

---

# La technique du pipeline



Pierre Langlois

<http://creativecommons.org/licenses/by-nc-sa/2.5/ca/>

# La technique du pipeline

## Sujets de ce thème

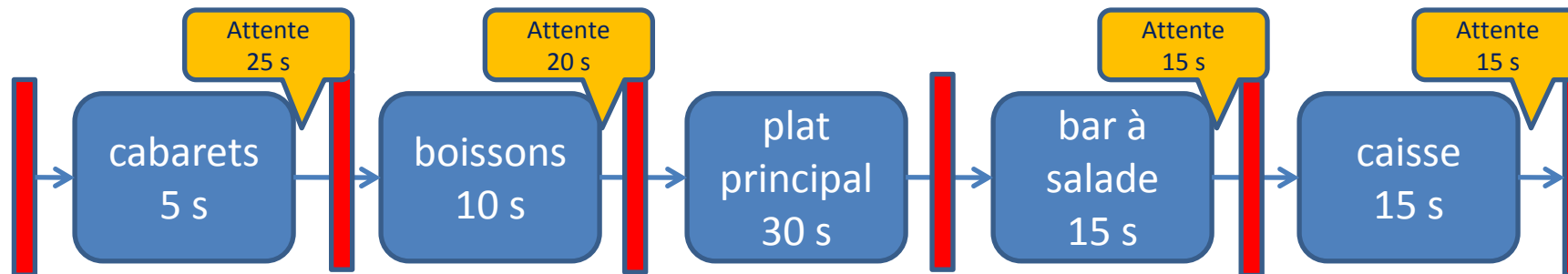
---

- Rappel
- Architecture à pipeline pour un circuit numérique
- Considérations pratiques
- Stratégie pour pipeliner un circuit
- Exemples

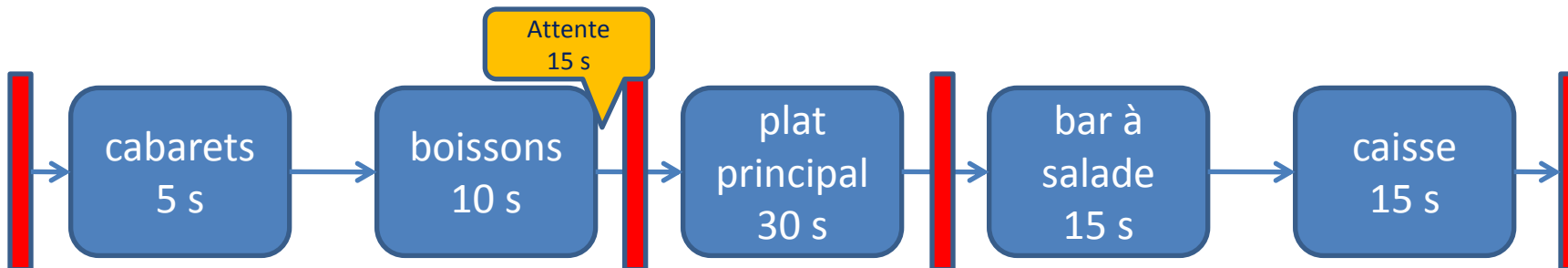
# La cafétéria pipelinée



Original:  
période d'horloge 75 s  
latence 1 cycle = 75 s  
débit 48 clients par heure



Pipeline naïf:  
1 client par station  
période d'horloge 30 s  
latence 5 cycles = 150 s  
débit 120 clients par heure

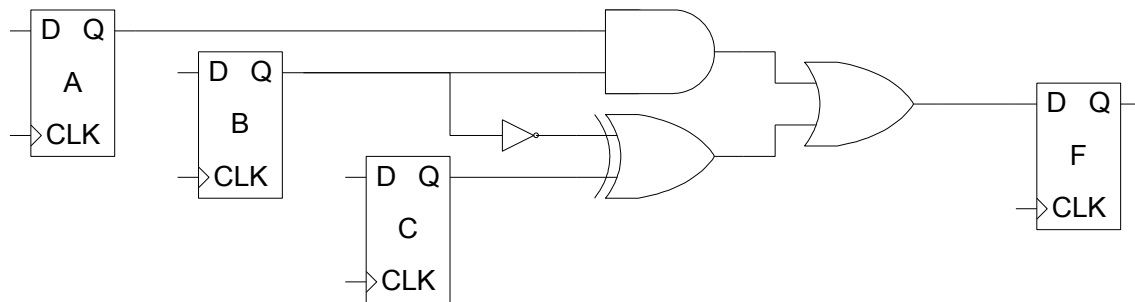


Pipeline ajusté:  
1 client par station  
période d'horloge 30 s  
latence 3 cycles = 90 s  
débit 120 clients par heure

# Architecture à pipeline pour circuit numérique

- Le pipeline est une technique puissante pour augmenter la fréquence d'horloge d'un système et son débit.
- La période minimale de l'horloge est donnée par :

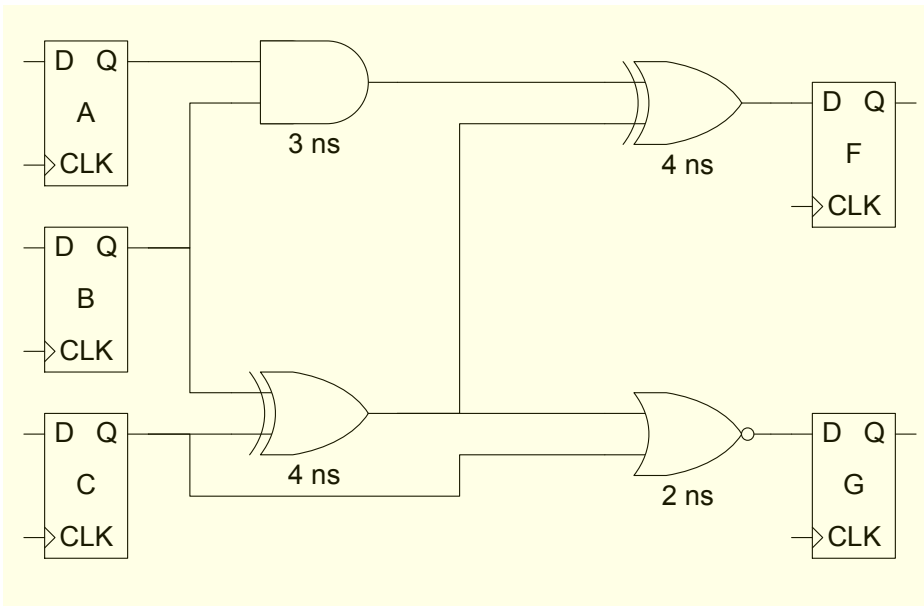
$$T_{\min} = t_d + t_{comb} + t_{prop} + t_{su}$$



- Pour diminuer  $T_{\min}$ , il faut réduire la somme des termes.
  - $t_d, t_{su}$  (délai de propagation et de préparation des bascules):  
Ces valeurs sont en général fixes.
  - $t_{comb}$  (délai de propagation de la logique combinatoire):  
Une architecture à pipeline s'attaque aux délais de la logique combinatoire en décomposant le chemin critique.
  - $t_{prop}$  (délai de propagation des interconnexions):  
Il faut réduire la longueur des interconnexions en disposant judicieusement les composantes sur le chemin critique.

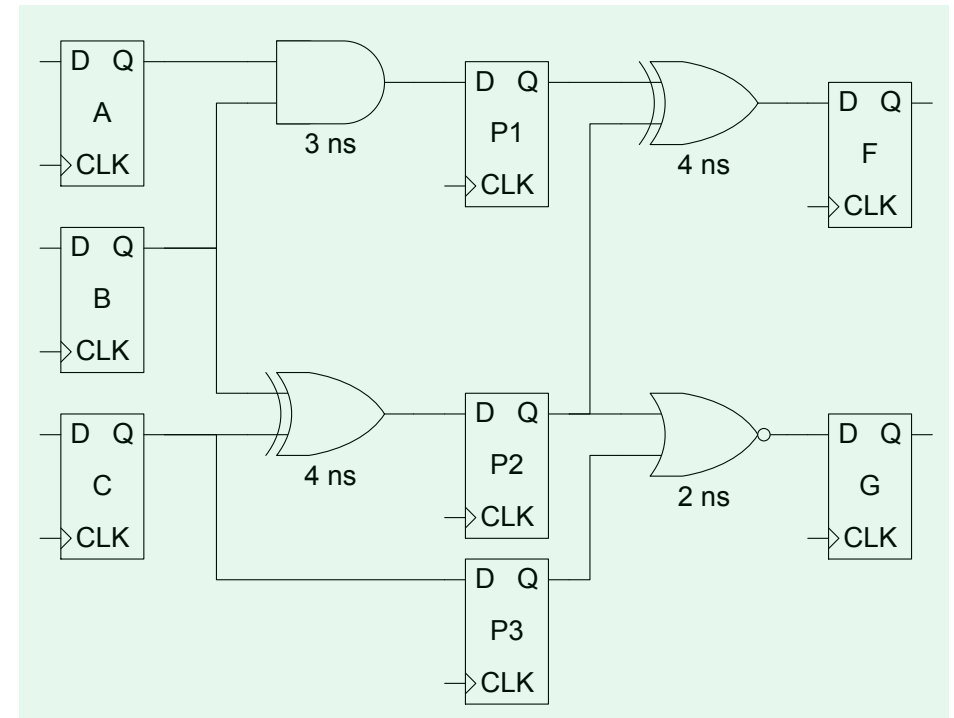
# Architecture à pipeline pour circuit numérique

Supposons  $(t_d, t_{su}, t_{prop}) = (2 \text{ ns}, 1 \text{ ns}, 0 \text{ ns})$



$$T_{\min} = t_d + t_{comb} + t_{prop} + t_{su}$$

$$T_{\min} = 2 + 8 + 0 + 1 = 11 \text{ ns}$$



$$T_{\min 1} = 2 + 4 + 0 + 1 = 7 \text{ ns}$$

$$T_{\min 2} = 2 + 4 + 0 + 1 = 7 \text{ ns}$$

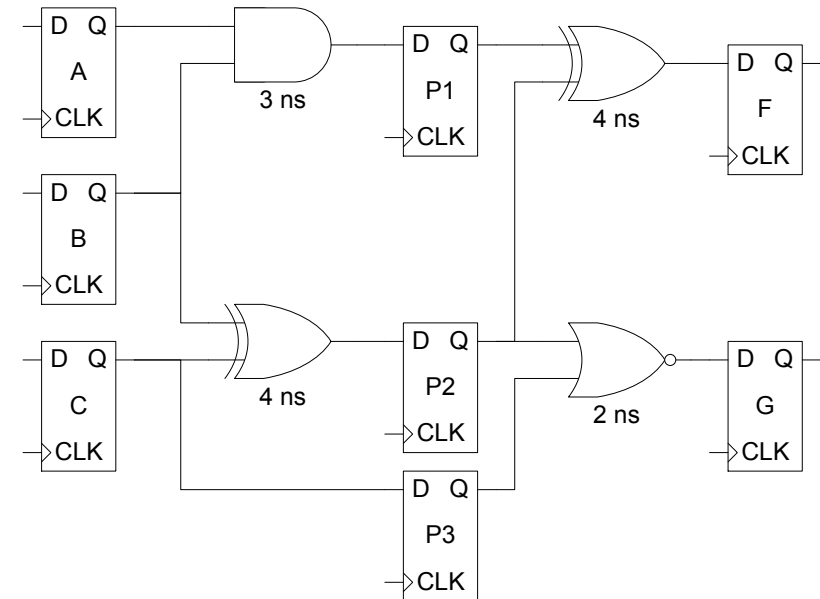
$$T_{\min} = 7 \text{ ns}$$

Le pipeline vient avec une pénalité.  
On ne peut pas réduire  $T_{\min}$  de 50%. On ne peut pas réduire ni  $t_d$  ni  $t_{su}$ .

# Le pipeline en pratique

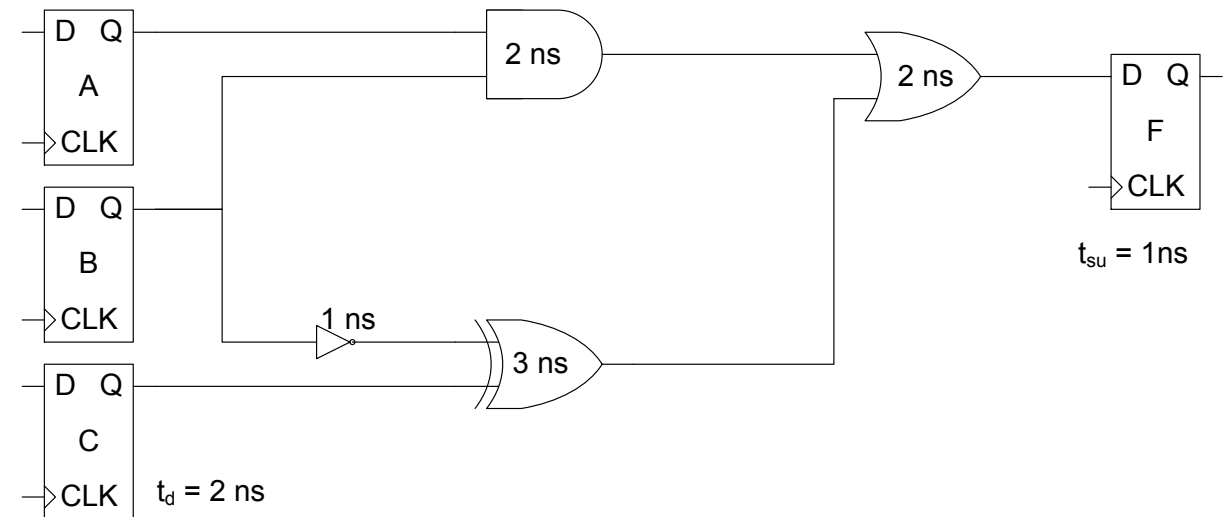
- Le désavantage principal d'une architecture avec pipeline est son coût élevé en matériel.
- Quand on introduit un étage de pipeline il faut synchroniser tous les signaux de cet étage, même s'ils ne sont pas dans le chemin critique (p. ex., dans le circuit présent, de la bascule C à la porte NON-OU).
- Pour les FPGA, la présence d'une très grande quantité de bascules prédéfinies à l'intérieur des blocs de logique programmable rend possible et efficace l'utilisation d'architectures à pipeline.

- On ne peut réduire le délai  $t_d$  ni le temps de préparation  $t_{su}$  des étages de pipeline.
- Il faut bien balancer les nouveaux chemins. La fréquence maximale d'opération du circuit est limitée par le chemin le plus lent du circuit.



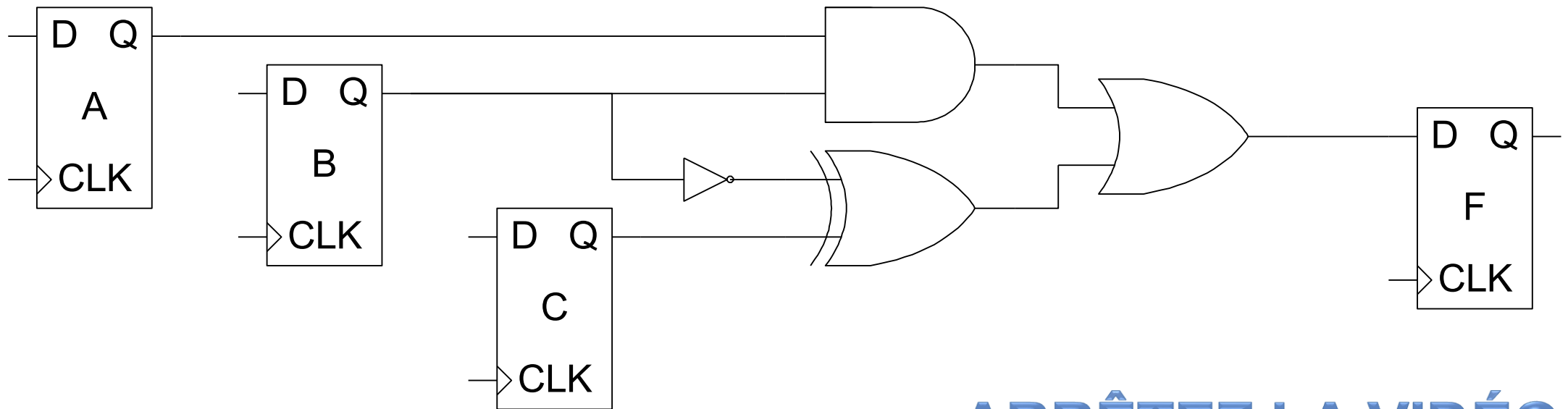
# Le pipeline: stratégie

- Pour ajouter un étage de pipeline, on peut suivre la stratégie suivante:
  1. Représenter le circuit avec les signaux qui vont de gauche à droite.
  2. Si une boucle de rétroaction est présente, la décomposer en répétant le registre cible à la droite du registre source.
  3. Identifier le chemin critique et placer un registre de façon à le couper en formant deux moitiés avec les délais les plus semblables possible.
  4. Former une courbe verticale qui traverse le circuit et qui passe à travers le registre de pipeline ajouté.
  5. Ajouter un registre de pipeline à chaque intersection de la courbe et d'un signal horizontal.



# Exercice de pipeline #1

- Les bascules et les portes logiques INV, ET, OU et OUX ont des délais de propagation de 2, 1, 2, 2 et 3 ns, respectivement. Les bascules ont un temps de préparation de 1 ns.
- Insérez des registres de pipeline dans le circuit suivant pour atteindre une fréquence d'horloge de 140 MHz.

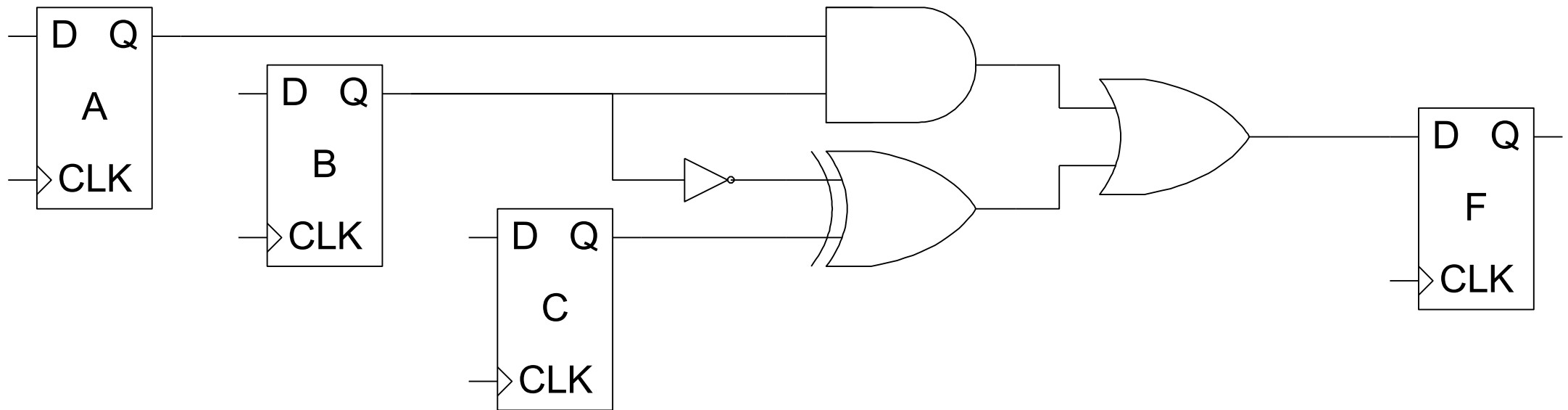


**ARRÊTEZ LA VIDÉO  
ET FAITES L'EXERCICE!**



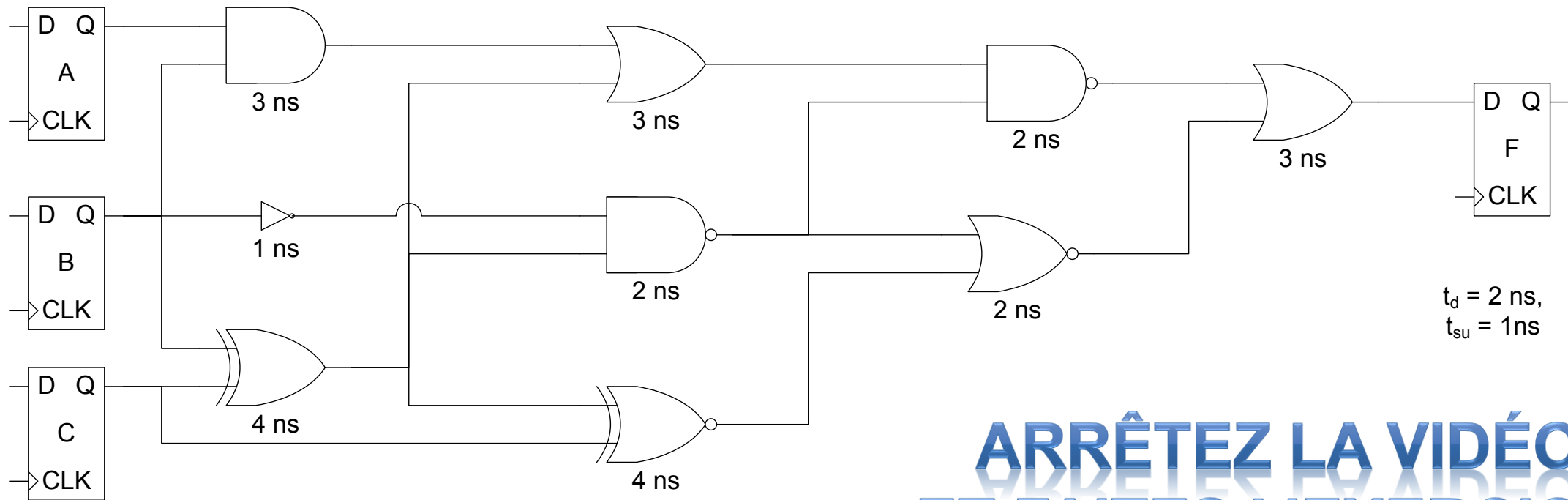
# Exercice de pipeline #1

- Les bascules et les portes logiques INV, ET, OU et OUX ont des délais de propagation de 2, 1, 2, 2 et 3 ns, respectivement. Les bascules ont un temps de préparation de 1 ns.
- Une fréquence d'horloge de 140 MHz correspond à une période de 7.14 ns.



## Exercice de pipeline #2

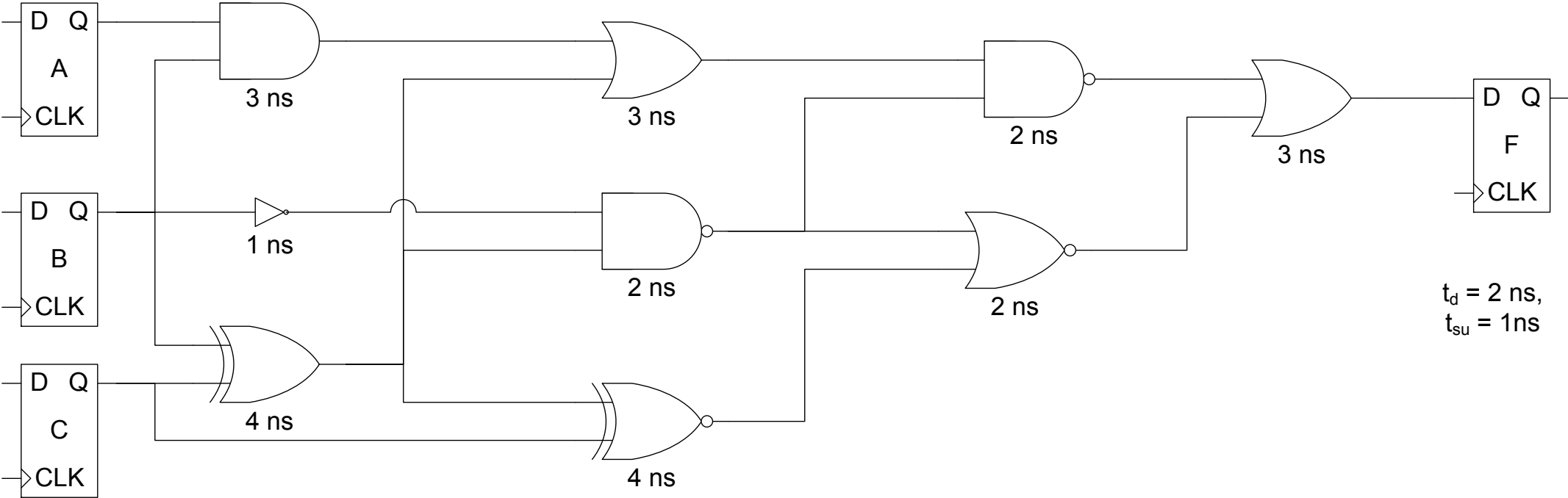
- Insérez des registres de pipeline dans le circuit suivant pour atteindre une fréquence d'horloge de 140 MHz.



**ARRÊTEZ LA VIDÉO  
ET FAITES L'EXERCICE!**

# Exercice de pipeline #2

- Une fréquence d'horloge de 140 MHz correspond à une période de 7.14 ns.



# Vous devriez maintenant être capable de ...

---

- Pipeliner un circuit numérique pour en augmenter le débit. Synchroniser correctement le pipeline en tenant compte du principe d'équilibre entre les chemins. Donner la nouvelle latence de calcul, le nouveau débit et les coûts. (B3)

Code	Niveau ( <a href="http://fr.wikipedia.org/wiki/Taxonomie_de_Bloom">http://fr.wikipedia.org/wiki/Taxonomie_de_Bloom</a> )
B1	Connaissance – mémoriser de l'information.
B2	Compréhension – interpréter l'information.
B3	Application – confronter les connaissances à des cas pratiques simples.
B4	Analyse – décomposer un problème, cas pratiques plus complexes.
B5	Synthèse – expression personnelle, cas pratiques plus complexes.