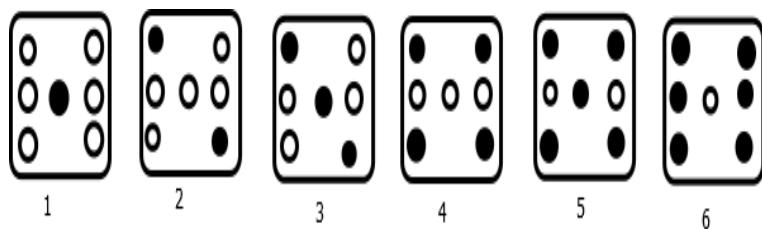
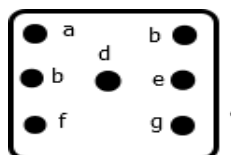


Exercice 1.

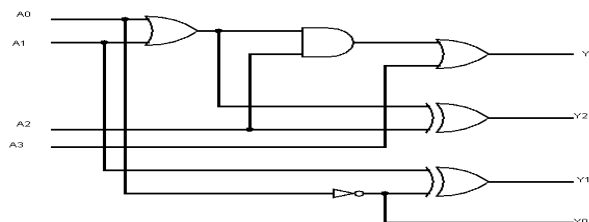
La figure suivante montre un dé électronique à diodes LED ; si on veut afficher 1, il faut allumer la diode d, si on veut afficher 2, on allumera les diodes a et g, comme le montre la figure ci-dessous.



On souhaite réaliser le circuit logique de commande pour allumer les diodes.

1. Etablir la table de vérité du circuit.
2. Déterminer les expressions simplifiées des sorties en fonction des entrées.
3. Tracer le logigramme.

Exercice 2. Analyser le circuit suivant



Exercice 3. Représenter un circuit logique qui implémente le Complément à 2 sur 4 bits en utilisant des demi-additionneurs en cascade.

Exercice 4.

On définit la fonction combinatoire f par sa table de vérité. Depuis cette table, synthétiser la fonction f avec.

1. MUX 8 \rightarrow 1.
2. MUX 4 \rightarrow 1.
3. MUX 2 \rightarrow 1.

a	b	c	f
0	0	0	1
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

Exercice 5. On définit la fonction combinatoire $f(a, b, c)$ par le tableau de Karnaugh suivant.

		ab			
		00	01	11	10
c	0	1	1	1	0
	1	0	1	1	0

1. Synthétiser cette fonction avec un multiplexeur $8 \rightarrow 1$.
2. Synthétiser cette fonction avec un multiplexeur $4 \rightarrow 1$.

Exercice 6. On considère le schéma bloc ci-contre d'un Mux 2 vers 1.

1. Faire la synthèse d'un multiplexeur 2 vers 1.
2. Réaliser le schéma bloc d'un multiplexeur 4 vers 1, en utilisant que trois multiplexeurs 2 vers 1.



Exercice 7. Soit f une fonction booléenne définie par la forme suivante

$$f(a, b, c, d) = \sum_m(0, 2, 5, 10, 13, 12, 15)$$

1. Réaliser f avec un Mux $16 \rightarrow 1$ et sans portes logiques.
2. Réaliser f avec un Mux $4 \rightarrow 1$ et des portes logiques.
3. Réaliser f avec un Mux $2 \rightarrow 1$ et des portes logiques.

Exercice 8. Synthétiser la fonction logique f définie par $f(a, b, c, d) = \sum_m(1, 3, 7, 9, 15)$, en utilisant un décodeur 3 parmi 8 et deux portes logiques NANDs, l'une à 2 entrées et l'autre à 5 entrées.

Exercice 9. On souhaite réaliser un transcodeur binaire-Gray qui permet de convertir le code binaire à 2 bits en code de Gray.

1. Dresser la table de vérité qui réalise le fonctionnement.
2. Déterminer les équations de sorties et simplifier.
3. Tracer le logigramme.

