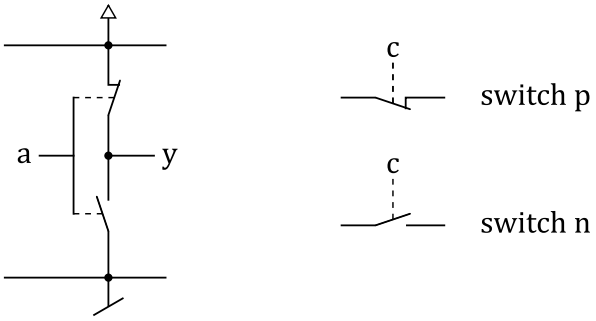
États Logiques

Exercices Conception numérique

1 | LST - Portes logiques ne fournissant qu'un état

1.1 Circuit à interrupteurs

Déterminer la relation entre l'entrée et la sortie pour le circuit de la figure suivante.

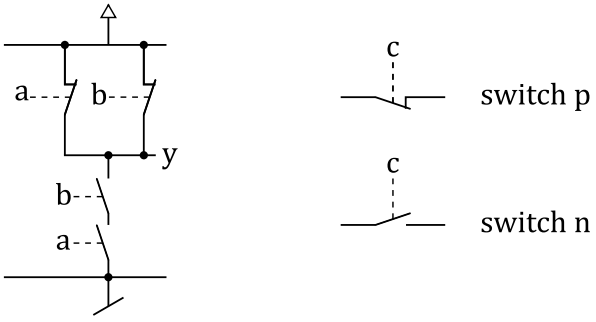


Un interrupteur n est fermé quand le signal de commande est à `'1'`. Un interrupteur p est ouvert quand le signal de commande est à `'1'`.

lst/one-state-01-01

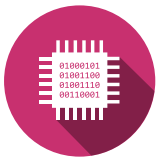
1.2 Circuit à interrupteurs

Déterminer la relation entre l'entrée et la sortie pour le circuit de la figure suivante.



Un interrupteur n est fermé quand le signal de commande est à `'1'`. Un interrupteur p est ouvert quand le signal de commande est à `'1'`.

lst/one-state-01-02



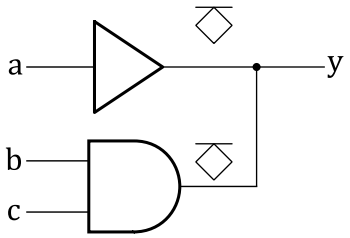
1.3 Circuit à interrupteurs

A l'aide d'interrupteurs uniquement, proposer le schéma d'une porte OU-exclusif à 2 entrées.

lst/one-state-01-03

1.4 Circuit à source ouverte

Compléter le circuit de la figure ci-contre.
Donner la table de vérité de la sortie **y** en fonction des 3 entrées.



lst/one-state-02-01

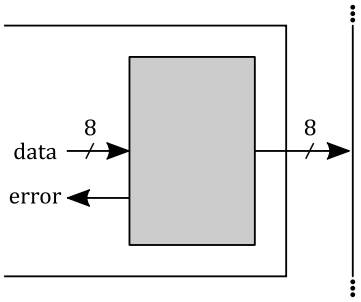
1.5 Alarme

Proposer le schéma d'un circuit de protection contre les incendies pour un bâtiment.
Le bâtiment comporte 16 capteurs de fumée répartis dans les différentes pièces et reliés par un câble à 3 fils: 2 pour l'alimentaton et 1 pour la transmission de l'information. L'activation de n'importe lequel des capteurs doit provoquer l'enclenchement d'une sirène.
Un capteur fournit un `'1'` en cas de détection de fumée. La sirène est enclenchée par le passage à `'1'` de son signal de commande.

lst/one-state-02-02

1.6 Détection de collision

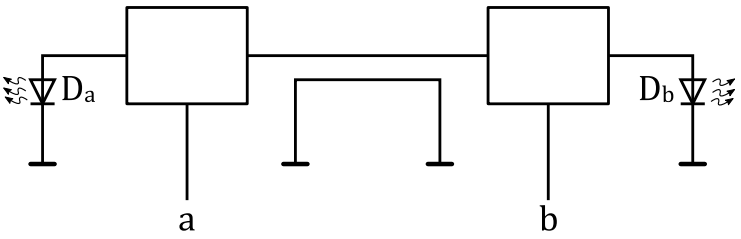
Dessiner le schéma d'un interface de bus où plusieurs composants sont capable d'écrire une donnée de 8 bits sur un bus commun et d'en vérifier l'intégrité.
Si la donnée sur la ligne est différente de la donnée voulue, l'interface de bus retourne un signal d'erreur au composant pour lui indiquer le problème de transmisssion.
Proposer une technique de codage pour permettre de fixer les priorités des données mises sur le bus.



lst/one-state-02-03

1.7 Transmission d'information sur un seul fil

Dessiner le schéma interne des blocs du circuit représenté à la figure suivante.



a	b	D_a	D_b
0	0	0	0
0	1	1	0
1	0	0	1
1	1	0	0

lst/one-state-02-04

2 | LST - Portes logiques avec sortie à haute impédance

2.1 Branchement de périphériques en série

Proposer le schéma de raccordement de composants à un même bus de données.
Le système comprend 3 composants qui fournissent chacun des données de 8 bits. Pour les sélectionner, le maître du système fournit une adresse codée sur 2 bits. Le fonctionnement est le suivant:

- si l'adresse vaut 0, aucun composant ne transmet de données,
- si l'adresse vaut 1, le composant 1 transmet sa donnée,
- si l'adresse vaut 2, le composant 2 transmet sa donnée,
- si l'adresse vaut 3, le composant 3 transmet sa donnée.

lst/hiz-01

2.2 Réalisation de fonction avec des circuits haute impédance

Réaliser un multiplexeur de 2 à 1 à l'aide des tampons avec sortie haute impédance (tri-state).

lst/hiz-02

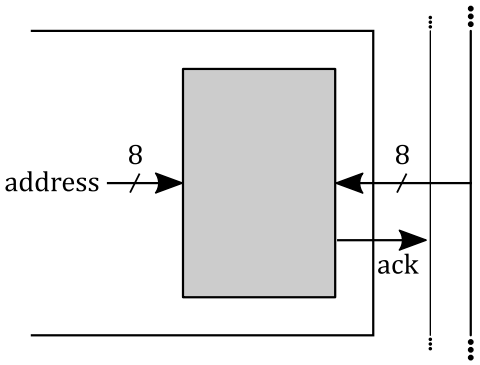
2.3 Réalisation de fonction avec des circuits à haute impédance

Proposer le schéma d'un inverseur à collecteur ouvert réalisé à l'aide d'inverseurs à sortie à haute impédance.

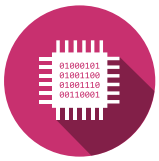
lst/hiz-03

2.4 Détection de collision

Proposer le schéma d'un interface de bus où les composants annoncent leur présence à l'apparition de leur adresse.
Si l'adresse sur le bus externe est celle du composant, l'interface de bus signale sa présence sur la ligne **ack** (acknowledge) commune à tous les composants.
Proposer une technique pour permettre d'ajouter un à un des composants identiques dans le système en cours de fonctionnement sans avoir à les configurer manuellement.

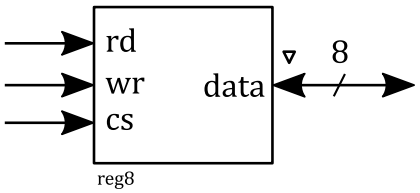


lst/hiz-04

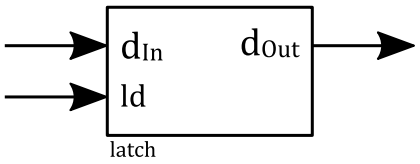


2.5 Registre à bus bidirectionnel

Donner le schéma interne d'un registre de 8 bits à bus de données bidirectionnel.
Le registre est relié à un bus de données bidirectionnel. Pour une écriture, une donnée de 8 bits est mise sur le bus de données et les commandes **`cs`** (chip select) et **`wr`** (write) sont activées. Pour une lecture, le commandes **`cs`** et **`rd`** (read) sont activées et le registre place son contenu sur le bus de données.



Le registre est composé de 8 éléments de mémoire (latch) comme celui représenté à la figure ci-contre. Lorsque l'entrée **`ld`** (load) est active, la donnée d_{In} (data in) est chargée dans l'élément de mémoire, lequel la transfère aussitôt à sa sortie d_{Out} (data out). L'élément de mémoire conserve sa donnée inchangée tant que **`ld`** est inactif.
A l'aide de ce composant, donner le schéma d'un mémoire vive de 4 registres.



lst/hiz-05