

Le langage Ladder Diagrams (LD)

1 – PRINCIPE DE FONCTIONNEMENT

2 – SYMBOLES DU LANGAGE LD

3 – APPELS DE FONCTIONS

4 – LIMITATIONS ET EXTENSIONS

1 – Principe de fonctionnement

2 – SYMBOLES DU LANGAGE LD

3 – APPELS DE FONCTIONS

4 – LIMITATIONS ET EXTENSIONS

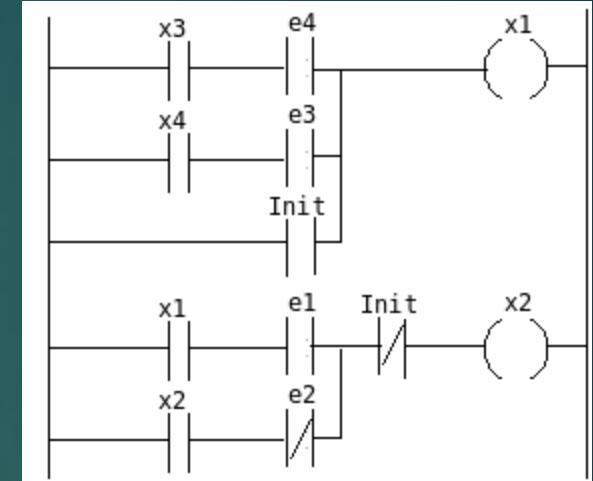
Introduction

- ▶ Ladder = échelle
- ▶ Le Ladder Diagram est un langage de programmation graphique



Ladder

Crédit image : Hedwig von Ebbel
Licence : domaine public
https://commons.wikimedia.org/wiki/File:Leiter_ladder.jpg

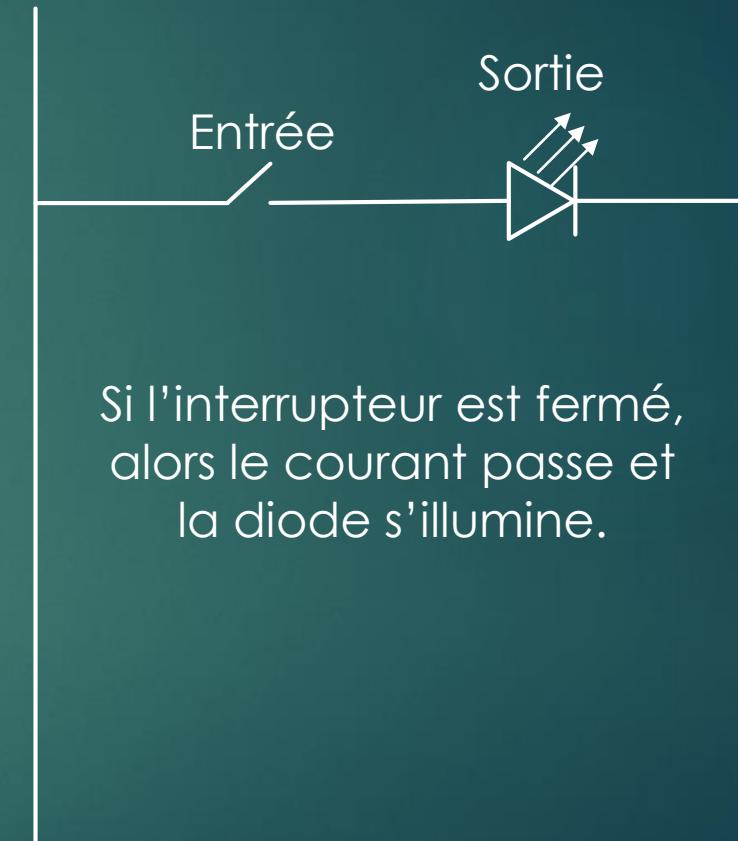


Ladder diagram

Crédit image : SergeMoutou
Licence : CC-BY-SA-3.0
<https://commons.wikimedia.org/wiki/File:AutomLadder1.png>

Principe de fonctionnement

- ▶ Analogie avec un système électrique
 - ▶ « L'alimentation » arrive par la gauche
 - ▶ Interrupteurs laissant passer le « courant » ou non
 - ▶ « Actionneurs » réagissant à la présence de courant



Si l'interrupteur est fermé,
alors le courant passe et
la diode s'illumine.

VCC

GND

Fonctions logiques fondamentales

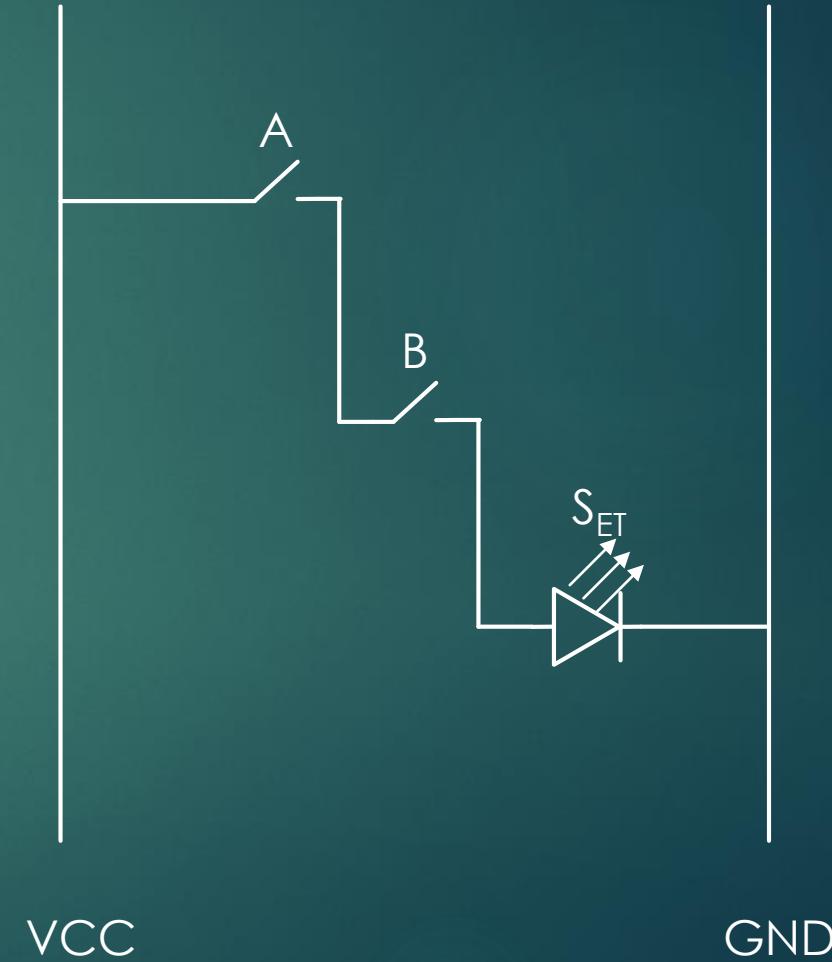
5

▶ Fonction ET

- ▶ Sur ce schéma électrique, comment câbler pour que la diode S_{ET} s'active lorsque les interrupteurs A et B sont tous deux actifs, et uniquement dans ce cas là ?

→ En série

A	B	S_{ET}
0	0	0
0	1	0
1	0	0
1	1	1



Fonctions logiques fondamentales

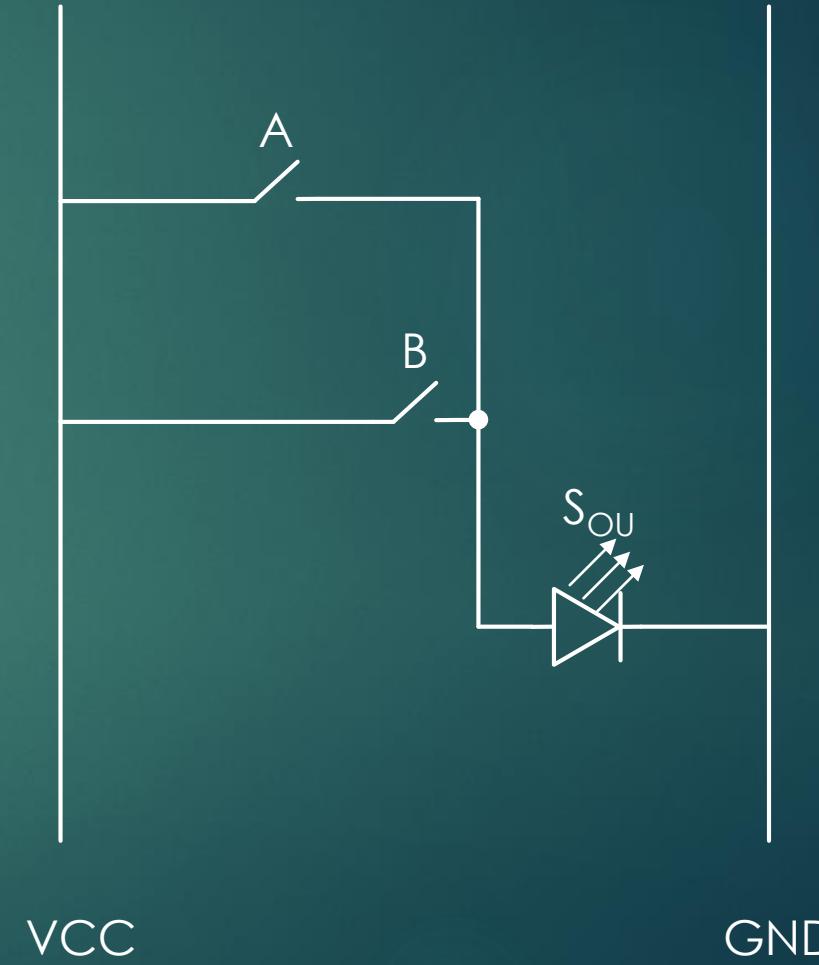
6

▶ Fonction OU

- ▶ Sur ce schéma électrique, comment câbler pour que la diode S_{OU} s'active dès que l'un des interrupteurs A ou B est actif ?

➔ En parallèle

A	B	S_{OU}
0	0	0
0	1	1
1	0	1
1	1	1



1 – PRINCIPE DE FONCTIONNEMENT

2 – Symboles du langage LD

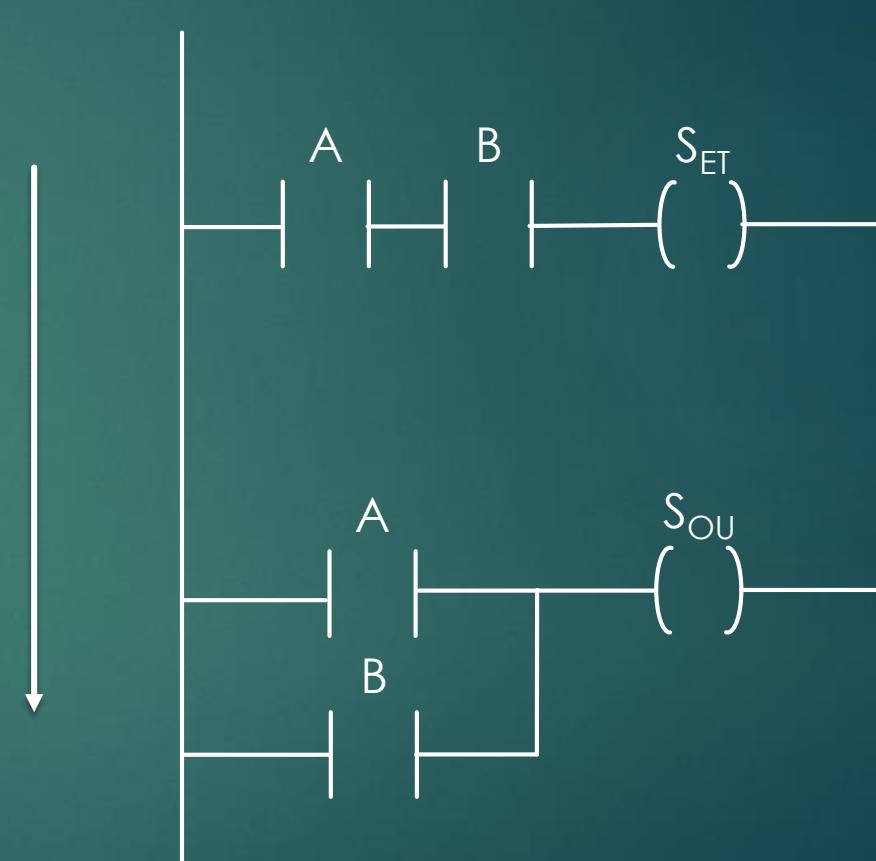
3 – APPELS DE FONCTIONS

4 – LIMITATIONS ET EXTENSIONS

Exécution des diagrammes

8

- ▶ L'exécution est séquentielle, ligne après ligne, comme dans un langage de programmation textuel



Symboles des diagrammes ladder : entrées

- ▶ Interrupteur / contact normalement ouvert
 - ▶ Le courant passe lorsque la variable est active (à 1)
- ▶ Interrupteur / contact normalement fermé
 - ▶ Le courant passe lorsque la variable est inactive (à 0)



Les variables peuvent être des variables d'entrée ou des variables internes

Symboles des diagrammes ladder : sorties

10

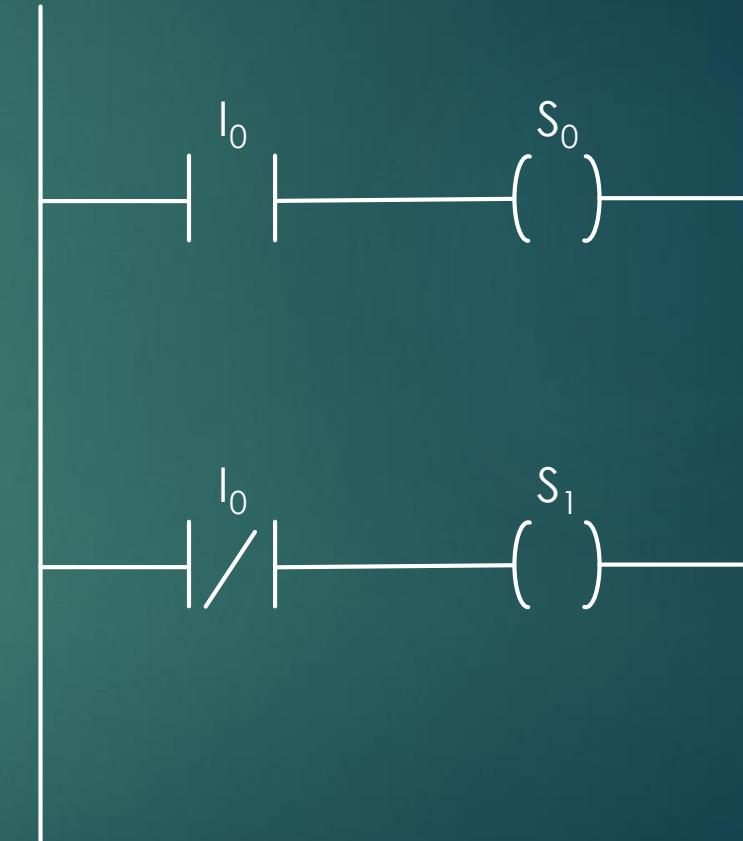
- ▶ Bobine normalement ouverte
 - ▶ La variable est activée (reçoit la valeur 1) lorsque le courant arrive à gauche
 - ▶ Elle revient à zéro lorsqu'il n'y a plus de courant
- ▶ Bobine normalement fermée
 - ▶ La variable est activée (reçoit la valeur 1) lorsque il n'y a pas de courant qui arrive à gauche
 - ▶ Elle passe à 0 dès qu'il y a du courant



Les variables peuvent être des variables internes ou des variables de sortie

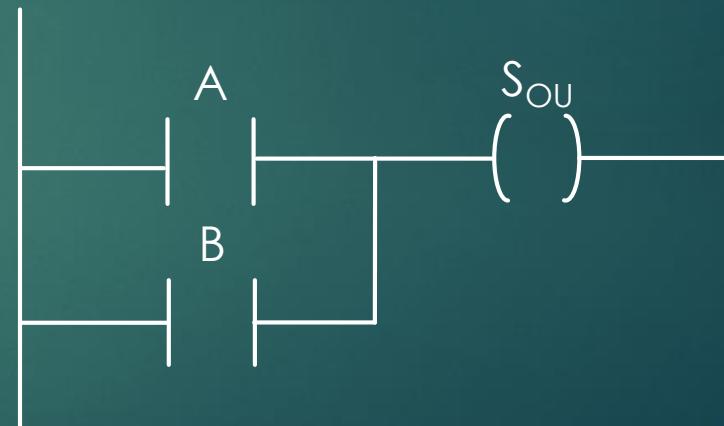
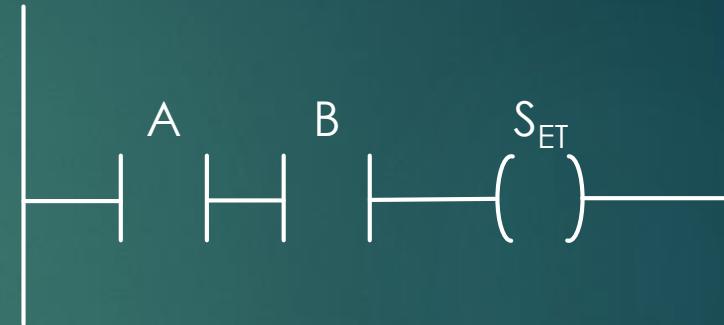
Exemple

- ▶ La sortie S_0 s'active lorsque l'entrée I_0 est activée. Elle revient à 0 lorsque l'entrée est désactivée.
- ▶ La sortie S_1 s'active lorsque l'entrée I_0 est désactivée. Elle revient à 0 lorsque l'entrée est activée.



Application au Ladder

- ▶ Dessiner un schéma Ladder réalisant la fonction ET et un réalisant la fonction OU
 - ▶ Entrées
 - ▶ A
 - ▶ B
 - ▶ Sorties
 - ▶ S_{ET}
 - ▶ S_{OU}



Exercice

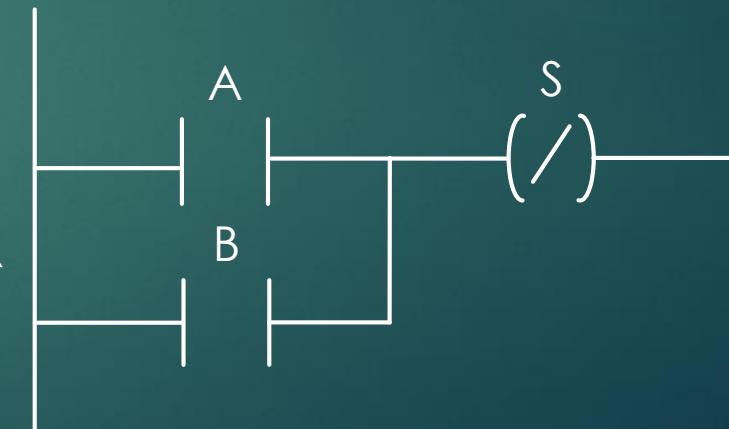
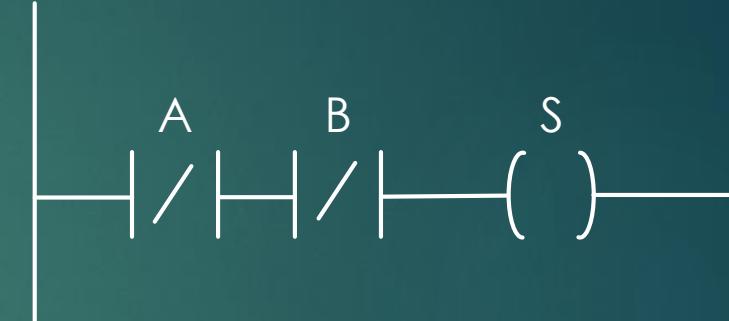
- ▶ Dessiner deux schémas Ladder différents correspondant à l'équation logique suivante :

- ▶ $S = \overline{A + B}$

- ▶ Schéma 1 : $S = \overline{A} \cdot \overline{B}$

- ▶ Schéma 2 : $\overline{S} = A + B$

- Transformer l'équation en deux autres équations équivalentes
 - Utiliser la loi de Morgan



Symboles des diagrammes ladder : entrées à détection de front

- ▶ Contact POSITIVE EDGE
 - ▶ Le contact laisse passer le courant uniquement si la variable vaut 1 alors qu'elle valait 0 lors du cycle précédent.
- ▶ Contact NEGATIVE EDGE
 - ▶ Le contact laisse passer le courant uniquement si la variable vaut 0 alors qu'elle valait 1 lors du cycle précédent.



Symboles des diagrammes ladder : sorties maintenues

► Bobine SET

- La variable est activée (reçoit la valeur 1) lorsque le courant arrive à gauche. Sa valeur est inchangée lorsque le courant s'arrête.

Variable
—(s)—

► Bobine RESET

- La variable est désactivée (reçoit la valeur 0) lorsque le courant arrive à gauche. Sa valeur est inchangée lorsque le courant s'arrête.

Variable
—(R)—

Exercice

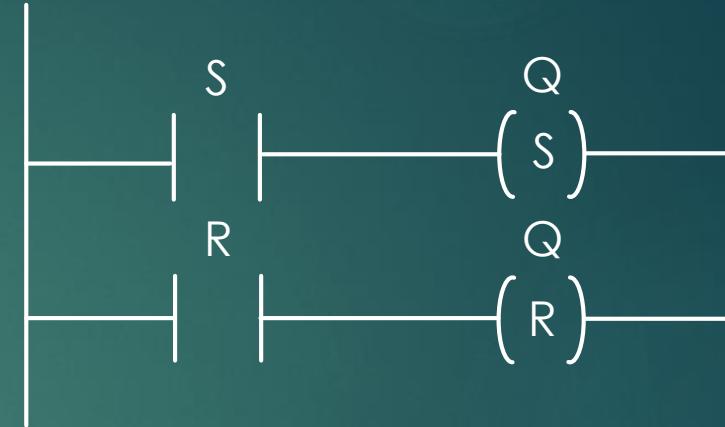
16

- ▶ Dessiner un schéma Ladder correspondant au fonctionnement d'une bascule RS

R	S	Q_n
0	0	Q_{n-1}
0	1	1
1	0	0
1	1	(interdit)

Faire en sorte que la valeur ne change pas dans ce cas

Réponse simple



Réponse complète

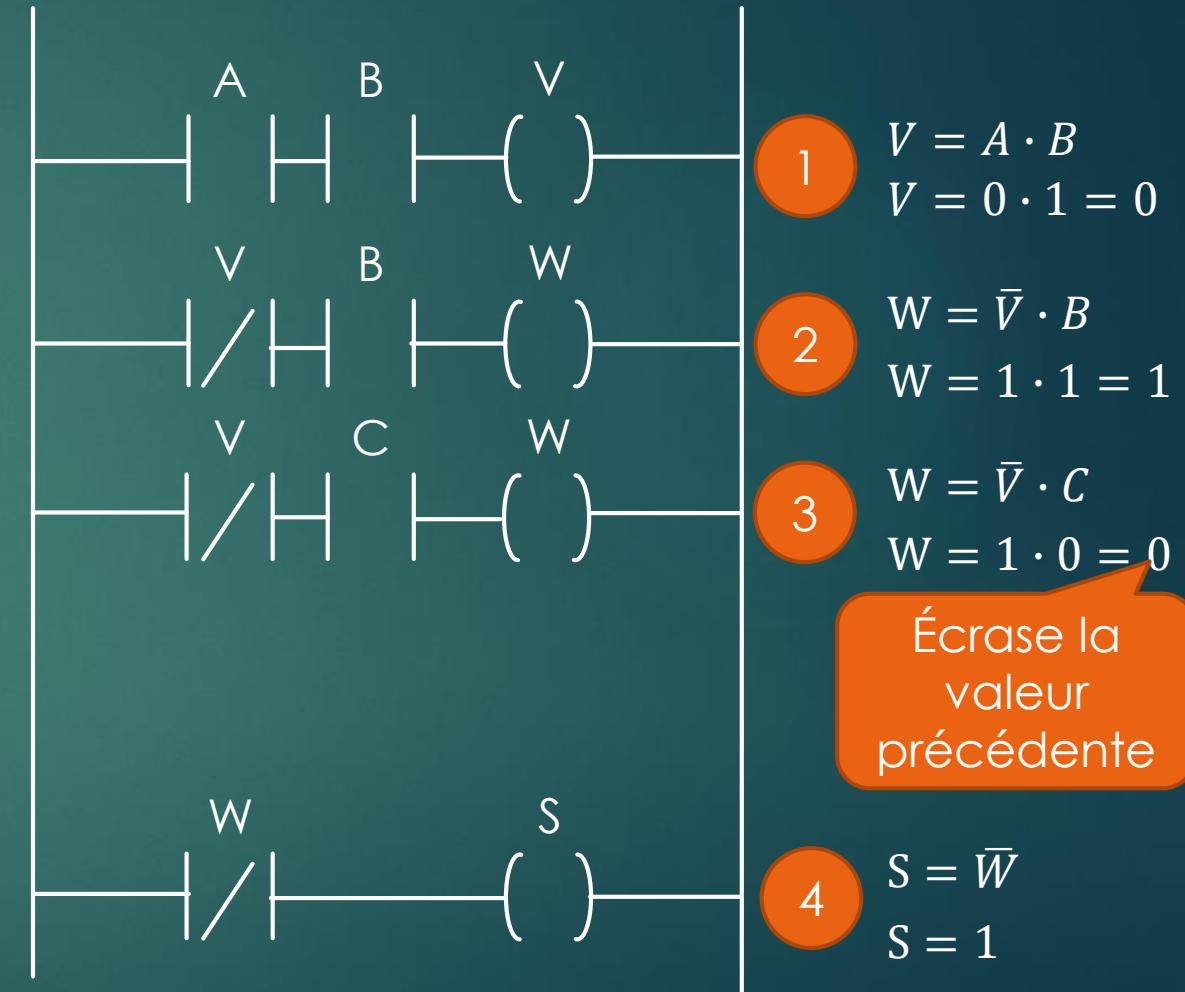


Ajouter l'autre variable inversée assure que le circuit ne sera pas fermé si les deux variables sont actives

Exercice

- ▶ Quelle sera la valeur de la variable S après l'exécution de ce diagramme ?
- ▶ On suppose les valeurs suivantes des entrées :

A	B	C
0	1	0



1 – PRINCIPE DE FONCTIONNEMENT

2 – SYMBOLES DU LANGAGE LD

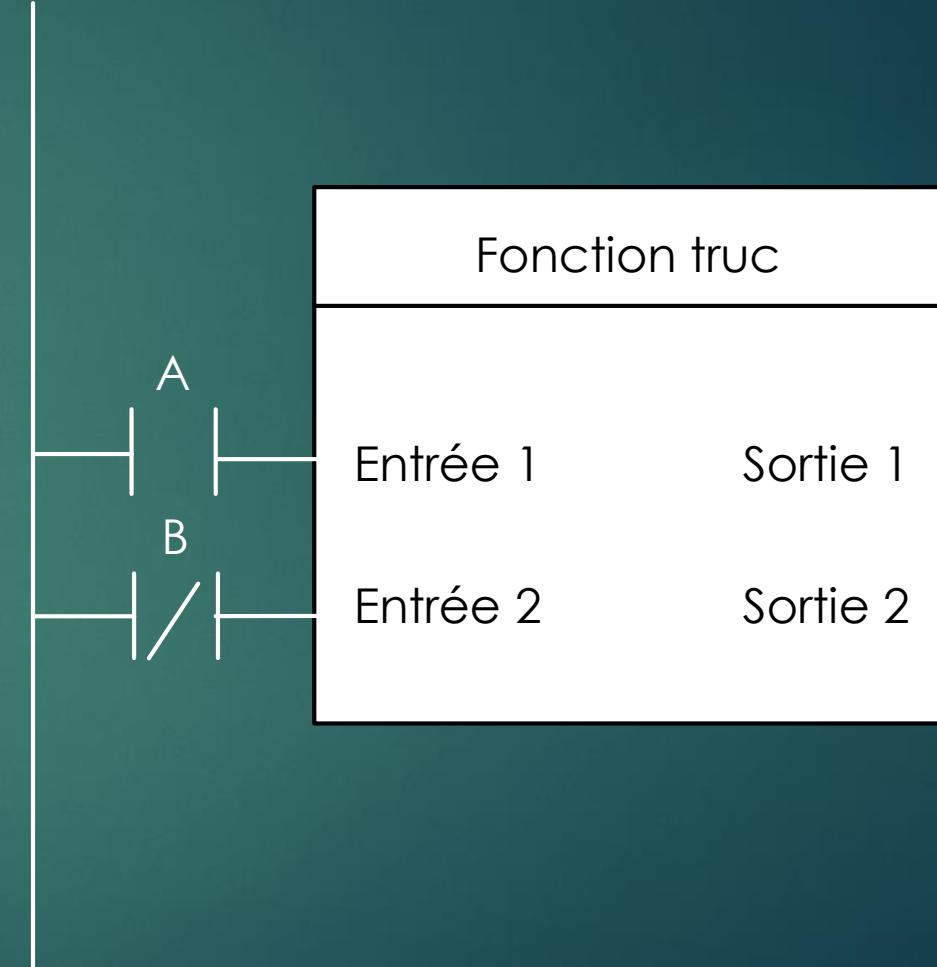
3 – Appels de fonctions

4 – LIMITATIONS ET EXTENSIONS

Fonctions

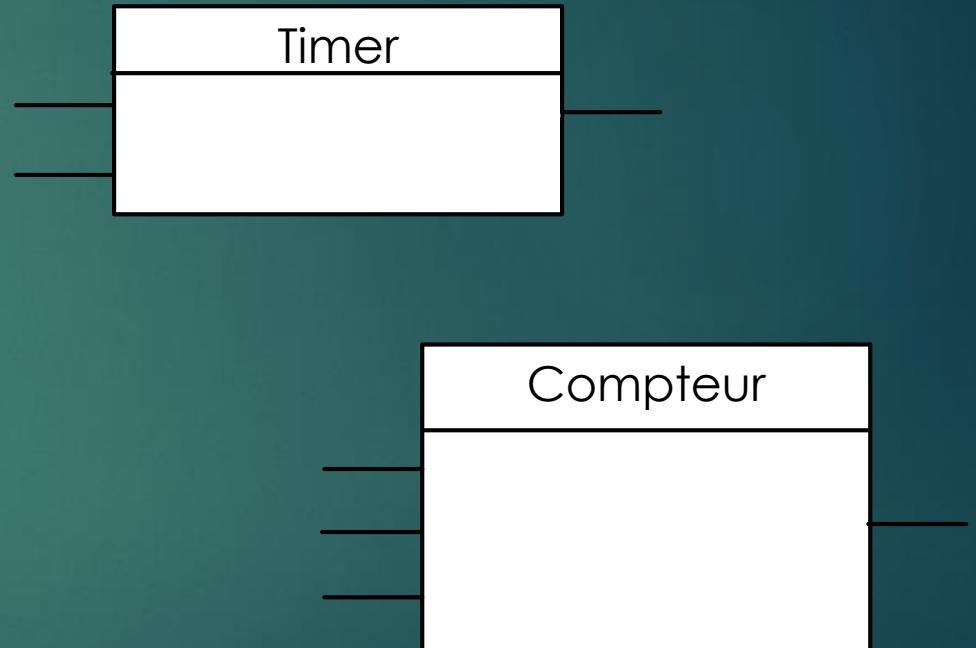
19

- ▶ En Ladder, les fonctions sont représentées par des blocs ayant
 - ▶ Des entrées
 - ▶ Des sorties
 - ▶ Un nom
- ▶ Ça ne vous rappelle rien, ça ?
- ▶ On peut les intégrer directement dans un diagramme



Fonctions disponibles

- ▶ Il existe de nombreuses fonctions standard
 - ▶ Timer
 - ▶ Compteur / décompteur
 - ▶ Bascule
 - ▶ Etc.
- ▶ Il faut se référer à la documentation du logiciel que l'on utilise pour savoir quelles fonctions sont disponibles
- ▶ Il est aussi possible d'appeler des fonctions que l'on a soi-même créé !



1 – PRINCIPE DE FONCTIONNEMENT

2 – SYMBOLES DU LANGAGE LD

3 – APPELS DE FONCTIONS

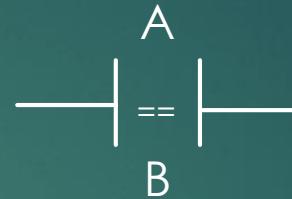
4 – Limitations et extensions

- ▶ Le langage LD est très bon pour représenter des équations booléennes, mais...
 - ▶ Complexité dès que l'on travaille sur des vecteurs de bits
 - ▶ Pas de boucle : tout doit être dupliqué !
 - ▶ Pas adapté pour travailler sur des entiers
 - ▶ Dans ce cadre, les fournisseurs d'API ajoutent parfois des extensions

Extensions (Siemens)

23

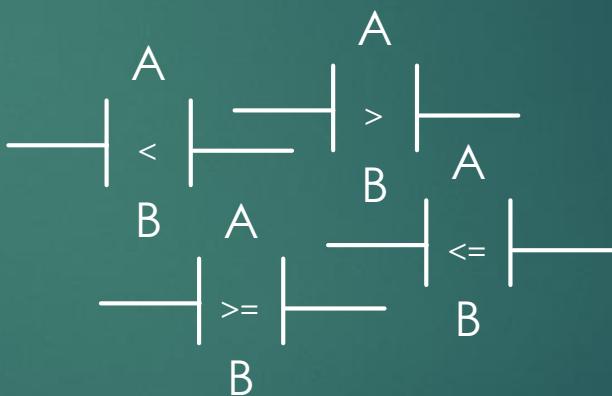
- ▶ Dans le langage CONT, Siemens définit les opérateurs suivants
 - ▶ Opérateurs de comparaison
 - ▶ Opérateurs arithmétiques
 - ▶ Etc.
- ▶ Voir la doc du logiciel pour plus de détails



Égalité : le courant passe si A est égal à B



Différence : le courant passe si A est différent de B



Comparaison

