

Compétences :

- ✓ Générer automatiquement le programme et l'implanter dans le système cible.
- ✓ Tester le fonctionnement.

1ère S si

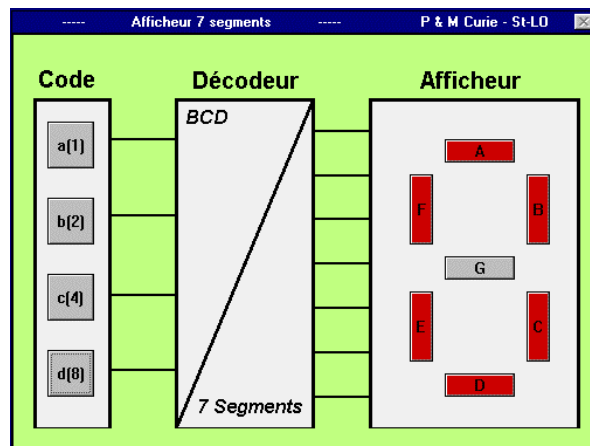
Pré-requis et documents ressources :

- Logique combinatoire.
- 📖 Documentation "Automgen"

Durée : 2 h

Matériel : Poste informatique équipé du logiciel "Automgen"

AFFICHEUR NUMERIQUE



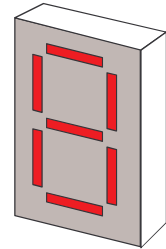
Axe de la formation :

C - Principe et comportement des produits
C2 - La chaîne d'information

NOM :

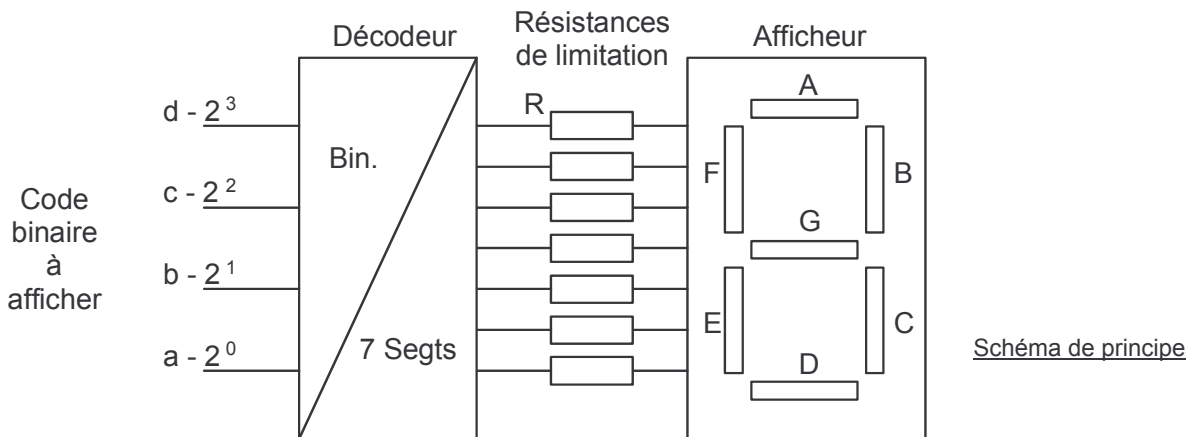
PRENOM :

AFFICHEUR NUMERIQUE

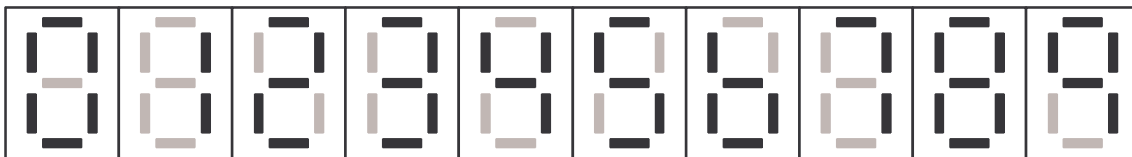


1. Principe

Dans de nombreuses applications, il est nécessaire d'afficher des données numériques. L'afficheur est associé à un décodeur spécifique au code à visualiser.



- ◆ L'afficheur numérique est constitué de 7 segments à LED, repérés A à F.
- ◆ 4 entrées a, b, c et d permettent le codage du chiffre à afficher en binaire.
- ◆ Le décodeur permet de commander les segments en fonction du chiffre à afficher.



Représentation des chiffres

Codage binaire du nombre à afficher

Le nombre à afficher est codé en binaire pur sur quatre bits d, c, b, a (entrées binaires 0 ou 1)

Chaque nombre est codé sous la forme

$$d \times 2^3 + c \times 2^2 + b \times 2^1 + a \times 2^0$$

$$d \times 8 + c \times 4 + b \times 2 + a \times 1$$

Exemple : le nombre décimal 5_{10} est codé en binaire

$$0 \times 8 + 1 \times 4 + 0 \times 2 + 1 \times 1$$

Soit 0101_2

2. Table de vérité

- ◆ Compléter la table de vérité

Chiffre

Code

Segments

	d (2^3)	c (2^2)	b (2^1)	a (2^0)
0	0	0	0	0
1	0	0	0	1
2				
3				
4				
5				
6				
7				
8				
9				

A	B	C	D	E	F	G
1	1	1	1	1	1	0
0	1	1	0	0	0	0

3. Equations des segments

$$\text{SEGMENT A} = d + \bar{a} \cdot \bar{c} + a \cdot (b + c)$$

$$\text{SEGMENT B} = c + \bar{a} \cdot b + a \cdot \bar{b}$$

$$\text{SEGMENT C} = c + \bar{b} + a$$

$$\text{SEGMENT D} = a \cdot (b + c) + \bar{b} \cdot \bar{c} + a \cdot \bar{b} \cdot \bar{c}$$

$$\text{SEGMENT E} = a \cdot (b + c)$$

$$\text{SEGMENT F} = d + \bar{a} \cdot (b + c) + \bar{b} \cdot \bar{c}$$

$$\text{SEGMENT G} = d + \bar{b} \cdot \bar{c} + c \cdot (a + \bar{b})$$

4. Codage du programme

Le programme des segments sera codé de la façon suivante :

Segments A, B et C : Equation logique

Segments D et E : Logigramme

Segments F et G : Langage à contacts

L'utilisation de symboles facilitera l'écriture des programmes.

Entrées (code)	Symbole	Adresse
a - 2^0	<code>_a_</code>	i0
b - 2^1	<code>_b_</code>	i1
c - 2^2	<code>_c_</code>	i2
d - 2^3	<code>_d_</code>	i3

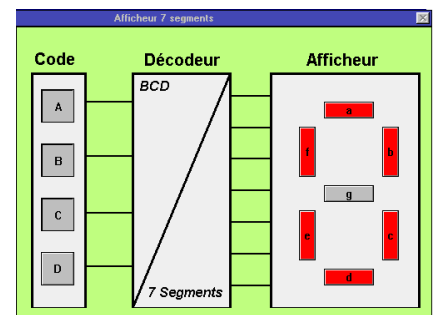
Sorties (segments)

A	<code>_SEGMENT A_</code>	O0
B	<code>_SEGMENT B_</code>	O1
C	<code>_SEGMENT C_</code>	O2
D	<code>_SEGMENT D_</code>	O3
E	<code>_SEGMENT E_</code>	O4
F	<code>_SEGMENT F_</code>	O5
G	<code>_SEGMENT G_</code>	O6

- ◆ Consulter au préalable le document ressources page 6/6
- ◆ Sélectionner votre répertoire courant.
- ◆ Coder le programme.
- ◆ Enregistrer le programme sous "aff.GR7"
- ◆ Compiler le programme en définissant le fichier des symboles
- ◆ Installer le programme et le tester en affichant les nombres 0 à 9.

Un objet (C:\POV\Afficheur\aff.aof vous permettra de simuler votre programme).

- ◆ **FAIRE CONTROLER VOTRE PROGRAMME.**
- ◆ Imprimer votre programme et le coller sur la page 5.



PROGRAMME à compléter

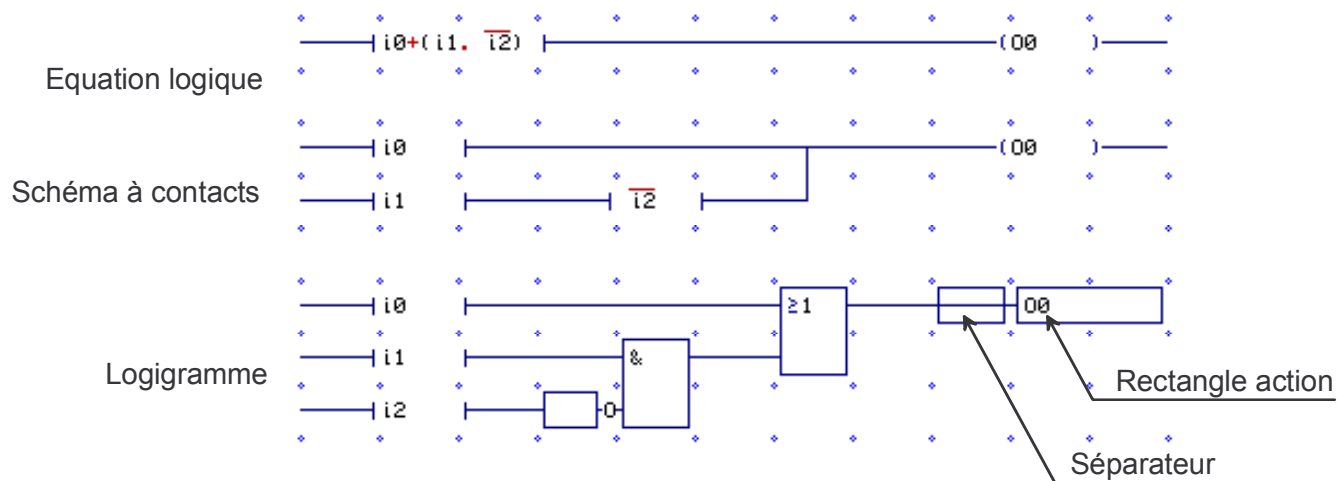
Segment a

—|_d_ + (_a_ . _c_) + (_a_ . (_b_ + _c_)) |—————(_SEGMENT A_)——

Programme

Documentation Automgen (Extrait)

A - LANGAGES



Remarques

Equation logiques : Niveaux de priorité

Par défaut les équations sont évaluées de la gauche vers la droite. Pour définir un autre ordre d'évaluation il faut utiliser des parenthèses. **L'opération 'ET LOGIQUE' n'est donc pas prioritaire.**

LOGIGRAMME

Un séparateur "touche 0" sera intercalé entre la zone de test et les actions.
Les rectangles d'action " touches W, X et Y" permettent de commander les sorties.

B – SYMBOLES

Pour faciliter la mise au point des programmes, il est possible d'associer les variables à des symboles.

Exemple :

i0	:	<u>_marche_</u>
i1	:	<u>_benne en haut_</u>
O0	:	<u>_PERCER_</u>

Le début et la fin des symboles sont définis par le caractère "_" (souligné, généralement associé à la touche [8] sur les claviers).

Le symbole passera automatiquement en vert. Il est conseillé d'éviter les accents dans les symboles.

Lors de la compilation du programme, il faudra définir la variable associée à chaque symbole.

Symbole _marche_ → Variable i0

Les symboles sont déclarés dans un fichier texte portant le nom du premier folio de l'application suivi de l'extension ".SYM".

On accède automatiquement à ce fichier avec la commande "Fichier des symboles" du menu "Boîte à outils".