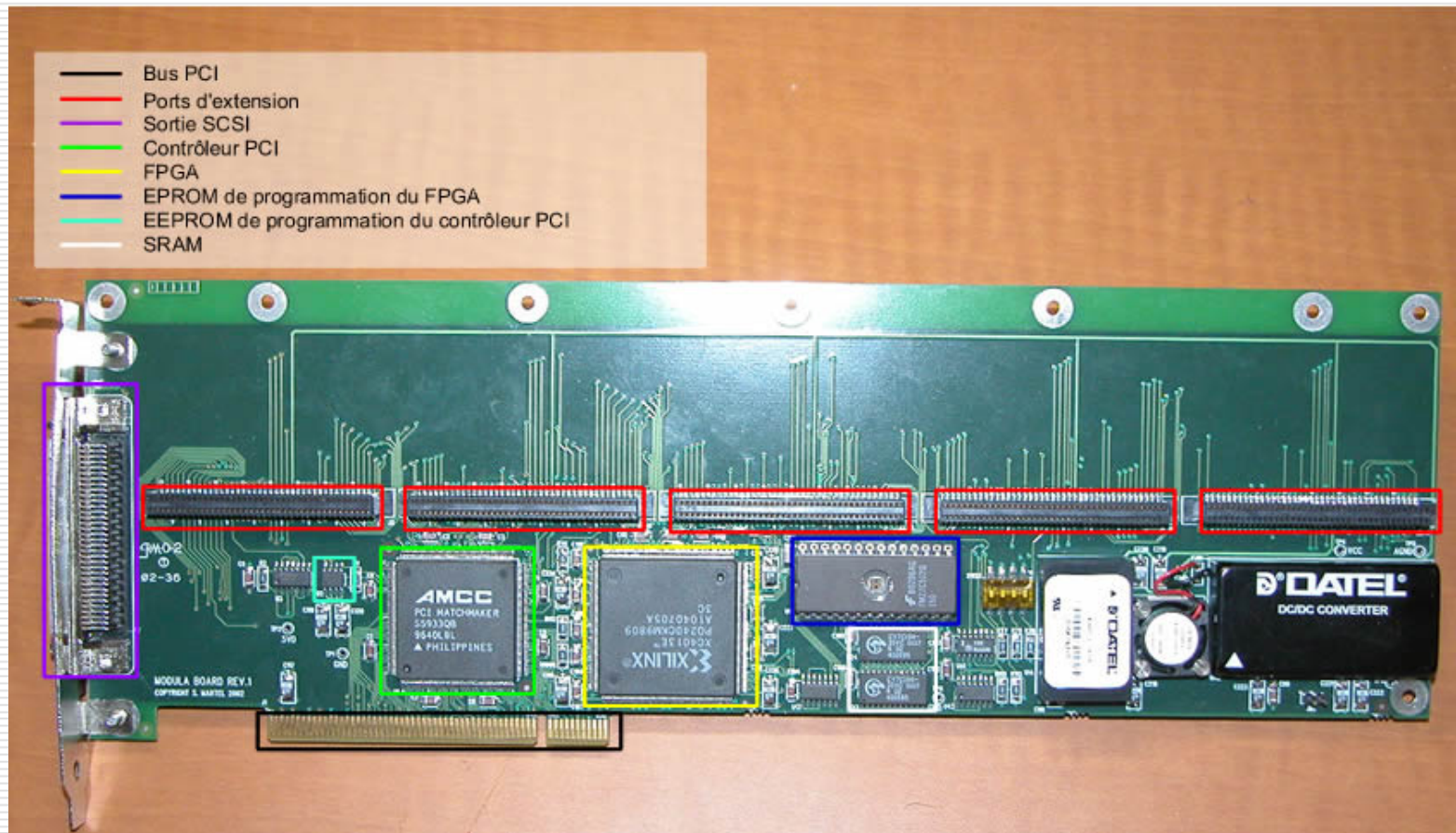


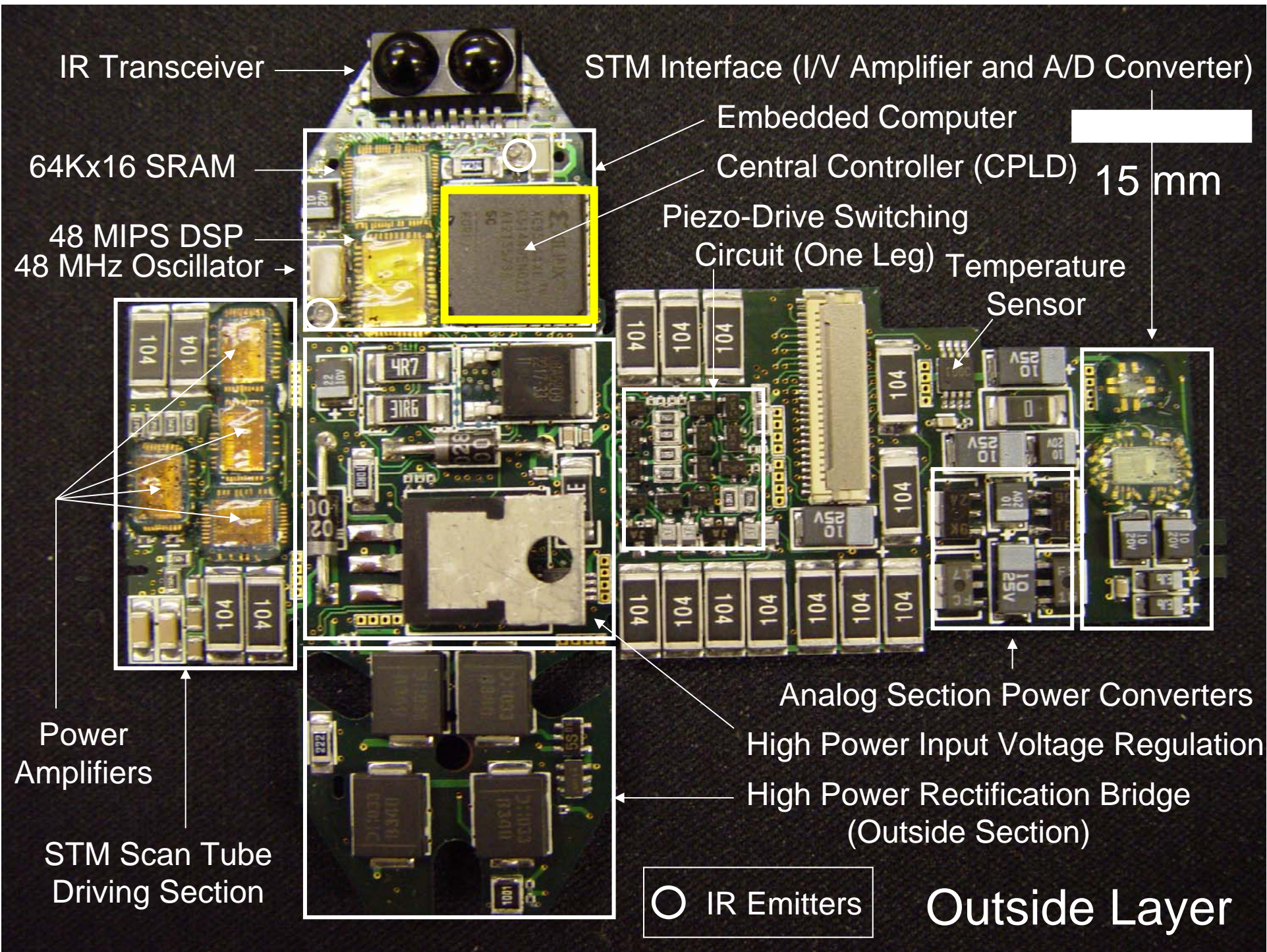
Choix – Electronique programmable

- FPGA (*Field Programmable Gate Array*)
 - Offre la plus grande capacité logique.
 - Un FPGA est une matrice de blocs logiques, entourés de blocs d'E/S, reliés par des connexions programmables.
 - Un FPGA peut contenir entre 64 et plusieurs dizaines de milliers de blocs logiques et un nombre de bascules encore plus grand.
 - Il existe deux types de FPGA:
 - À gros grains (*'coarse-grained'*): contient des blocs logiques relativement gros, contenant deux tables de conversion ou plus ainsi que deux bascules ou plus.
 - À grains fins (*'fine-grained'*): contient d'avantage de blocs logiques simples. Ces blocs contiennent soit une fonction d'entrée double ou un multiplexeuru 4-1 et des bascules.
 - Les FPGAs les plus denses sont construits à l'aide de la technologie des SRAMs
 - Le procédé d'anti-fusibles est également utilisé parfois.



Carte MODULA





IR Transceiver

STM Interface (I/V Amplifier and A/D Converter)

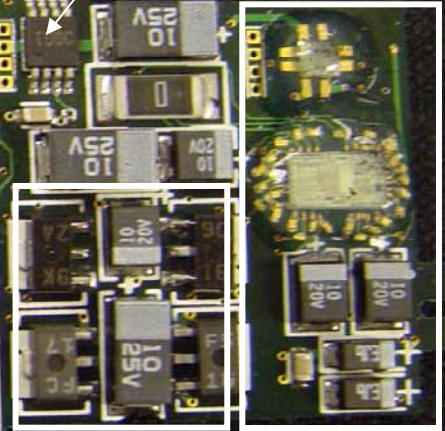
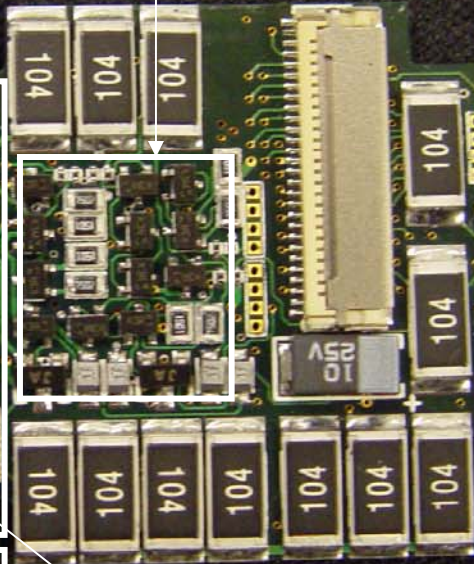
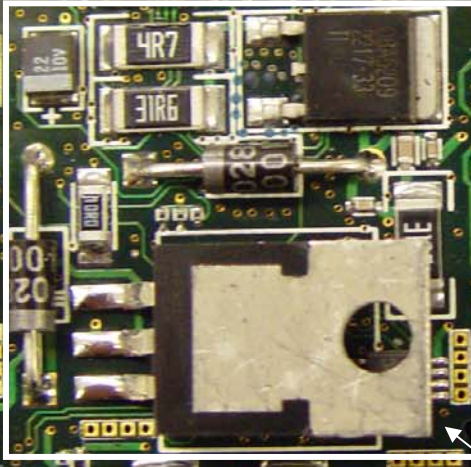
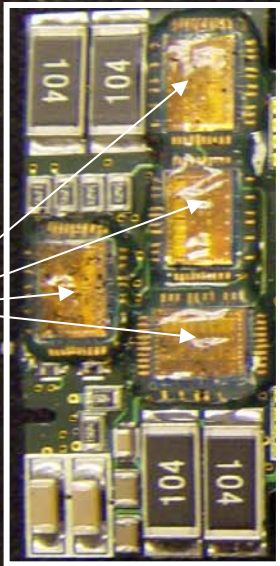
64Kx16 SRAM

Embedded Computer

Central Controller (CPLD) 15 mm

48 MIPS DSP
48 MHz Oscillator

Piezo-Drive Switching
Circuit (One Leg) Temperature
Sensor



Power
Amplifiers

STM Scan Tube
Driving Section

Analog Section Power Converters
High Power Input Voltage Regulation
High Power Rectification Bridge
(Outside Section)

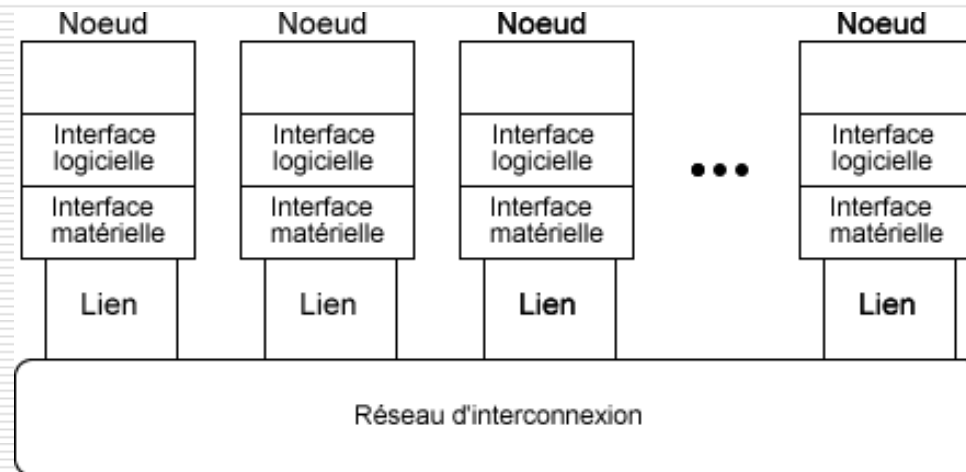
○ IR Emitters

Outside Layer

Réseaux d'interconnexion

Introduction

- ❑ Utilité : connecter un ensemble d'ordinateurs pour former une communauté.
- ❑ Une communauté est composée de:
 - Nœuds (*node*)
 - Interface logicielle (*software interface*)
 - Interface matérielle (*hardware interface*)
 - Lien (*link*)
 - Réseau d'interconnexion (*interconnection network*)

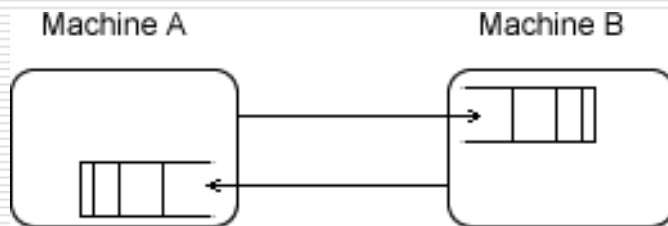


Introduction

- En fonction de nombre de nœuds et de leur proximité, les réseaux ont plusieurs noms:
 - Réseau de machine massivement parallèle (*massively parallel network processor*)
 - Peut connecter des milliers de nœuds
 - Distance maximale généralement de moins de 25 mètres
 - Réseau local (*local area network LAN*)
 - Peut connecter des centaines à des milliers d'ordinateurs
 - La distance peut atteindre quelques kilomètres
 - Réseau longue distance (*wide area network*)
 - Connexion de milliers d'ordinateurs
 - Connexion à travers le monde

Un réseau simple

- ❑ Réseau de 2 ordinateurs:



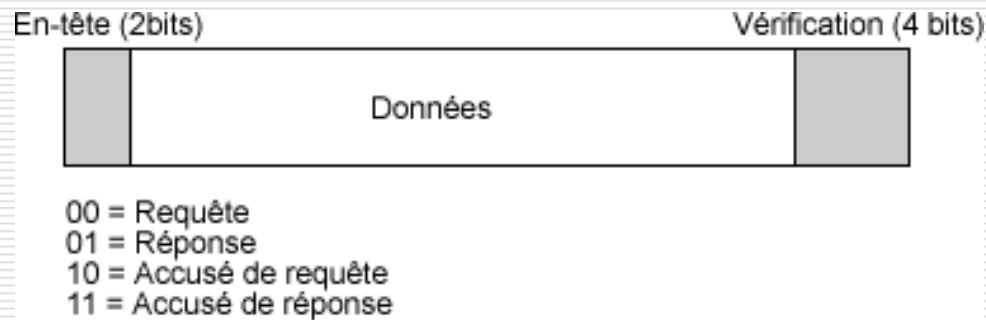
- ❑ Les machines communiquent par messages
- ❑ C'est une partie logicielle du système d'exploitation qui permet de traduire les requêtes (messages)

Envoyer et recevoir un message

- L'envoi du message se déroule comme ci-dessous:
 - L'application copie la donnée à envoyer dans un tampon du système d'exploitation
 - Le système d'exploitation calcule le code de redondance, l'incluse dans l'en-tête ou la queue du message, et démarre alors le chronomètre
 - Le système d'exploitation envoie la donnée dans le matériel d'interface réseau et indique au matériel d'envoyer le message.
- La réception du message suit alors:
 - Le système copie la donnée depuis le matériel de l'interface réseau dans le tampon du système d'exploitation.
 - Le système calcule le code de redondance sur les données. Si ce code correspond au code de redondance de l'émetteur, le récepteur renvoie un accusé de réception à l'émetteur. Sinon, il supprime le message, en supposant que l'émetteur l'expédiera à nouveau quand le temps associé à l'envoi sera écoulé.
 - Si la donnée passe le test, le système copie la donnée dans l'espace d'adressage de l'utilisateur et indique à l'application de continuer.

Format de message

- Format de message



- Cette suite de règles que suit le logiciel se nomme un protocole
 - Ex: UDP/IP, TCP/IP

Mesures de la performance du réseau

- Pour quantifier les performances des réseaux, on utilise plusieurs mesures:
 - La bande passante (*bandwidth*): Délai maximal avec lequel le réseau d'interconnexion peut propager l'information une fois que le message est entré dans le réseau.
 - Le temps de transmission (*transmission time*): Temps pour que le message qui passe à travers le réseau soit égal à la taille du message divisé par la bande passante.
 - Le temps de transit (*time of flight*): temps pour que le premier bit du message arrive au récepteur. Cela comprend les délais dûs aux répéteurs ou autres éléments matériels dans le réseau.
 - La latence de transport (*transport latency*): La somme du temps de transit et du temps de transmission.
 - Surcoût émetteur (*sender overhead*): temps pour que le processeur injecte le message dans le réseau d'interconnexion.

Mesures de la performance du réseau (suite)

- Surcoût récepteur (*receiver overhead*): Temps pour que le processeur retire le message du réseau d'interconnexion.

$$Latence\ totale = Surcoût\ émetteur + Temps\ transit + \frac{Taille\ message}{Débit} + Surcoût\ récepteur$$

Connecter le réseau d'interconnexion à l'ordinateur

- ❑ Les machines massivement parallèle utilisent typiquement le bus mémoire alors que les réseaux locaux et de longue distance se connectent sur le bus d'E/S
- ❑ En général, les gros messages sont envoyés plus efficacement par un DMA.

Les supports des réseaux d'interconnexion

- Il existe plusieurs supports pour permettre aux machines de communiquer entre elles:
 - Paire torsadée (*twisted pair*): peut transférer quelques megabits/s sur plusieurs kilomètres sans amplification.
 - Câble coaxial : un câble de 50 ohms peut délivrer 10 mégabits/s sur 1 kilomètre.
 - Fibre optique (*fiber optic*): Transmettent l'information sous forme d'impulsion de lumière. Elle sont unidirectionnelles.
 - Fibre multimode : permet à la lumière de se disperser et utilise comme source de lumière des LED bon marché. Permet de transporter l'information à 1000M bits/s sur quelques centaines de mètres ou à 100 M bits/s sur quelques kilomètres.
 - La fibre monomode: possède une seule longueur d'onde. Transporte des gigabits d'information par seconde sur des centaines de kilomètres.
 - NOTE: Connecter des fibres monomodes est beaucoup plus difficile que de connecter des fibres multimodes.

Connecter plus de deux ordinateurs

- Support partagé ou commuté
 - La manière la plus simple de connecter plusieurs ordinateurs est de les faire partager un seul support d'interconnexion. Le réseau local le plus populaire est ethernet.
 - Si le support est partagé, il doit y avoir un mécanisme pour coordonner l'utilisation du support partagé, pour qu'un seul message soit envoyé à la fois.
 - Si le réseau contient un grand nombre de nœuds sur un plusieurs kilomètres, il faut alors un arbitrage distribué.
 - Un nœud écoute d'abord pour être sûr qu'il n'envoie pas un message pendant qu'un autre message est sur le réseau.
 - Si par hasard, deux messages se trouvent sur le réseau en même temps, il y a collision (deux nœuds envoient un message en même temps)

Connecter plus de deux ordinateurs

- L'interface réseau peut essayer de détecter ces collisions en regardant si plusieurs messages ont été mélangés. C'est la détection de collision.
- Les supports partagés sont bon marché mais ils ont une bande passante limitée
- Une alternative au partage du support est d'avoir une ligne dédiée vers un commutateur (*switch*) qui fournit à son tour une ligne dédiée vers d'autres destinations. La bande passante est grandement améliorée.
- Les commutateurs permettent une communication directe de la source vers la destination, sans que des nœuds intermédiaires interfèrent avec cette communication (*point to point communication*)

Connecter plus de deux ordinateurs

- Topologie de commutateur
 - Plusieurs topologies théoriques, mais très peu sont implantées dans l'industrie. On en nomme deux:
 - Topologie des machines massivement parallèles (MMP)
 - Topologie des réseaux locaux et longue distance
 - Un réseau crossbar permet à tout nœud de communiquer avec tout nœud avec une seule traversée du réseau.
 - Un réseau Oméga utilise moins de matériel que le réseau crossbar ($n \log_2 n$ au lieu de n^2). On appelle contention les différents blocages possibles entre les nœuds.
 - Une autre topologie consiste en un arbre avec une augmentation de la bande passante dans l'arbre pour répondre aux besoins des configurations courantes de communication.

Connecter plus de deux ordinateurs

- Une autre solution pour un commutateur distribué est un réseau qui connecte une suite de noeuds. C'est la topologie de l'anneau. Un anneau est capable de plusieurs transferts simultanés, contrairement aux lignes partagées.
- Une variante des anneaux qui est utilisée dans les réseaux locaux est appelée anneau à jetons (*token ring*). Un seul emplacement (le jeton) est passé le long de l'anneau pour déterminer que noeud peut envoyer un message. Un noeud ne peut envoyer un message que s'il obtient le jeton.
- Routage: délivrer les messages
 - Le support commuté utilise trois solutions de routage
 - Le routage à partir de la source: le message spécifie le chemin vers la destination.
 - Le circuit virtuel: Un circuit est établi entre la source et la destination. Le message ensuite le chemin qu'il suit.

Connecter plus de deux ordinateurs

- Le routage fondé sur la destination: un message contient une adresse destination et le commutateur doit choisir un chemin pour délivrer le message.
- Dans les réseaux longue distance, les commutateurs routent les message avec la politique ranger et propager (*store and forward*). Chaque commutateur attend que le message complet arrive au commutateur avant de l'envoyer au commutateur suivant.
- Le contrôle de congestion
 - Lorsque la bande passante diminue, l'interconnexion peut devenir engorgée avec trop de paquets.
 - La solution de base: éviter de faire entrer de nouveaux paquets avant que le trafic soit réduit. Pour ce faire, trois méthodes existent:
 - Suppression de paquets : si un paquet arrive à un commutateur et qu'il n'y a pas de place dans le tampon, le paquet est supprimé.

Connecter plus de deux ordinateurs

- ❑ Contrôle de flux: Utiliser une contre réaction pour indiquer à l'émetteur quand il est autorisé à envoyer le paquet suivant.
- ❑ Paquets de notification de congestion: c'est à chaque commutateur de voir son niveau d'occupation et d'entrer dans un état d'avertissement quand il dépasse un certain seuil. Chaque paquet reçu dans un état d'avertissement sera renvoyé à la source via un paquet de notification de congestion qui comprend la destination voulue.

Les questions pratiques pour les réseaux d'interconnexion commerciaux

□ La normalisation

- les standards populaires sont peu chers et très stables.
- Une norme permet à beaucoup de sociétés de fabriquer de produits avec des interfaces pour la norme, et le client n'a pas à attendre d'une seule compagnie qu'elle développe les interfaces pour tous les produits qui peuvent l'intéresser.
- Il faut cependant un temps important avant qu'une norme voit le jour en raison de la latence des comités à se mettre d'accord.
- Les réseaux d'interconnexion des MMP sont propriétaires alors que les réseaux locaux et longue distance sont normalisés.

□ La tolérance aux pannes

- Savoir si tous les nœuds de l'interconnexion sont opérationnels pour fonctionner correctement.
- Cependant, aucun réseau comportant un grand nombre de nœuds peut espérer aucune défaillance

Exemple de réseaux d'interconnexion

- Ethernet
 - Standard 10 Mb/s proposé en 1978
 - Utilisé pratiquement partout
 - Utilise la détection de paquets avec une loi exponentielle pour arbitrer l'accès au réseau
 - Codifié sous la norme IEEE 802.3
- Selon la manière dont les composants passent le trafic et les types de réseaux d'interconnexion qu'ils peuvent relier, ils ont des noms différents:
 - Ponts: connectent des réseaux locaux entre eux, passant le trafic d'un côté à l'autre selon l'adresse du paquet.
 - Routeurs: Connectent des réseaux locaux à des réseaux longue distance ou des réseaux longue distance entre eux et résolvent les problèmes d'adresses incompatibles.

Exemple de réseaux d'interconnexion

- Un successeur potentiel d'Ethernet est FDDI (Interface Données Distribuées par Fibre) (*Fiber Distributed Data Interface*). Ce réseau est fondé sur l'optique à 100 Mb/s et pourrait utiliser des câbles beaucoup plus longs.

Questions transversales pour les réseaux d'interconnexion

- ❑ Une interface efficace à la hiérarchie mémoire ou au réseau d'interconnexion
- ❑ Processeurs optimisés pour le traitement et surcoût récepteur
- ❑ Où tracer la frontière matériel/logiciel pour les fonctions d'interconnexion
- ❑ Protection est accès utilisateur au réseau

L'interconnexion de réseaux

- ❑ Une des inventions les plus importantes dans le domaine de communications a été la connexion des réseaux.
- ❑ Les technologies nécessaires pour la connexion de réseaux sont des normes logicielles qui permettent une communication fiable sans nécessiter des réseaux fiables.
- ❑ Chaque ordinateur, réseau et commutateur implémente sa couche de normes, s'appuyant sur les autres composants pour remplir fidèlement ses obligations (OSI).
- ❑ La norme la plus populaire est TCP/IP .
- ❑ Une famille de protocoles permet de simplifier la norme en divisant de manière hiérarchique la responsabilité entre les couches. Chaque niveau fournit les services nécessaires à la couche supérieure.

L'interconnexion de réseaux

□ Les différentes couches OSI

Numéro de couche	Nom de la couche
7	Application
6	Présentation
5	Session
4	Transport
3	Network
2	Data Link
1	Physical

