

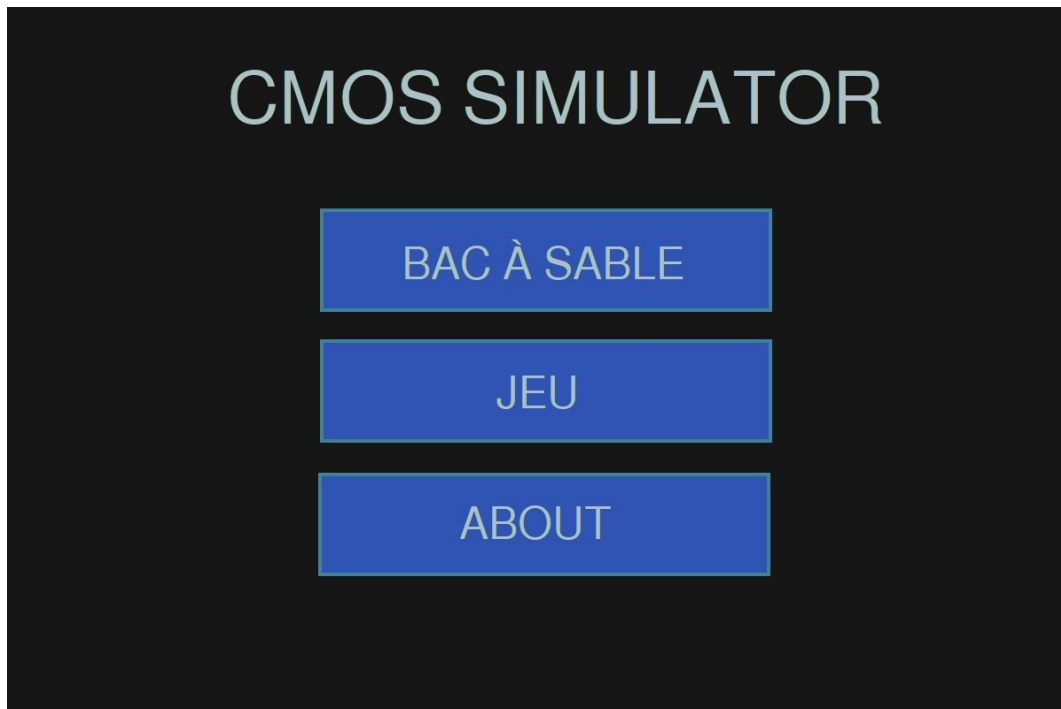
CAHIER DES CHARGES

Nous envisageons de réaliser un programme ludo-éducatif de simulation de circuits logiques combinatoires et séquentiels, dont le cahier des charges constitue le présent document.

Dans le cadre d'un projet informatique, le cahier des charges correspond peu ou prou à l'interface utilisateur du programme final. Nous allons donc nous attacher à décrire précisément les différents éléments de cette interface.

1) [optionnel] un logo / une animation lors du lancement du programme et du chargement de ses ressources.

2) Une page d'accueil contenant des liens vers les activités ouvertes à l'utilisateur, notamment :

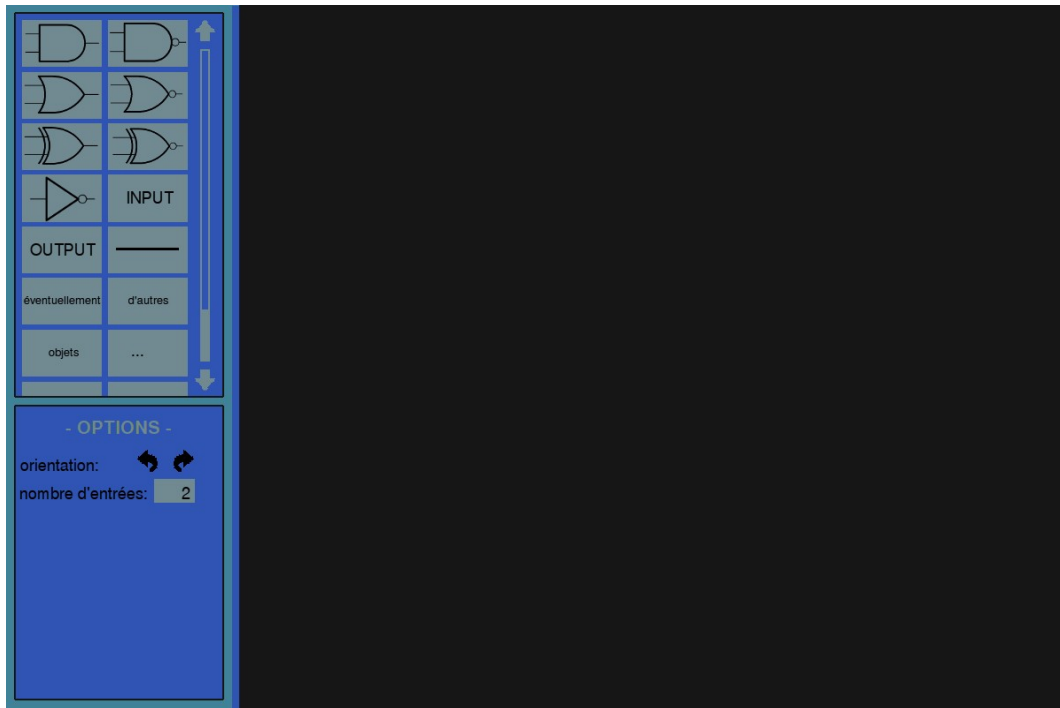


(3) Le mode "*bac à sable*".

(4) [optionnel] Le mode "*jeu*".

Un clic sur un de ces liens permettra à l'utilisateur de quitter la page d'accueil et de se rendre sur l'activité correspondante.

3) Bac à sable :



Cette activité est la partie centrale et indispensable du programme. Elle consiste à présenter à l'utilisateur :

- (a) une "boite à outils" contenant des circuits, des portes logiques, des entrées et sorties.
- (b) un espace de travail occupant la majeure partie de l'écran, dans lequel il peut ajouter et relier entre eux les objets de la boite à outil afin de créer des circuits logiques.
- (c) une zone permettant d'agir sur l'objet actuellement sélectionné pour en modifier les caractéristiques, par exemple le nombre d'entrées.
- (d) une interface permettant la simulation du circuit logique créé par l'utilisateur, en choisissant les valeurs en entrée, et en lisant les valeurs en sortie.

a) La boite à outil est un composant visuel dans lequel les portes logiques sont représentées par leur représentation schématique usuelle (le triangle du NOT, le demi-ovale du AND, etc...). Cette boite à outil peut aussi contenir les circuits préalablement sauvegardés par l'utilisateur, afin de pouvoir les réutiliser.

Les objets présent à minima sont les portes basiques : ET, OU, NON-ET, NON-OU, NON, OU EXCLUSIF.

La boite à outils comprend un mécanisme permettant de classer un grand nombre d'objets, par exemple des onglets "portes de base", "cellules mémoires", "mux/demux", "circuits utilisateurs".

La boite à outils comprend un mécanisme permettant d'afficher plus d'outils que l'écran n'en contient, par exemple un mécanisme de scrolling.

[optionnel] Fournir plus d'outils, en particulier les flips-flops, mux et demux. Bien que l'utilisateur puisse les créer à l'aide du programme lui même, la performance d'outils intégrés sera supérieure à celle d'outils

simulés.

b) Par un mécanisme de sélection à déterminer, l'utilisateur prend des objets de la boîte à outils et les pose dans l'espace de travail; ils y sont repositionnables; on peut aussi les supprimer de l'espace de travail.

Un mécanisme permet de relier les uns aux autres les entrées et sorties de ces objets; une représentation visuelle de ce lien apparaît à l'écran sous la forme de segments de droites. L'utilisateur peut déplacer ces chemins pour clarifier la lecture de son circuit.

[optionnel] créer un algorithme optimisant les chemins empruntés par les liens entre entrées et sorties.

À titre d'exemple à minima, l'utilisateur place deux entrées dans l'espace de travail, une porte ET, et une sortie. Il peut ensuite relier les deux entrées aux entrées de la porte ET, et la sortie de cette porte à la sortie. Des traits apparaissent, matérialisant ces liaisons.

c) Cette zone permet en particulier de spécifier le nombre d'entrées d'un outil. Si l'on souhaite un ET entre quatre entrées, il sera plus propre de placer une porte ET à quatre entrées que trois portes ET à deux entrées.

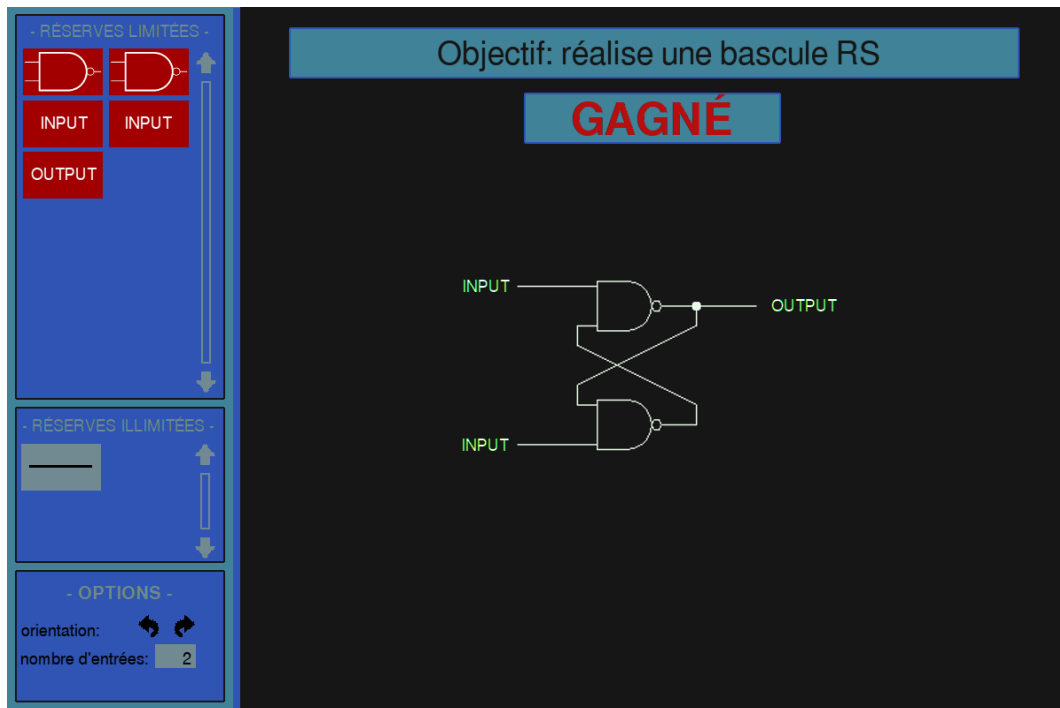
On peut aussi spécifier l'orientation (N S E O) d'un objet.

Cette zone permet aussi de spécifier la valeur 0 ou 1 présente dans une entrée.

d) Cette interface pourrait soit faire partie de l'espace de travail, par exemple en juxtaposant aux entrées et sorties leur valeur, soit faire l'objet d'une zone séparée.

Il reste à déterminer ce qui serait le plus lisible.

4) Le mode ludique :



Il reprend tous les éléments du mode *bac à sable*, à la différence suivante : l'utilisateur n'a pas accès à tous les objets, mais juste aux objets nécessaires à réaliser le challenge proposé par le *jeu*.

S'y ajoute une interface de dialogue, par exemple sous la forme de popup indiquant au joueur le but à réaliser.

S'y ajoute enfin un moyen pour le joueur de soumettre sa réalisation au programme, qui doit alors juger si elle répond au challenge proposé.

Exemple : le dialogue propose "réalisez un inverseur."

la boîte à outil contient une porte NAND, deux entrées et une sortie.

le joueur cable les deux entrées sur celles du NAND, et la sortie du NAND sur celle du circuit

le joueur soumet sa proposition

le programme lui indique qu'il a gagné et passe au niveau suivant

5) Un menu options :

On doit offrir un menu à l'utilisateur lui permettant à minima à tout moment de :

- a) Quitter le programme
- b) Sauver sa création en cours

[optionnel] il y a certainement beaucoup d'autres choses possibles