

Dernière mise à jour	Codage de l'information et systèmes logiques	Denis DEFAUCHY
17/02/2016		Cours

A. Logique combinatoire

A.I. Introduction - Définitions

Les **systèmes logiques combinatoires** sont des systèmes pour lesquels des causes combinées impliquent immédiatement des effets. A tout moment, les mêmes causes impliquent toujours les mêmes effets. Causes et effets sont des **propositions logiques** associés à des **états** « vrai » ou « faux » traduits par deux variables 1 (vrai) ou 0 (faux). Il est ainsi possible d'utiliser **l'algèbre de Boole** afin de représenter les relations entre causes et effets par des **fonctions logiques**. L'ensemble des relations dites **causales** (à une cause correspond un effet) constitue un « **modèle de fonctionnement logique** » d'un système.

La réalisation pratique des fonctions logiques permettant d'induire des causes connaissant les effets s'effectue par des moyens divers, les plus utilisés étant l'électrique, le pneumatique et l'hydraulique.

Un système logique peut être combinatoire ou séquentiel. Les systèmes combinatoires sont des systèmes dans lesquels une combinaison des entrées induit une et une seule sortie. Prenons l'exemple d'un portail automatique : si une action à un instant t sur une commande d'ouverture ou de fermeture induit une ouverture complète ou une fermeture complète après avoir relâché le bouton, on voit que pour un même état des entrées, il existe 2 situations possibles : ouverture ou fermeture. Ceci est un système séquentiel dans lequel il faudra un automate pour gérer les séquences des actions. Si par contre il faut nécessairement laisser un bouton appuyé pour fermer et un bouton appuyé pour ouvrir, le système est combinatoire et des fonctions logiques suffiront à elles seules pour le faire fonctionner.

A.II. Représentation d'un modèle de fonctionnement logique

Il existe différents outils permettant de représenter un modèle de fonctionnement logique d'un système. Traitons un exemple pour introduire ces outils.

Soit une porte coulissante automatique gérée à l'aide

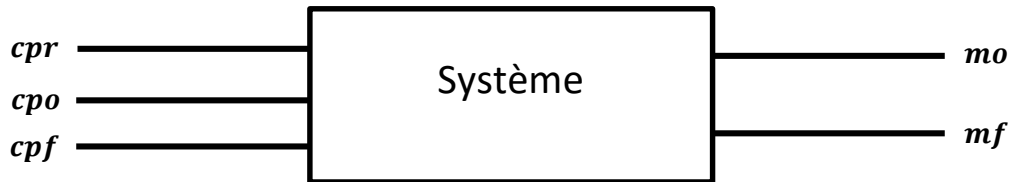
- d'un capteur de présence qui couvre en même temps les zones intérieure et extérieure (fonction de détection de présence d'un usager) et la zone de passage (fonction de sécurité) associé à la variable « *cpr* »
- d'un capteur « Porte ouverte » associé à la variable « *cpo* »
- d'un capteur « Porte fermée » associé à la variable « *cpf* »
- d'un moteur recevant deux ordres qui ne peuvent être envoyés simultanément
 - o ordre d'ouverture associé à la variable « *mo* »
 - o ordre de fermeture associé à la variable « *mf* »

Le modèle de fonctionnement logique de ce système doit permettre de représenter les ordres « *mo* » et « *mf* » à l'aide des diverses entrées issues des capteurs « *cpr* », « *cpo* » et « *cpf* ».

Dernière mise à jour	Codage de l'information et systèmes logiques	Denis DEFAUCHY
17/02/2016		Cours

A.II.1 Représentation globale du système

Le système logique proposé se schématise en fonction de ses entrées et sorties ainsi :



A.II.2 Table de vérité

Le premier outil à notre disposition est la table de vérité. On réalise un tableau répertoriant

- dans sa partie gauche les entrées en colonne et l'intégralité des états possibles de celles-ci
- dans sa partie droite les sorties et les états qu'elles doivent prendre en fonction des combinaisons d'entrées existantes

Pour créer ce tableau :

- on met autant de colonnes que de variables, ici 5
- on met 2^n lignes, n correspondant au nombre d'entrées, ici 3, soit 8 lignes

Entrées			Sorties	
<i>cpr</i>	<i>cpo</i>	<i>cpf</i>	<i>mo</i>	<i>mf</i>

On remplit ensuite les colonnes associées aux entrées afin de représenter l'intégralité des combinaisons possibles des entrées. On peut par exemple utiliser le code binaire :

Entrées			Sorties	
<i>cpr</i>	<i>cpo</i>	<i>cpf</i>	<i>mo</i>	<i>mf</i>
0	0	0		
0	0	1		
0	1	0		
0	1	1		
1	0	0		
1	0	1		
1	1	0		
1	1	1		

Dernière mise à jour	Codage de l'information et systèmes logiques	Denis DEFAUCHY
17/02/2016		Cours

Détaillons les différents cas rencontrés et voyons si la table de vérité peut être remplie directement.

<i>cpr</i>	<i>cpo</i>	<i>cpf</i>	Etat porte	Présence usager	Action
0	0	0	Ni ouverte - Ni fermée	Non	Fermer
0	0	1	Fermée		Ne rien faire
0	1	0	Ouverte		Fermer
0	1	1	Ouverte et fermée		<i>Etat impossible</i>
1	0	0	Ni ouverte - Ni fermée	Oui	Ouvrir
1	0	1	Fermée		Ouvrir
1	1	0	ouverte		Ne rien faire
1	1	1	Ouverte et fermée		<i>Etat impossible</i>

On voit apparaître deux états impossibles. Dans le cas **d'états impossibles**, ou **d'états indéterminés** (le cahier des charges ne précise pas les actions à mener dans des cas particuliers), ce sera au concepteur de choisir

- Pour des états pouvant mettre en danger la sécurité des usagers, le concepteur devra choisir la solution la plus appropriée. Ici, si l'un des capteurs « ouvert » ou « fermé » est défaillant et que les informations « ouvert » et « fermé » arrivent en même temps, il sera préférable d'imposer une ouverture (ou ne rien faire) afin de ne pas risquer de fermer la porte sur un usager, en particulier dans le cas de la dernière ligne du tableau.
- Pour des états indéterminés, le concepteur choisira la solution qui l'arrange. Il arrive souvent que ce choix permette de simplifier la fonction logique finale.

Nous pouvons donc choisir les actions à mener en fonction des entrées :

<i>cpr</i>	<i>cpo</i>	<i>cpf</i>	Action
0	0	0	Fermer
0	0	1	Ne rien faire
0	1	0	Fermer
0	1	1	<i>Ouvrir</i>
1	0	0	Ouvrir
1	0	1	Ouvrir
1	1	0	Ne rien faire
1	1	1	<i>Ouvrir</i>

Voici la table de vérité représentant le modèle de fonctionnement logique du système étudié.

Entrées			Sorties	
<i>cpr</i>	<i>cpo</i>	<i>cpf</i>	<i>mo</i>	<i>mf</i>
0	0	0	0	1
0	0	1	0	0
0	1	0	0	1
0	1	1	1	0
1	0	0	1	0
1	0	1	1	0
1	1	0	0	0
1	1	1	1	0

**Table de vérité
du système étudié**

A.II.3 Fonctions logique

Les fonctions logiques permettent de représenter les variables de sortie « *mo* » et « *mf* » en fonction des entrées à l'aide des outils de l'algèbre de Boole.

Entrées			Sorties	
<i>cpr</i>	<i>cpo</i>	<i>cpf</i>	<i>mo</i>	<i>mf</i>
0	0	0	0	1
0	0	1	0	0
0	1	0	0	1
0	1	1	1	0
1	0	0	1	0
1	0	1	1	0
1	1	0	0	0
1	1	1	1	0

En utilisant la table de vérité, on montre les relations suivantes :

$$\begin{cases} mo = \overline{cpr}.cpo.cpf + cpr.\overline{cpo}.\overline{cpf} + cpr.\overline{cpo}.cpf + cpr.cpo.cpf \\ mf = \overline{cpr}.\overline{cpo}.\overline{cpf} + \overline{cpr}.cpo.\overline{cpf} \end{cases}$$

Cette forme non simplifier s'appelle la **première forme canonique** des expressions de *mo* et *mf*.

Il suffit alors d'utiliser les outils de l'algèbre de Boole afin de simplifier ces expressions :

$$\begin{aligned} mo &= \overline{cpr}.cpo.cpf + cpr.\overline{cpo}.\overline{cpf} + cpr.\overline{cpo}.cpf + cpr.cpo.cpf \\ &= cpo.cpf.(cpr + \overline{cpr}) + cpr.\overline{cpo}.(cpr + \overline{cpr}) \\ &= cpo.cpf + cpr.\overline{cpo} \end{aligned}$$

$$mf = \overline{cpr}.\overline{cpo}.\overline{cpf} + \overline{cpr}.cpo.\overline{cpf} = \overline{cpr}.\overline{cpf}(\overline{cpo} + cpo) = \overline{cpr}.\overline{cpf}$$

Soit finalement :

$\begin{cases} mo = cpo.cpf + cpr.\overline{cpo} \\ mf = \overline{cpr}.\overline{cpf} \end{cases}$	<p>Fonctions logiques du système étudié</p>
---	--

Remarques :

- cette étape est très importante car plus la simplification sera bien faite, plus le logigramme et le schéma électrique du câblage du système seront concis et simples.
- La simplification de ces expressions s'effectue simplement à l'aide des tableaux de Karnaugh qui ne sont plus au programme

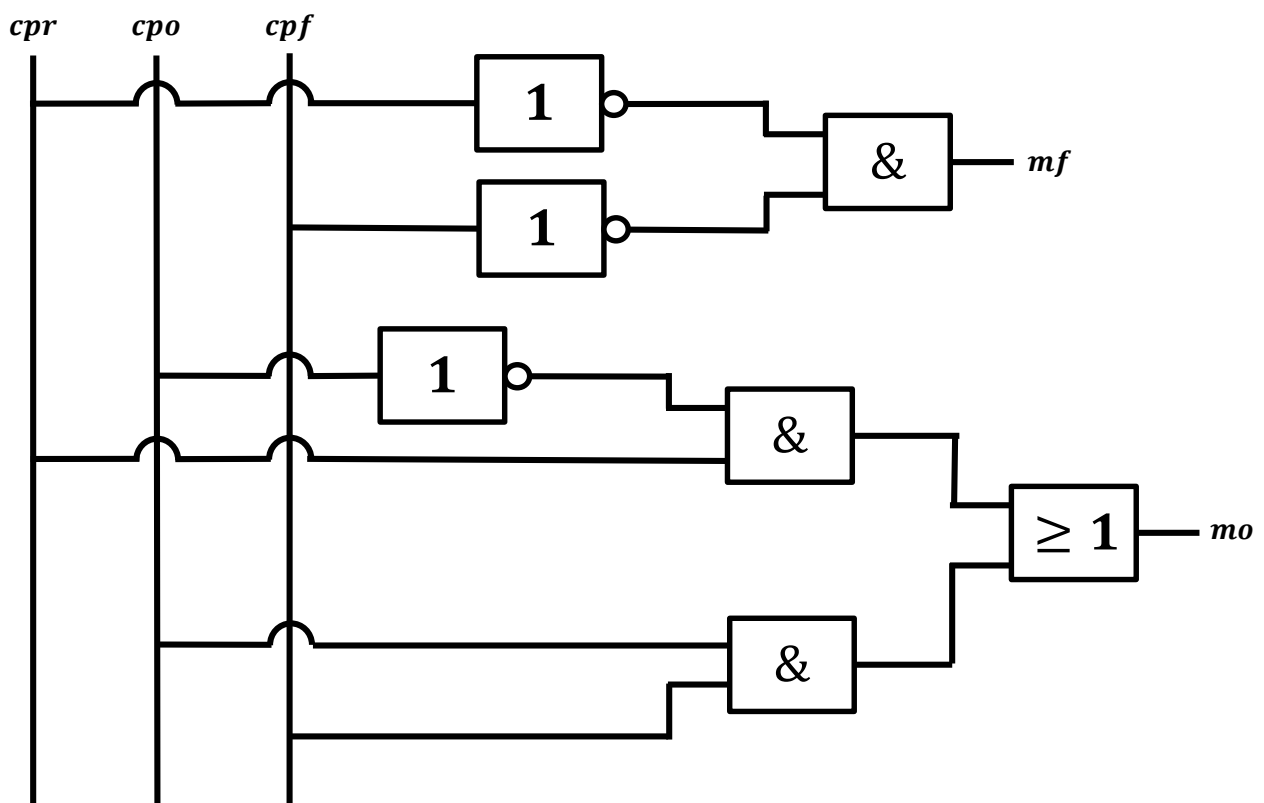
A.II.4 Logigramme

Un logigramme est une représentation quasi fonctionnelle du câblage du système.

On place dans sa partie gauche autant de traits verticaux qu'il y a d'entrées. Ce sont comme des fils électriques comportant la valeur 1 associée à la variable concernée.

Il suffit alors d'utiliser les opérateurs logiques à notre disposition dans le but de d'obtenir en sortie la valeur des variables de sortie en fonction des variables d'entrée en utilisant les fonctions logiques du système :

$$\begin{cases} mo = cpo \cdot cpf + cpr \cdot \overline{cpo} \\ mf = \overline{cpr} \cdot \overline{cpf} \end{cases}$$



**Logigramme
du système étudié**

Dernière mise à jour 17/02/2016	Codage de l'information et systèmes logiques	Denis DEFAUCHY Cours
------------------------------------	---	-------------------------

A.II.5 Schéma à contacts

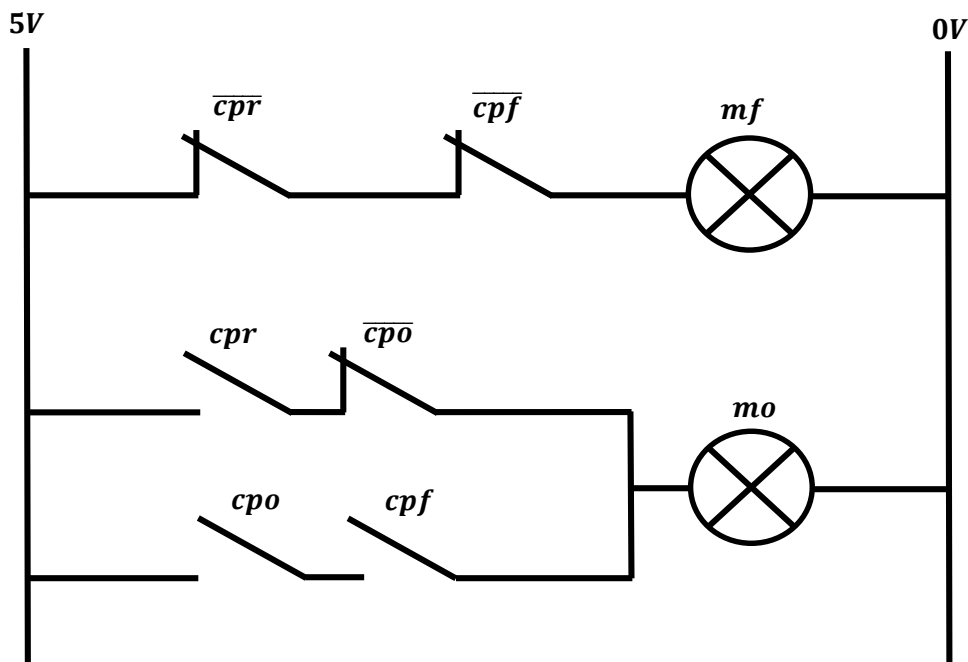
On peut finalement représenter le modèle de fonctionnement logique du système étudié à l'aide du schéma à contacts, première étape vers la réalisation électrique du système.

Les quelques caractéristiques de réalisation des schémas à contacts à connaître sont les suivantes :

- La fonction ET est représentée par une mise en série des contacts
- La fonction OU est représentée par une mise en parallèle des contacts
- Une variable « normale » (a) est représentée par un contacteur normalement ouvert
- Une variable « opposée » (\bar{a}) est représentée par un contacteur normalement fermé
- La sortie est représentée par un cercle avec une croix à l'intérieur

On obtient ainsi pour notre système :

$$\begin{cases} mo = cpo \cdot cpf + cpr \cdot \overline{cpo} \\ mf = \overline{cpr} \cdot \overline{cpf} \end{cases}$$



**Schéma à contacts
du système étudié**

Remarque :

- Un schéma à contact représente le système lorsque toutes les variables sont à l'état 0
- Un changement d'une variable implique la fermeture du contact s'il est normalement ouvert et son ouverture s'il est normalement fermé