

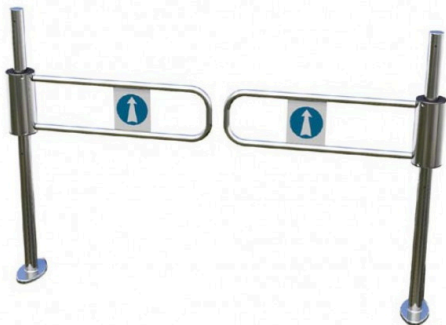
# Architecture des ordinateurs VII

Frédéric Boulanger

CentraleSupélec



# Entrées-sorties



# Architecture de von Neumann

- ▶ unité arithmétique et logique (UAL) OK



# Architecture de von Neumann

- ▶ unité arithmétique et logique (UAL) OK
- ▶ mémoire (données et programme) OK



# Architecture de von Neumann

- ▶ unité arithmétique et logique (UAL) OK
- ▶ mémoire (données et programme) OK
- ▶ unité de contrôle (séquençage des opérations) OK



# Architecture de von Neumann

- ▶ unité arithmétique et logique (UAL) OK
- ▶ mémoire (données et programme) OK
- ▶ unité de contrôle (séquençage des opérations) OK
- ▶ dispositifs d'entrée-sortie ?



# Entrées-sorties

- ▶ faire entrer des informations extérieures dans le processeur

# Entrées-sorties

- ▶ faire entrer des informations extérieures dans le processeur
- ▶ communiquer des informations du processeur vers l'extérieur





# Entrées-sorties

- ▶ faire entrer des informations extérieures dans le processeur
- ▶ communiquer des informations du processeur vers l'extérieur
- ▶ comment s'adresser à l'*extérieur* ?



# Entrées-sorties

- ▶ faire entrer des informations extérieures dans le processeur
- ▶ communiquer des informations du processeur vers l'extérieur
- ▶ comment s'adresser à l'*extérieur* ?
  - ▶ de la même façon qu'à la mémoire



# Entrées-sorties

- ▶ faire entrer des informations extérieures dans le processeur
- ▶ communiquer des informations du processeur vers l'extérieur
- ▶ comment s'adresser à l'*extérieur* ?
  - ▶ de la même façon qu'à la mémoire
  - ▶ certaines adresses correspondent à l'environnement :



# Entrées-sorties

- ▶ faire entrer des informations extérieures dans le processeur
- ▶ communiquer des informations du processeur vers l'extérieur
- ▶ comment s'adresser à l'*extérieur* ?
  - ▶ de la même façon qu'à la mémoire
  - ▶ certaines adresses correspondent à l'environnement :
    - ▶ lecture = acquisition, entrée



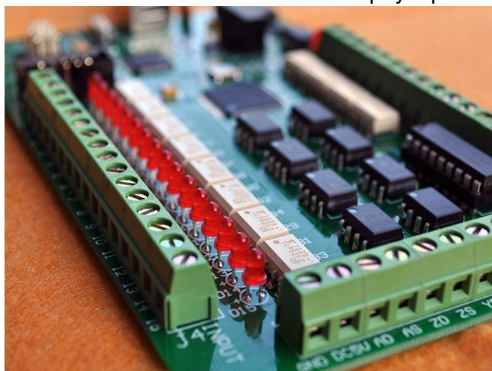
# Entrées-sorties

- ▶ faire entrer des informations extérieures dans le processeur
- ▶ communiquer des informations du processeur vers l'extérieur
- ▶ comment s'adresser à l'*extérieur* ?
  - ▶ de la même façon qu'à la mémoire
  - ▶ certaines adresses correspondent à l'environnement :
    - ▶ lecture = acquisition, entrée
    - ▶ écriture = action, sortie



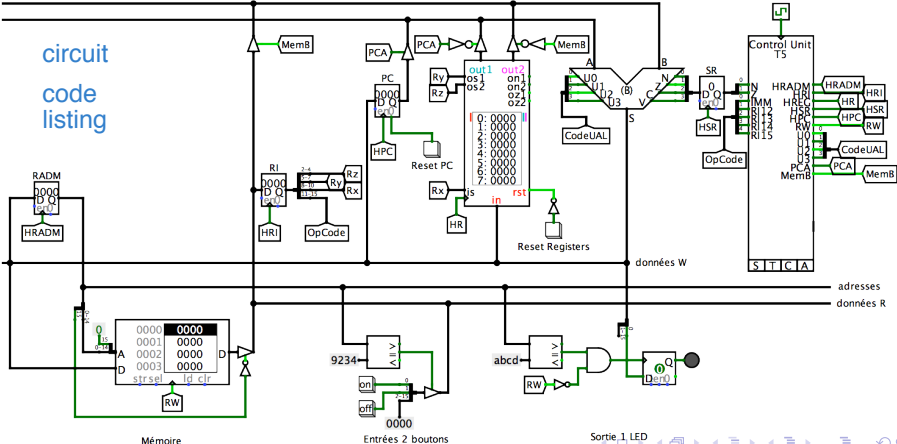
# Entrées-sorties

- ▶ faire entrer des informations extérieures dans le processeur
- ▶ communiquer des informations du processeur vers l'extérieur
- ▶ comment s'adresser à l'*extérieur* ?
  - ▶ de la même façon qu'à la mémoire
  - ▶ certaines adresses correspondent à l'environnement :
    - ▶ lecture = acquisition, entrée
    - ▶ écriture = action, sortie
  - ▶ interface entre les données et le monde physique : coupleur



# Entrées-sorties pour notre processeur

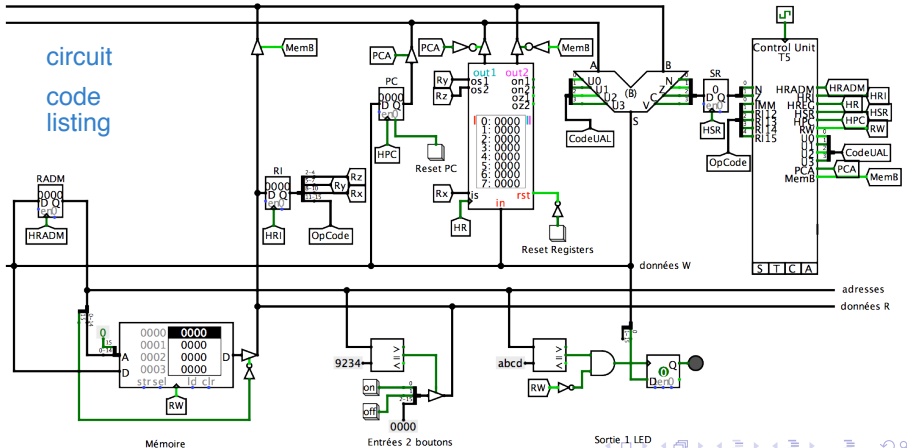
- ▶ la sortie du RADM devient le bus d'adresse



circuit  
code  
listing

# Entrées-sorties pour notre processeur

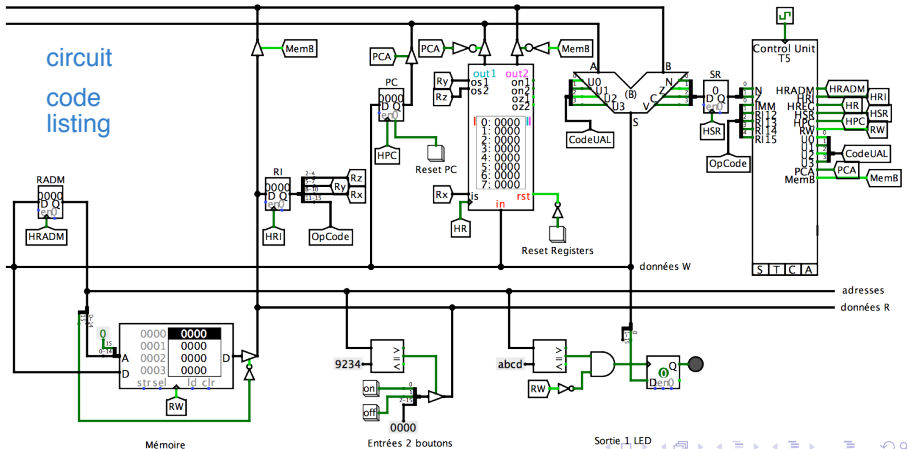
- ▶ la sortie du RADM devient le bus d'adresse
- ▶ on réserve la moitié des adresses pour les E/S





# Entrées-sorties pour notre processeur

- ▶ la sortie du RADM devient le bus d'adresse
- ▶ on réserve la moitié des adresses pour les E/S
- ▶ chaque périphérique compare l'adresse avec la sienne



# Entrées-sorties

## Périphériques plus complexes

- ▶ le contrôleur possède plusieurs registres

# Entrées-sorties

## Périphériques plus complexes

- ▶ le contrôleur possède plusieurs registres
- ▶ ces registres pilotent des signaux plus complexes



# Entrées-sorties

## Périphériques plus complexes

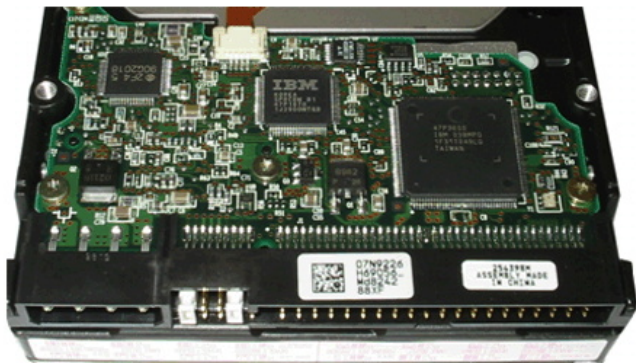
- ▶ le contrôleur possède plusieurs registres
- ▶ ces registres pilotent des signaux plus complexes
- ▶ il peut même y avoir un processeur dans le périphérique !



# Entrées-sorties

## Périphériques plus complexes

- ▶ le contrôleur possède plusieurs registres
- ▶ ces registres pilotent des signaux plus complexes
- ▶ il peut même y avoir un processeur dans le périphérique !
- ▶ Exemple : contrôleur de disque dur :



# Bus



# Bus

## Rôle

- ▶ transporter les données

# Bus

## Rôle

- ▶ transporter les données
- ▶ éviter les liaisons point à point



# Bus

## Rôle

- ▶ transporter les données
- ▶ éviter les liaisons point à point
- ▶ connecter différents types de périphériques



# Bus

## Rôle

- ▶ transporter les données
- ▶ éviter les liaisons point à point
- ▶ connecter différents types de périphériques

## Nature

# Bus

## Rôle

- ▶ transporter les données
- ▶ éviter les liaisons point à point
- ▶ connecter différents types de périphériques

## Nature

- ▶ série : transmission bit par bit

## Rôle

- ▶ transporter les données
- ▶ éviter les liaisons point à point
- ▶ connecter différents types de périphériques

## Nature

- ▶ série : transmission bit par bit
- ▶ parallèle : transmission mot par mot

## Rôle

- ▶ transporter les données
- ▶ éviter les liaisons point à point
- ▶ connecter différents types de périphériques

## Nature

- ▶ série : transmission bit par bit
- ▶ parallèle : transmission mot par mot
- ▶ synchrone : échanges régis par une horloge

## Rôle

- ▶ transporter les données
- ▶ éviter les liaisons point à point
- ▶ connecter différents types de périphériques

## Nature

- ▶ série : transmission bit par bit
- ▶ parallèle : transmission mot par mot
- ▶ synchrone : échanges régis par une horloge
- ▶ asynchrone : échanges régis par un protocole

# Bus synchrone

## Principe

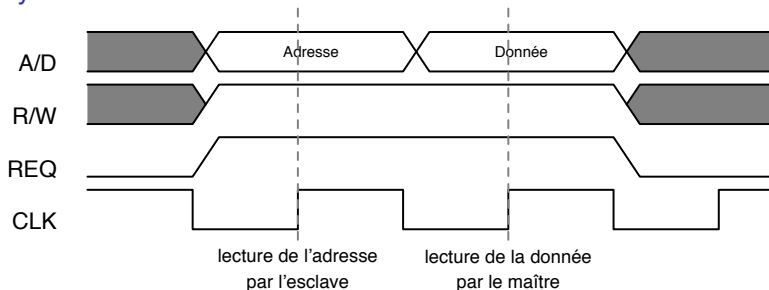
- ▶ une horloge indique quand échanger les données

# Bus synchrone

## Principe

- ▶ une horloge indique quand échanger les données

## Cycle de lecture



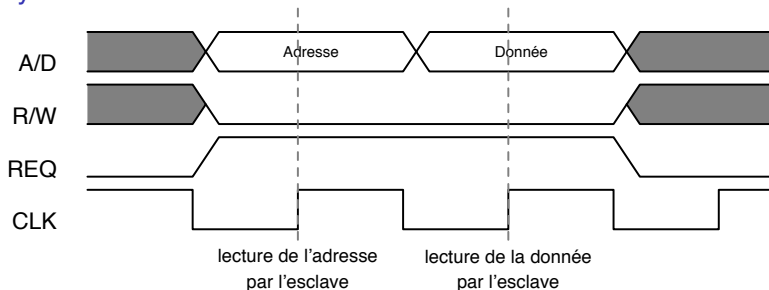


# Bus synchrone

## Principe

- ▶ une horloge indique quand échanger les données

## Cycle d'écriture



# Bus asynchrone

## Principe

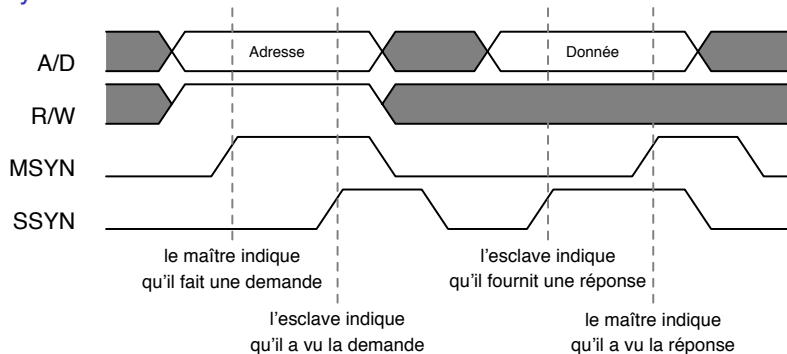
- ▶ un protocole permet de savoir quand échanger les données (handshake)

# Bus asynchrone

## Principe

- ▶ un protocole permet de savoir quand échanger les données (handshake)

## Cycle de lecture

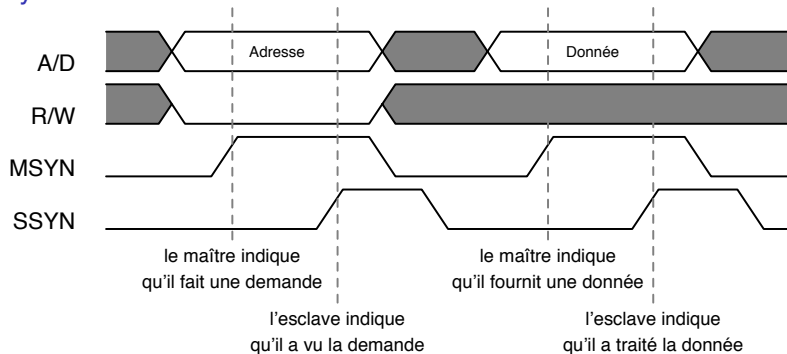


# Bus asynchrone

## Principe

- ▶ un protocole permet de savoir quand échanger les données (handshake)

## Cycle d'écriture



# Arbitrage

Et si plusieurs maîtres veulent utiliser le bus en même temps ?

- ▶ un maître ne peut demander le bus que s'il est libre

# Arbitrage

Et si plusieurs maîtres veulent utiliser le bus en même temps ?

- ▶ un maître ne peut demander le bus que s'il est libre
- ▶ si plusieurs maîtres le demandent en même temps il faut en choisir un : **arbitrage**



Et si plusieurs maîtres veulent utiliser le bus en même temps ?

- ▶ un maître ne peut demander le bus que s'il est libre
- ▶ si plusieurs maîtres le demandent en même temps il faut en choisir un : **arbitrage**
- ▶ méthodes d'arbitrage :

Et si plusieurs maîtres veulent utiliser le bus en même temps ?

- ▶ un maître ne peut demander le bus que s'il est libre
- ▶ si plusieurs maîtres le demandent en même temps il faut en choisir un : **arbitrage**
- ▶ méthodes d'arbitrage :
  - ▶ choisir le maître qui a l'adresse la plus grande



Et si plusieurs maîtres veulent utiliser le bus en même temps ?

- ▶ un maître ne peut demander le bus que s'il est libre
- ▶ si plusieurs maîtres le demandent en même temps il faut en choisir un : **arbitrage**
- ▶ méthodes d'arbitrage :
  - ▶ choisir le maître qui a l'adresse la plus grande
  - ▶ donner tour à tour la priorité à chacun des maîtres

Et si plusieurs maîtres veulent utiliser le bus en même temps ?

- ▶ un maître ne peut demander le bus que s'il est libre
- ▶ si plusieurs maîtres le demandent en même temps il faut en choisir un : **arbitrage**
- ▶ méthodes d'arbitrage :
  - ▶ choisir le maître qui a l'adresse la plus grande
  - ▶ donner tour à tour la priorité à chacun des maîtres
  - ▶ n'en choisir aucun et les laisser réessayer

# Critères de choix

## Bus parallèle

- ▶ liaisons courtes (car beaucoup de fils)

# Critères de choix

## Bus parallèle

- ▶ liaisons courtes (car beaucoup de fils)
- ▶ débit élevé (1 à 2 Go/s pour le PCI-X)

# Critères de choix

## Bus parallèle

- ▶ liaisons courtes (car beaucoup de fils)
- ▶ débit élevé (1 à 2 Go/s pour le PCI-X)
- ▶ bus CPU-mémoire (synchrone)

# Critères de choix

## Bus parallèle

- ▶ liaisons courtes (car beaucoup de fils)
- ▶ débit élevé (1 à 2 Go/s pour le PCI-X)
- ▶ bus CPU-mémoire (synchrone)
- ▶ bus CPU-carte graphique, CPU-disques etc.

# Critères de choix

## Bus parallèle

- ▶ liaisons courtes (car beaucoup de fils)
- ▶ débit élevé (1 à 2 Go/s pour le PCI-X)
- ▶ bus CPU-mémoire (synchrone)
- ▶ bus CPU-carte graphique, CPU-disques etc.

## Bus série

# Critères de choix

## Bus parallèle

- ▶ liaisons courtes (car beaucoup de fils)
- ▶ débit élevé (1 à 2 Go/s pour le PCI-X)
- ▶ bus CPU-mémoire (synchrone)
- ▶ bus CPU-carte graphique, CPU-disques etc.

## Bus série

- ▶ liaisons longues (car peu de fils)



# Critères de choix

## Bus parallèle

- ▶ liaisons courtes (car beaucoup de fils)
- ▶ débit élevé (1 à 2 Go/s pour le PCI-X)
- ▶ bus CPU-mémoire (synchrone)
- ▶ bus CPU-carte graphique, CPU-disques etc.

## Bus série

- ▶ liaisons longues (car peu de fils)
- ▶ débits de plus en plus élevés (8 Gbit/s pour PCI Express)

# Critères de choix

## Bus parallèle

- ▶ liaisons courtes (car beaucoup de fils)
- ▶ débit élevé (1 à 2 Go/s pour le PCI-X)
- ▶ bus CPU-mémoire (synchrone)
- ▶ bus CPU-carte graphique, CPU-disques etc.

## Bus série

- ▶ liaisons longues (car peu de fils)
- ▶ débits de plus en plus élevés (8 Gbit/s pour PCI Express)
- ▶ possibilité de chaînage de périphériques

# Critères de choix

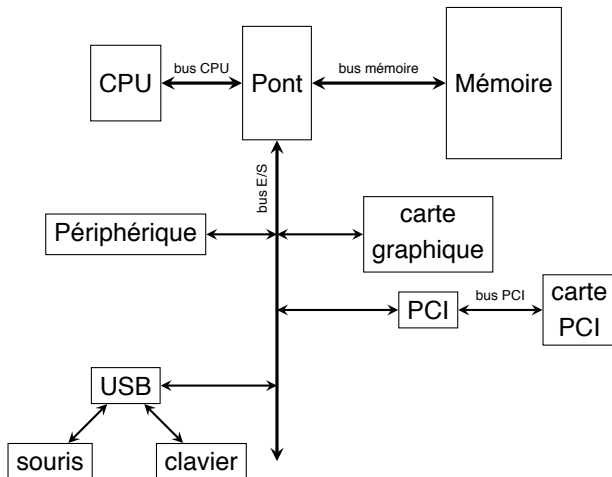
## Bus parallèle

- ▶ liaisons courtes (car beaucoup de fils)
- ▶ débit élevé (1 à 2 Go/s pour le PCI-X)
- ▶ bus CPU-mémoire (synchrone)
- ▶ bus CPU-carte graphique, CPU-disques etc.

## Bus série

- ▶ liaisons longues (car peu de fils)
- ▶ débits de plus en plus élevés (8 Gbit/s pour PCI Express)
- ▶ possibilité de chaînage de périphériques
- ▶ souris, clavier, imprimante (USB)

# Bus d'un ordinateur



Suite...

## Interruptions

