Électronique et logique

par Michel HENRY,
Université Pierre-et-Marie-Curie, Paris.

Résumé.

Nous introduisons dans cet article les premières notions de logique binaire et indiquons les dispositifs techniques permettant de réaliser les opérations correspondantes, puis nous décrivons un certain nombre d'applications.

INTRODUCTION.

Les programmes de collège qui sont entrés en vigueur à la dernière rentrée font une large place à l'électronique en commençant par ce qu'il est convenu d'appeler les circuits de logique. Ces circuits constituent la base de l'électronique digitale, celle des micro-ordinateurs, par opposition à l'électronique analogique, celle des chaînes Hi-Fi.

Si le programme de 6e restreint l'étude de la logique au modèle simple des interrupteurs électromécaniques, celui de 5e, par contre, fait explicitement appel aux circuits intégrés et au transistor en mode saturé-bloqué.

Il doit être bien entendu que cet article n'est pas un cours « clés en main », mais une information à l'usage des collègues. Les programmes, aussi bien que les commentaires, sont on ne peut plus explicites sur ce point : l'introduction de l'électronique au collège doit être expérimentale, et les divers composants considérés comme des « boîtes noires ».

Nous supposons connues du lecteur les notions fondamentales d'électronique classique, en particulier le fonctionnement d'une diode, électroluminescente ou non, d'un transistor bipolaire ou à effet de champ.

Après avoir rappelé les notions fondamentales de logique binaire et d'algèbre de BOOLE, nous décrirons un certain nombre de réalisations techniques et présenterons quelques applications immédiates.

ELEMENTS DE LOGIQUE BINAIRE.

Propositions logiques.

La logique — si j'ose dire — de l'exposé aurait peut-être voulu que nous attaquions directement par l'algèbre de BOOLE. Toutefois, nous avons préféré introduire la logique binaire par le jeu des propositions logiques. Celles-ci, par la multitude de jeux auxquelles elles se prêtent peuvent, sans doute, aider de jeunes enfants à se familiariser avec cette tournure d'esprit un peu particulière où tout est toujours vrai ou faux.

Il s'agit de propositions auxquelles nous pouvons attribuer sans ambiguité la qualification « vrai » ou « faux ». Nous pouvons aussi les considérer comme des questions qui n'appellent que l'une des deux réponses « oui » ou « non ».

Exemple :

— Le corbeau est un oiseau.

— Le cheval est-il un insecte ?

Contre-exemple :

— Quel âge avez-vous ?

Nous appelons « propositions élémentaires » de telles affirmations ou questions.

Il est, bien entendu, possible d'imaginer des propositions plus compliquées, également considérées comme logiques si elles obéissent au critère précédent.

Exemple :

— J'irai en cours s'il fait beau et si je me réveille à temps.

Les propositions complexes de ce type peuvent être divisées en propositions élémentaires reliées par les conjonctions habituelles « et », « ou », « non ».

Dans l'exemple précédent, il existe trois propositions élémentaires :

— J'irai en cours.

— Il fait beau.

— Je me réveille à temps.

La première est vraie si les deux autres le sont en même temps.

Exemple :

— Je mangerai du dessert si j'ai encore faim ou si ce n'est pas de la tarte aux pommes.

Les trois propositions élémentaires sont :

— Je mangerai du dessert.

— J'ai encore faim.

— Le dessert est de la tarte aux pommes.

La première est vraie si la seconde est vraie ou si la troisième n'est pas vraie.

Donnons enfin un exemple des problèmes de logique tel qu'il est possible d'en trouver dans les livres ou les revues spécialisés.

Un générateur est surveillé par deux ordinateurs qui peuvent être ou non déréglés. On admet qu'un ordinateur déréglé fournit au moins une réponse fausse. Les indications données par les deux ordinateurs sont les suivantes :

O_1 : Le générateur est hors service,
L'ordinateur O_2 est déréglé.

O_2 : Le générateur est hors service,
L'ordinateur O_1 est en bon état.

La question est de savoir s'il est possible de tirer une conclusion logique de ces indications, concernant le fonctionnement du générateur.

Nous devons choisir une hypothèse de départ, à savoir le bon fonctionnement ou non des ordinateurs. Par exemple, nous admettons que O_1 est en état de marche et donc qu'il fournit deux indications « vraies ». Il est alors facile de voir que les indications fournies par O_2 ne peuvent en aucun cas être compatibles avec celles fournies par O_1. Puisqu'il y a contradiction logique, c'est que notre hypothèse de départ est fausse. Nous prenons l'hypothèse alternative et aboutissons à la conclusion que le générateur est en parfait état mais que les deux ordinateurs sont hors service...

Algèbre de Boole.

Pour simplifier l'écriture de telles propositions, les logiciens ont imaginé un système de notation « algébrique ». Il se trouve que ce système correspond point par point avec une algèbre mathématique dite *algèbre de Boole,* du nom de l'un de ses principaux fondateurs. Le lecteur a certainement déjà fait connaissance avec l'algèbre de BOOLE, ne serait-ce qu'en théorie des ensembles, aussi nous limitons-nous aux notions utilisées en logique électronique.

Comme toute algèbre, celle de BOOLE combine des « variables » à l'aide d' « opérateurs » pour obtenir des « fonctions ».

En logique formelle, les variables correspondent aux propositions élémentaires, les fonctions aux propositions complexes. En électronique, nous définissons plutôt les variables comme des actions ou des états imposés à un système et les fonctions comme le résultat de ces actions. Bien entendu, le système ne doit pouvoir se trouver que dans l'un ou l'autre de deux états différents et mutuellement exclusifs, aussi bien « en entrée » (variables) qu' « en sortie » (fonctions).

Les seuls résultats admissibles étant « vrai » ou « faux » ou bien « oui » ou « non », nous *convenons* de représenter :

« vrai » ou bien « oui » par 1

« faux » ou bien « non » par 0.

Il est possible de montrer, et nous admettrons, que trois opérateurs et trois seulement, outre l'implication qui est toujours sous-entendue, suffisent pour résoudre tous les problèmes de logique binaire, électronique ou non.

Ces trois opérateurs sont :

a) La négation ou opérateur NON.

Il s'agit de la négation au sens usuel du terme : si une proposition est vraie, la négation de cette proposition est fausse. De façon analogue, si un état est réalisé, l'état défini par la négation du précédent ne l'est pas. La négation est notée en surlignant le symbole de la proposition — ou de l'état — en question.

Exemples :

A = je suis au cinéma,

$\overline{A}$ = je ne suis pas au cinéma.

B = l'ampoule est allumée,

$\overline{B}$ = l'ampoule est éteinte.

b) L'addition ou opérateur OU.

Cet opérateur combine deux variables ou plus et donne une fonction, laquelle vaut 1 (*i.e.* est vraie) si l'une quelque des variables ou toutes valent 1. L'addition est notée avec le symbole de... l'addition (numérique).

Exemple :

A = il fait froid,

B = il pleut,

C = je mets mon manteau.

L'équation logique $C = A + B$ est équivalente à : « Je mets mon manteau s'il fait froid ou s'il pleut. »

c) Le produit ou opérateur ET.

Comme le précédent, cet opérateur combine plusieurs variables pour former une fonction. Celle-ci ne prend la valeur 1 que si toutes les variables ont cette valeur. Le produit est noté avec le symbole de la multiplication numérique.

Exemple :

A = le gardien est présent,

B = j'ai un rendez-vous,

C = je puis entrer.

L'équation logique $C = A \cdot B$ est équivalente à : « Je puis entrer si le gardien est présent et si j'ai un rendez-vous. »

Tables de vérité.

Il est souvent utile, ou simplement pratique, de déterminer les valeurs d'une fonction logique en fonction de celles des variables dont elle dépend. Un procédé commode consiste à dresser un tableau, dit *table de vérité.*

Dans les colonnes du tableau, nous portons les valeurs des variables, puis celles des fonctions, chaque ligne représentant un cas particulier. Chaque variable logique ne prenant que l'une des deux valeurs 0 ou 1, la table de vérité complète d'une fonction de *n* variables comporte 2^n lignes. Il apparaît immédiatement qu'elle ne garde des dimensions raisonnables que si *n* reste faible, mettons de l'ordre de 4 ou 5 pour fixer les idées. Au-delà, il est préférable de dresser des tables partielles, par regroupemnt des variables.

Les tables de vérité des trois opérateurs de base sont les suivantes :

Opérateur NON :

A	$B = \overline{A}$
0	1
1	0

Opérateur OU :

A	B	A + B
0	0	0
0	1	1
1	0	1
1	1	1

Opérateur ET :

A	B	A • B
0	0	0
0	1	0
1	0	0
1	1	1

Une fonction logique peut être définie soit par son « expression mathématique » à partir des variables et des opérateurs, soit par sa table de vérité.

Ce dernier cas est le plus fréquent dans la pratique et le problème est alors de trouver la combinaison d'opérateurs conduisant au résultat cherché.

Voyons quelques exemples de ces deux procédés.

a) Expression mathématique.

Deux fonctions très utilisées en électronique, en raison de la facilité avec laquelle il est possible de les réaliser sous forme de circuits intégrés sont la fonction « ET-NON » (NAND en anglais) et la fonction « OU-NON » (NOR en anglais), dont les expressions mathématiques sont respectivement :

$$C = \overline{A \cdot B} \text{ (ET-NON)}$$

$$C = \overline{A + B} \text{ (OU-NON).}$$

Nous en déduisons immédiatement leurs tables de vérité :

A	B	A • B	$\overline{A \cdot B}$
0	0	0	1
0	1	0	1
1	0	0	1
1	1	1	0

Nous pouvons interpréter cette fonction par l'expression « pas les deux simultanément ».

A	B	A + B	$\overline{A + B}$
0	0	0	1
0	1	1	0
1	0	1	0
1	1	1	0

De la même façon, nous pourrions dire que cette fonction correspond à « ni l'un ni l'autre ».

b) Table de vérité.

Nous prenons l'exemple de la fonction OU EXCLUSIF (EXOR en anglais) dont la valeur est 1 si celles des variables sont différentes, et 0 si les deux variables ont même valeur. Il résulte de la définition précédente que sa table de vérité est :

A	B	A * B
0	0	0
0	1	1
1	0	1
1	1	0

Il n'existe pas de règle permettant de déterminer à coup sûr l'expression mathématique associée à une table de vérité donnée à l'avance. En fait, il faut procéder de proche en proche par « intuition raisonnée », et il existe en général plusieurs solutions acceptables. Le lecteur vérifiera directement, à titre d'exercice, qu'une telle solution pour la fonction OU exclusif est :

$$A * B = (A + B) \cdot \overline{A \cdot B}.$$

Propriétés des opérateurs.

Les opérateurs logiques ET, OU, NON possèdent un certain nombre de propriétés classiques : associativité, commutativité... que nous nous bornons à rappeler ici. Le lecteur pourra les vérifier directement, par exemple à l'aide des tables de vérité.

a) Associativité.

$$A + (B + C) = (A + B) + C$$

$$A \cdot (B \cdot C) = (A \cdot B) \cdot C.$$

b) Commutativité.

$$A + B = B + A$$

$$A \cdot C = C \cdot A.$$

c) Distributivité.

$$A \cdot (B + C) = A \cdot B + A \cdot C$$

$$A + B \cdot C = (A + B) \cdot (A + C).$$

d) *Relations de de Morgan.*

$$\overline{A+B} = \overline{A} \cdot \overline{B}$$

$$\overline{A \cdot B} = \overline{A} + \overline{B}.$$

Représentation graphique.

Les schémas électroniques ou électromécaniques utilisent une représentation graphique des opérateurs précédents. Il existe dans la littérature, et en fonction de l'origine et de l'âge de celle-ci, deux conventions principales : la représentation américaine (USA) et la représentation européenne (CIE). L'emploi de la seconde est recommandé par l'AFNOR et adopté dans la quasi-totalité des livres récemment édités en France, mais il est encore fréquent de trouver la première, aussi indiquons-nous les deux (fig. 1).

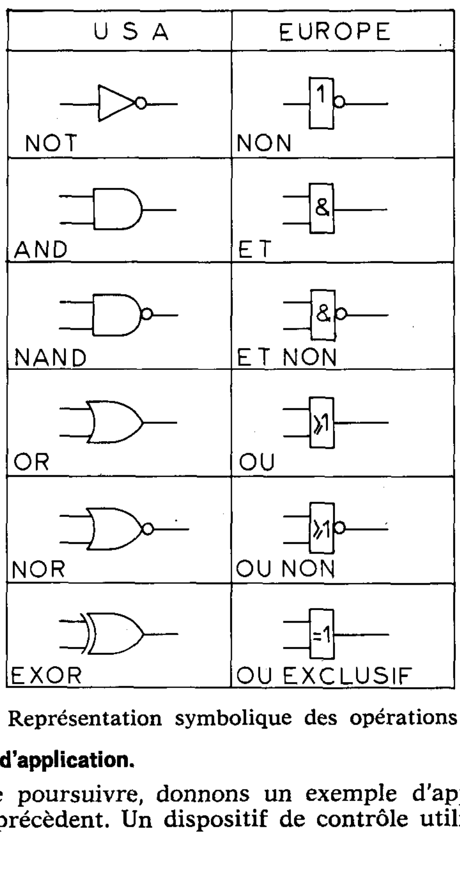

Fig. 1. — Représentation symbolique des opérations logiques.

Exemple d'application.

Avant de poursuivre, donnons un exemple d'application des notions qui précèdent. Un dispositif de contrôle utilise trois capteurs C_1, C_2, C_3 fonctionnant en tout ou rien. Il peut s'agir de détecteurs de lumière, de niveau, de température... Ceci nous donne trois variables logiques, donc huit possibilités, et nous pouvons par exemple mettre en fonctionnement autant d'indicateurs lumineux.

Avec un choix convenable des valeurs logiques 0 et 1 associées à chaque capteur et à chaque indicateur, la table de vérité du dispositif de contrôle est la suivante :

C_1	C_2	C_3	I_1	I_2	I_3	I_4	I_5	I_6	I_7	I_8
0	0	0	1	0	0	0	0	0	0	0
0	0	1	0	1	0	0	0	0	0	0
0	1	0	0	0	1	0	0	0	0	0
0	1	1	0	0	0	1	0	0	0	0
1	0	0	0	0	0	0	1	0	0	0
1	0	1	0	0	0	0	0	1	0	0
1	1	0	0	0	0	0	0	0	1	0
1	1	1	0	0	0	0	0	0	0	1

Nous devons former les huit équations liant les huit fonctions logiques I_1... I_8 aux trois variables logiques C_1, C_2 et C_3. Ainsi, I_1 doit prendre la valeur 1 quand C_1, C_2 et C_3 sont simultanément nulles. Nous pouvons donc poser :

$$I_1 = \overline{C}_1 \cdot \overline{C}_2 \cdot \overline{C}_3$$

et de la même façon :

$$I_2 = \overline{C}_1 \cdot \overline{C}_2 \cdot C_3 \qquad I_3 = \overline{C}_1 \cdot C_2 \cdot \overline{C}_3 \qquad I_4 = \overline{C}_1 \cdot C_2 \cdot C_3$$

$$I_5 = C_1 \cdot \overline{C}_2 \cdot \overline{C}_3 \qquad I_6 = C_1 \cdot \overline{C}_2 \cdot C_3 \qquad I_7 = C_1 \cdot C_2 \cdot \overline{C}_3$$

$$I_8 = C_1 \cdot C_2 \cdot C_3$$

d'où un schéma possible de réalisation (fig. 2).

Remarque.

Il est tentant de limiter le nombre d'indicateurs à trois, un par capteur. Toutefois une telle solution se prête mal à une surveillance automatique, qui exige de recevoir une information différente pour chacune des situations envisageables.

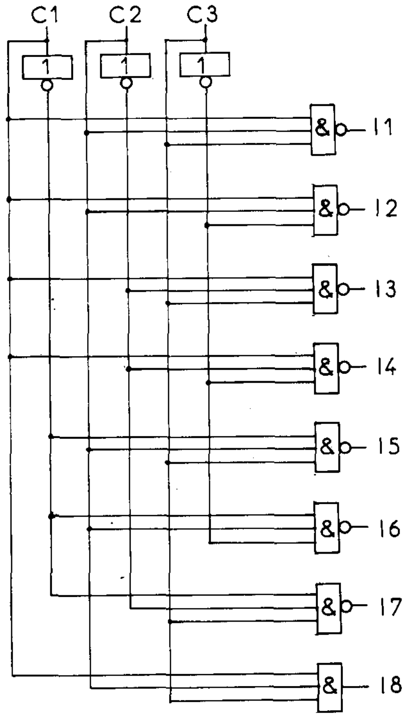

Fig. 2. — Schéma de principe d'un circuit de contrôle.

REALISATION TECHNIQUE.

Les opérations logiques sont, somme toute, assez simples. Le problème à présent est de réaliser des dispositifs matériels capables de les effectuer avec le minimum d'intervention humaine, ce qui permet — c'est en fait le but cherché — de réduire le temps de traitement et d'augmenter sa fiabilité. Ainsi, il est plus rapide et plus sûr d'effectuer une recherche bibliographique à partir d'un fichier plutôt qu'en passant en revue les piles de livres de la bibliothèque...

Cartes perforées.

Il s'agit sans doute du système le plus ancien, déjà utilisé par JACQUARD au XVIII^e^ siècle pour les métiers à tisser et aussi dans divers instruments de musique tel l'orgue de Barbarie. Jusqu'à ces dernières années, ces cartes constituaient l'un des principaux supports de données des ordinateurs, mais elles ont été vaincues par les progrès de l'enregistrement magnétique.

De nos jours, elles ne subsistent plus guère que pour des fichiers et aussi pour... le tiercé.

Ces fiches, qu'il est facile de réaliser à l'aide de bristol et d'un perforateur de bureau, portent une série de trous à leur périphérie, chaque trou étant affecté à une variable logique selon un code qui dépend de l'application envisagée.

Fixons les idées sur un exemple simple : nous désirons créer un fichier d'élèves par âge et par sexe. Nous utilisons une carte comportant 14 trous (fig. 3 *a*).

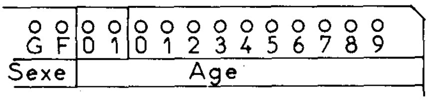

Fig. 3 *a*. — Carte perforée pour opérations logiques.

— Deux pour le sexe (garçon ou fille).

— Deux pour le chiffre des dizaines de l'âge : 0 ou 1, en admettant qu'aucun élève n'ait plus de 19 ans.

— Dix pour le chiffre des unités de l'âge.

La valeur de chacune de ces variables logiques est codée de la façon suivante : si elle vaut 0, la perforation est laissée intacte ; si elle vaut 1, la perforation est encochée (fig. 3 *b*). Une

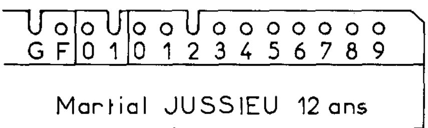

Fig. 3 *b*. — Exemple d'utilisation d'une carte perforée.

fiche est affectée à chaque élève. Ceci fait, les fiches sont entassées sans ordre particulier, mais en veillant à ce que les trous affectés à une même variable logique soient alignés.

Remarque.

Le codage des cartes perforées d'ordinateur consiste à trouer le carton lorsque la valeur de la variable logique est 1 et à le laisser intact lorsqu'elle vaut 0. Toutefois, ceci ne change rien au principe des opérations logiques décrites par la suite.

Pour effectuer un tri, il suffit de glisser dans le trou *ad hoc* une mince tige métallique (aiguille à tricoter) et de soulever le paquet de fiches. Toutes celles pour lesquelles la perforation est encochée tombent, les autres restent suspendues. Ainsi, ayant enfilé la tige dans le trou « G », les fiches des garçons tombent et celles des filles restent accrochées.

Il est aisé de réaliser les diverses opérations logiques : nous venons de voir l'opération « NON », puisque de notre point de vue les filles sont des « non-garçons ». De façon analogue, nous pouvons obtenir les opérations ET et OU. Supposons que nous cherchions les enfants âgés de 12 ans : nous trions le paquet des fiches pour lesquelles le chiffre des dizaines de l'âge vaut 1, puis une nouvelle sélection *dans ce paquet* nous donnera les fiches telles que le chiffre des unités de l'âge soit égal à 2. Si nous voulions trouver les élèves filles ou âgés de dix ans et plus, nous commencerions par sélectionner le paquet « filles » puis, *dans le reste,* les fiches pour lesquelles le chiffre des dizaines de l'âge est égal à 1.

Une caractéristique de ce procédé est que le codage est permanent, ce qui est un avantage ou un inconvénient, selon les cas. Au crédit, portons la simplicité et le faible coût, au débit, la lenteur et l'encombrement. Il permet toutefois de s'initier dans de bonnes conditions aux notions fondamentales de la logique binaire.

Interrupteurs électromécaniques.

Nous regroupons dans cette catégorie aussi bien les interrupteurs « *sensu stricto* » que les relais électromécaniques. Les interrupteurs peuvent être fermés et laisser passer le courant, ou bien ouverts et l'interrompre. La convention habituellement adoptée affecte la valeur logique 1 au premier état et 0 au second (fig. 4 *a*).

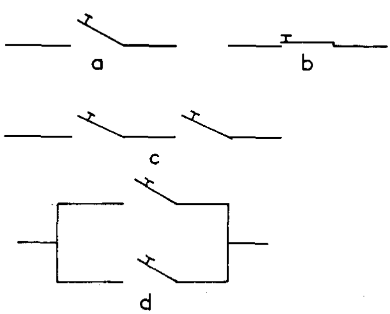

Fig. 4. — Etats logiques d'un interrupteur (*a* et *b*). Opérations ET (*c*) et OU (*d*) réalisées avec deux interrupteurs.

L'opération ET résulte de la mise en série de deux interrupteurs tandis que la fonction OU est obtenue par leur mise en parallèle (fig. 4 *b*). Il n'est pas possible de réaliser la fonction NON, ce qui est un inconvénient majeur.

Remarque.

Certains auteurs proposent la convention suivante : la variable logique associée à un interrupteur n'est plus l'état de celui-ci mais l'action exercée sur lui. Il est alors possible de définir une opération NON à l'aide d'interrupteurs « pousse-coupe » normalement fermés et qui interrompent le courant lorsqu'on agit sur eux. D'autres proposent de placer des interrupteurs en dérivation. Nous ne les suivrons pas dans cette voie, qui nous paraît compliquer inutilement les choses.

De même, le circuit « va et vient », dont l'étude est au programme de 6e, ne peut pas être décrit par des fonctions logiques simples car les deux états de chaque interrupteur sont équivalents.

Les interrupteurs électromécaniques sont simples, fiables, mais lents et bruyants. Leurs applications sont, pour l'essentiel, l'éclairage domestique, et divers dispositifs industriels de commande : machines-outils, et aussi lave-linge ou lave-vaisselle. Une autre application, en voie de disparition, est constituée par les centraux téléphoniques du type « crossbar » ou équivalent.

Dispositifs électroniques.

Un certain nombre de composants électroniques peuvent présenter deux états de fonctionnement distincts et mutuellement exclusifs. Citons — pour mémoire — les tubes à vide (triode, pentode), les diodes et surtout les transistors. Nous pouvons envisager leur utilisation en logique binaire, en affectant à chaque état une des deux valeurs d'une véritable logique.

Plus précisément, les variables et fonctions logiques sont ici des tensions, mesurées par rapport à une référence commune. Il est habituel d'affecter la valeur logique 0 à cette référence et la valeur logique 1 à une tension V_0 positive (logique positive) ou négative (logique négative). Nous nous limitons ici à la logique positive, qui est en fait la plus utilisée.

Les composants discrets, tubes, diodes et transistors ont été effectivement utilisés. Rappelons que le premier calculateur électronique, ENIAC construit en 1946, fonctionnait grâce à des tubes.

La fragilité mécanique de ceux-ci et le besoin d'un courant de chauffage les ont rapidement éliminés au profit des transistors. Ces derniers sont depuis une vingtaine d'années remplacés par les circuits intégrés, moins volumineux, plus économes et plus fiables.

Pour cette raison, nous nous bornons à signaler quelques exemples de montages à diodes et transistors. Leur seul intérêt à l'heure actuelle est probablement pédagogique, car ils permettent de décomposer une opération complexe en éléments plus simples, ce que n'autorise évidemment pas un circuit intégré.

Les entrées A, B... peuvent être portées au potentiel V_0 (valeur logique 1) ou au potentiel 0 (valeur logique 0). Dans le premier cas, les diodes sont passantes et les transistors saturés, tandis que dans le second cas, les diodes et transistors sont bloqués. Ceci fixe le potentiel de la sortie S, soit à V_0 (valeur logique 1) soit à 0 (valeur logique 0) et permet de réaliser diverses opérations élémentaires, comme l'indiquent les schémas ci-après (fig. 5). Par combinaison de ces opérations élémentaires, nous pouvons obtenir des opérations plus complexes. Le lecteur imaginera sans peine la rapidité avec laquelle croît l'intrication du câblage...

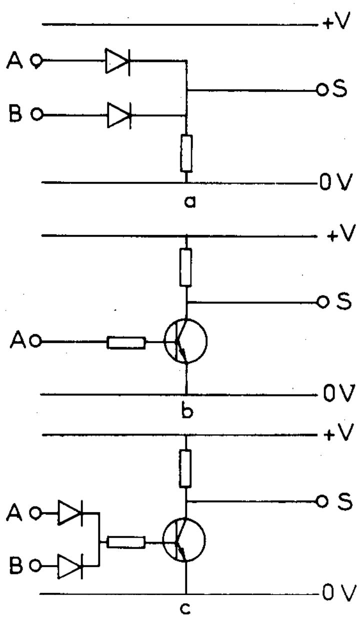

Fig. 5. — Réalisation de diverses opérations logiques à l'aide de diodes et de transistors : OU (*a*), NON (*b*), OU-NON (*c*).

Dans les circuits intégrés, les opérateurs logiques sont réalisés par des « transistors » regroupés sur une même plaquette de silicium (« puce »). Nous mettons le mot transistors entre guillemets, car les divers éléments sont de moins en moins identifiables au fur et à mesure que croît l'intégration.

Ces transistors sont bipolaires en logique dite TTL, ou à effet de champ en logique dite C. MOS. A titre d'exemple, le schéma suivant représente l'équivalent d'un opérateur ET-NON intégré, réalisé à partir de transistors bipolaires et à effet de champ (fig. 6).

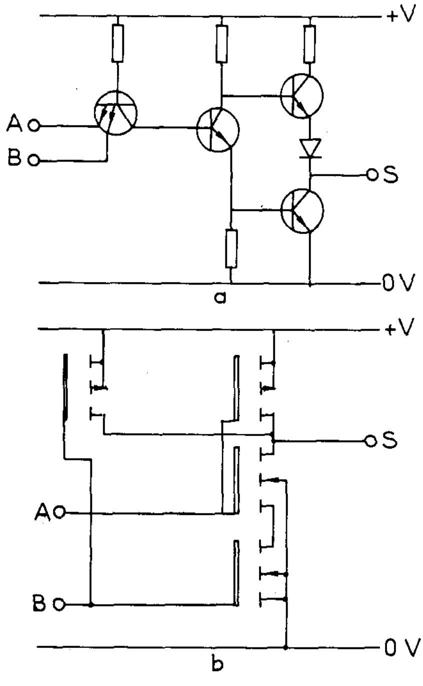

Fig. 6. — Schéma de principe d'une opération ET-NON réalisée à l'aide de transistors bipolaires (*a*) et de transistors à effet de champ (*b*).

En ce qui concerne l'emploi des circuits intégrés, il n'y a que peu de choses à dire, sinon de respecter les indications du fabricant... Signalons toutefois trois points qui nous paraissent importants.

— Il est possible, comme pour les composants discrets, de réaliser des opérations complexes à partir d'opérations simples du type ET, ET-NON... Il est plus simple, plus économique, plus fiable de chercher dans le catalogue le circuit *ad hoc,* qui existe à peu près certainement.

— Tous les fabricants se sont entendus pour donner la même référence et le même brochage aux circuits assurant la même fonction. Ainsi, la référence 7400 indique un circuit intégré en logique TTL comportant quatre opérateurs ET-NON indépendants ; la référence 4049 correspond à un circuit intégré en logique C. MOS comportant six opérateurs NON indépendants.

— La tension d'alimentation des circuits TTL doit être égale à 5 V, la tension V_0 étant comprise entre 2 V et 5 V. Ces conditions sont moins restrictives pour la logique C. MOS, les tensions étant comprises entre 3 V et 15 V.

APPLICATIONS.

Il existe une grande variété d'applications de la logique binaire, traduite par la multitude de circuits existants ou en projet.

L'usage veut que les applications soient regroupées en deux catégories, la *logique combinatoire* et la *logique séquentielle.* Il ne faut toutefois pas oublier que tous les codages passent par le système de numérotation binaire, soit pur, soit plus ou moins modifié.

Logique combinatoire.

Ici, l'état logique des sorties est déterminé de façon unique par celui des entrées. Nous nous limitons à trois exemples : l'addition de deux nombres, leur comparaison, et le transcodage d'un système numérique dans un autre.

Remarque.

Nous parlons de nombres pour fixer les idées, mais il doit être bien entendu que le système de numérotation binaire permet de représenter n'importe quelle autre information. C'est d'ailleurs là son intérêt et l'explication de son extraordinaire développement.

Addition.

Rappelons la table d'addition en système binaire :

n_1	n_2	$n_1 + n_2$
0	0	0
0	1	1
1	0	1
1	1	10

Cette table représente la table de vérité d'une opération logique particulière, que nous continuons à nommer addition, mais qu'il ne faut pas confondre bien entendu avec l'addition logique (opération « OU ») définie précédemment.

A partir de cette table, nous pourrions déterminer les équations logiques des divers états de la sortie $n_1 + n_2$ en fonction de ceux des entrées n_1 et n_2, puis en déduire un schéma de réalisation pratique. Nous allons le faire dans le cas plus général où il existe une retenue à combiner avec n_1 ou n_2. La table d'addition — qui est aussi la table de vérité — prend alors la forme suivante :

r	n_1	n_2	$n_1 + n_2$		R	S
			R	S		
0	0	0	0	0		
0	0	1	0	1		$\bar{r} \cdot \bar{n}_1 \cdot n_2$
0	1	0	0	1		$\bar{r} \cdot n_1 \cdot \bar{n}_2$
0	1	1	1	0	$\bar{r} \cdot n_1 \cdot n_2$	
1	0	0	0	1		$r \cdot \bar{n}_1 \cdot \bar{n}_2$
1	0	1	1	0	$r \cdot \bar{n}_1 \cdot n_2$	
1	1	0	1	0	$r \cdot n_1 \cdot \bar{n}_2$	
1	1	1	1	1	$r \cdot n_1 \cdot n_2$	$r \cdot n_1 \cdot n_2$

Dans la table d'addition, r est la retenue, n_1 et n_2 les deux chiffres à additionner, $n_1 + n_2$ le total. Côté table de vérité, r, n_1 et n_2 sont les entrées et R et S les sorties. Les deux dernières colonnes du tableau donnent les premières équations logiques permettant de calculer R et S à partir de r, n_1 et n_2. Avant d'expliciter ce point, insistons sur le fait qu'il ne s'agit que d'un exemple. Le lecteur trouvera sans peine d'autres équations, tout aussi valables que celles-ci.

Prenons par exemple la première équation logique (seconde ligne du tableau) $S = \bar{r} \cdot \bar{n}_1 \cdot n_2$. La table de vérité nous montre que S prend la valeur logique 1 quand r prend la valeur logique 0 ET n_1 la valeur logique 0 ET n_2 la valeur logique 1.

Maintenant, si $r = 0$ $\bar{r} = 1$ et de même pour n_1. Il est donc équivalent de dire que $S = 1$ si $\bar{r} = 1$ ET $\bar{n}_1 = 1$ ET $n_2 = 1$. Or nous connaissons l'opérateur logique traduisant cette condition : c'est simplement le produit logique des trois variables $\bar{r}$, $\bar{n}_1$ et n_2, ce qui nous permet d'écrire l'équation logique correspondante.

Le même raisonnement permet d'écrire les autres équations logiques du tableau. Ceci fait, nous pouvons passer aux équations logiques complètes permettant de calculer R et S dans tous les cas.

La table de vérité nous montre que S prend la valeur logique 1 si $r = 0$ ET $n_1 = 0$ ET $n_2 = 1$ OU BIEN si $r = 0$ ET $n_1 = 1$ ET $n_2 = 0$ OU BIEN si $r = 1$ ET $n_1 = 0$ ET $n_2 = 0$ OU BIEN si $r = 1$ ET $n_1 = 1$ ET $n_2 = 1$. L'opérateur logique correspondant est donc l'opérateur OU, ce qui nous donne l'équation logique de S :

$$S = \bar{r} \cdot \bar{n}_1 \cdot n_2 + \bar{r} \cdot n_1 \cdot \bar{n}_2 + r \cdot \bar{n}_1 \cdot \bar{n}_2 + r \cdot n_1 \cdot n_2.$$

De la même façon, nous trouvons l'équation logique de R :

$$R = \bar{r} \cdot n_1 \cdot n_2 + r \cdot \bar{n}_1 \cdot n_2 + r \cdot n_1 \cdot \bar{n}_2 + r \cdot n_1 \cdot n_2.$$

Quelques manipulations algébriques (de BOOLE !) simples, que nous ne détaillons pas ici conduisent aux relations équivalentes, mais plus faciles à utiliser :

$$S = r \star (n_1 \star n_2)$$

$$R = n_1 \cdot n_2 + r \cdot (n_1 \star n_2)$$

où le symbole $\star$ désigne, rappelons-le, l'opérateur logique OU EXCLUSIF.

Nous traduisons ces opérations par un schéma électronique (fig. 7 *a*). Des circuits intégrés TTL permettant de réaliser ce montage et donc l'addition algébrique de deux chiffres binaires avec retenue sont par exemple le 7408 (ET), le 7432 (OU) et le 7486 (OU EXCLUSIF).

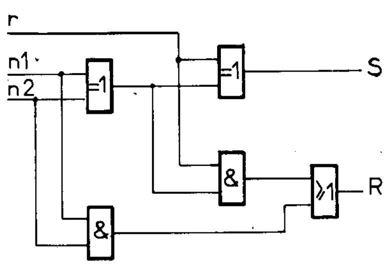

Fig. 7 *a*. — Addition de deux nombres avec retenue.

Pour additionner deux nombres binaires de n chiffres chacun, il convient de disposer n circuits équivalents au précédent en parallèle, l'entrée r de chacun étant liée à la sortie R du précédent, sauf celle des unités qui doit être bien entendu forcée à 0 (fig. 7 *b*).

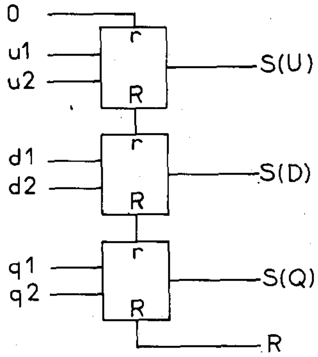

Fig. 7 *b*. — La mise en parallèle de trois circuits additionneurs permet d'ajouter deux nombres de trois chiffres en tenant compte des retenues.

Comparaison.

Il s'agit de circuits permettant de déterminer si deux nombres, toujours écrits dans le système binaire, sont égaux ou non et, dans la seconde hypothèse, de les classer par ordre de grandeur.

Voyons d'abord le cas simple de deux nombres ne comportant qu'un seul chiffre chacun. Nous devons prévoir deux entrées, les deux nombres n_1 et n_2, et trois sorties S_1, S_2, S_3, une pour chaque condition possible : $n_1 > n_2$, $n_1 = n_2$ et $n_1 < n_2$, respectivement. Nous convenons que $S_1 = 1$ si $n_1 > n_2$ et que $S_1 = 0$ sinon, et posons des conventions analogues pour les états logiques des deux autres sorties. Nous pouvons à présent dresser la table de vérité de la comparaison des deux nombres :

n_1	n_2	$n_1 > n_2$	$n_1 = n_2$	$n_1 < n_2$
0	0	0	1	0
0	1	0	0	1
1	0	1	0	0
1	1	0	1	0

Un raisonnement analogue à celui utilisé dans le paragraphe précédent nous conduit aux équations logiques donnant les états des sorties en fonction de ceux des entrées et à un schéma possible de circuit électronique (fig. 8 *a*).

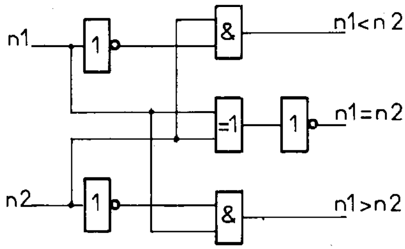

Fig. 8 *a*. — Circuit de comparaison de deux nombres binaires de un chiffre.

$$S_1 = n_1 \cdot \overline{n}_2$$

$$S_2 = \overline{n_1 * n}\cdot$$

$$S_3 = \overline{n}_1 \cdot n_2.$$

Pour deux nombres de deux chiffres A = $d_1 u_1$ et B = $d_2 u_2$, u_1 et u_2 étant les chiffres des unités et d_1 d_2 ceux des « deuxaines », nous trouverions de la même façon les équations logiques des trois sorties ainsi qu'un schéma électronique envisageable (fig. 8 *b*).

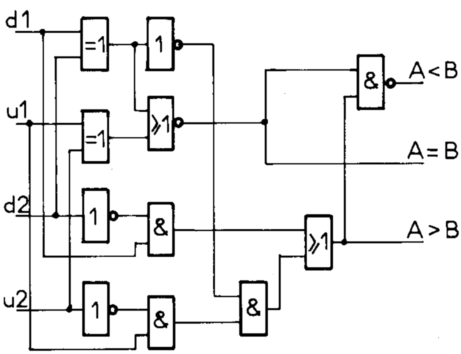

Fig. 8 *b*. — Circuit de comparaison de deux nombres binaires de deux chiffres.

$$S_1 = d_1 \cdot \overline{d}_2 + u_1 \cdot \overline{u}_2 \cdot (\overline{d_1 * d_2}) \qquad A > B$$

$$S_2 = \overline{(u_1 * u_2) + (d_1 * d_2)} \qquad A = B$$

Nous pourrions déterminer aussi l'équation logique de S_3, correspondant à $A < B$. Il est plus simple de remarquer que $S_3 = \overline{S_1 \cdot S_2}$.

Remarque.

Les circuits d'addition et de comparaison forment le cœur des microprocesseurs, circuits de calcul des ordinateurs. En micro-informatique, les microprocesseurs utilisés permettent de « traiter » des nombres de huit, seize ou, depuis peu, trente-deux chiffres. C'est là l'origine et la raison de leur classement en « 8 bits », « 16 bits » ou « 32 bits ».

Transcodage.

Les circuits logiques, quelle que soit leur réalisation pratique, ne connaissent que le système binaire. Dans la plupart des cas, cette écriture est mal adaptée à l'être humain, utilisateur final des informations ainsi obtenues.

Il convient donc de traduire les résultats en un langage plus accessible, par exemple le système décimal pour les chiffres ou l'alphabet pour les lettres. L'exemple le plus connu et sans doute le plus utilisé est l'afficheur « 7 segments » familier à tous les utilisateurs de calculettes. Un autre est la matrice de 8×8 pixels (points lumineux) grâce à laquelle les micro-ordinateurs affichent les chiffres et les lettres sur l'écran de visualisation.

Ces opérations sont réalisées à l'aide de circuits dits de transcodage dont nous allons à présent donner le principe. Il s'agit de circuits combinatoires à n entrées et p sorties, n et p dépendant de la traduction particulière envisagée.

Voyons le principe de l'afficheur « 7 segments ». Il est formé de sept barres qui peuvent être « allumées » ou « éteintes », au sens strict pour les afficheurs à diodes électroluminescentes, au sens figuré pour les afficheurs à cristaux liquides (fig. 9). Par

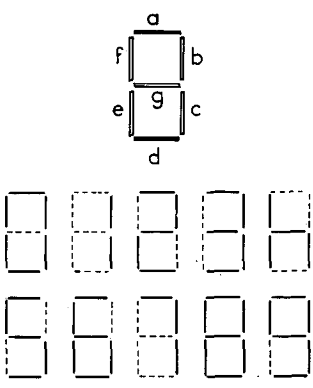

Fig. 9. — Afficheur « 7 segments » et formation des dix symboles du système décimal.

combinaison convenable des barres « allumées » et « éteintes », il est possible de représenter les dix chiffres du système décimal (fig. 9). Par ailleurs, il suffit de quatre chiffres pour représenter ces mêmes symboles dans le système binaire. Le circuit de transcodage commandant l'afficheur doit donc posséder quatre entrées et sept sorties.

Compte tenu de la représentation des chiffres décimaux par les sept segments, la table de vérité du circuit transcodeur est la suivante.

Nous désignons les quatre entrées par n_1, n_2, n_3 et n_4 et les sept sorties par a, b, c, d, e, f et g selon l'usage adopté par les fabricants.

n_1	n_2	n_3	n_4	a	b	c	d	e	f	g
0	0	0	0	1	1	1	1	1	1	1
0	0	0	1	1	1	0	0	0	0	0
0	0	1	0	1	1	0	1	1	0	1
0	0	1	1	1	1	1	1	0	0	1
0	1	0	0	0	0	1	0	1	1	1
0	1	0	1	1	0	1	1	0	1	1
0	1	1	0	1	0	1	1	1	1	1
0	1	1	1	1	1	1	0	0	0	0
1	0	0	0	1	1	1	1	1	1	1
1	0	0	1	1	1	1	1	0	1	1

A partir de cette table de vérité, nous pourrions établir un jeu d'équations logiques, selon la procédure exposée précédemment. Nous ne le ferons pas, afin de ne pas alourdir l'exposé et nous bornerons à donner un exemple de réalisation, d'après un document de fabricant (fig. 10).

Nous allons voir un exemple plus simple, permettant d'obtenir directement les dix chiffres décimaux à partir de leur équivalent binaire. Un tel transcodeur est par exemple utilisé pour commander les tubes Nixie, tubes à gaz contenant dix électrodes. Chaque électrode a la forme de l'un des dix chiffres décimaux et s'entoure d'une gaine luminescente quand elle est excitée.

Le circuit de commande doit donc avoir quatre entrées, comme pour l'afficheur sept segments, et dix sorties, une pour chaque électrode du tube Nixie. Sa table de vérité est la suivante.

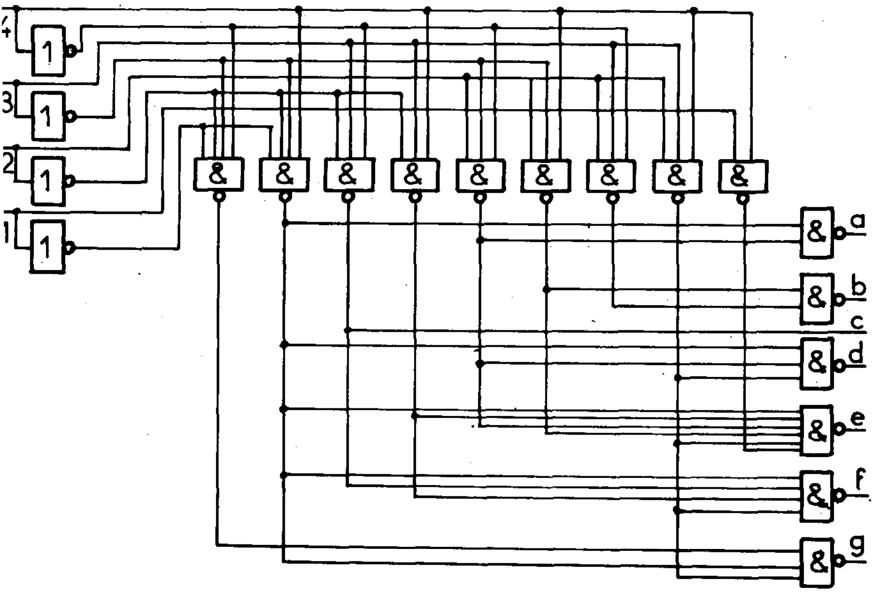

Fig. 10. — Schéma de principe d'un circuit transcodeur permettant de commander un afficheur à sept segments. Noter que les opérations ET-NON peuvent avoir plus de deux entrées.

n_1	n_2	n_3	n_4	0	1	2	3	4	5	6	7	8	9
0	0	0	0	1	0	0	0	0	0	0	0	0	0
0	0	0	1	0	1	0	0	0	0	0	0	0	0
0	0	1	0	0	0	1	0	0	0	0	0	0	0
0	0	1	1	0	0	0	1	0	0	0	0	0	0
0	1	0	0	0	0	0	0	1	0	0	0	0	0
0	1	0	1	0	0	0	0	0	1	0	0	0	0
0	1	1	0	0	0	0	0	0	0	1	0	0	0
0	1	1	1	0	0	0	0	0	0	0	1	0	0
1	0	0	0	0	0	0	0	0	0	0	0	1	0
1	0	0	1	0	0	0	0	0	0	0	0	0	1

Nous en déduisons les équations logiques.

$0 = \overline{n_1} \cdot \overline{n_2} \cdot \overline{n_3} \cdot \overline{n_4}$ $\quad$ $1 = \overline{n_1} \cdot \overline{n_2} \cdot \overline{n_3} \cdot n_4$

$2 = \overline{n_1} \cdot \overline{n_2} \cdot n_3 \cdot \overline{n_4}$ $\quad$ $3 = \overline{n_1} \cdot \overline{n_2} \cdot n_3 \cdot n_4$

$4 = \overline{n_1} \cdot n_2 \cdot \overline{n_3} \cdot \overline{n_4}$ $\quad$ $5 = \overline{n_1} \cdot n_2 \cdot \overline{n_3} \cdot n_4$

$6 = \overline{n_1} \cdot n_2 \cdot n_3 \cdot \overline{n_4}$ $\quad$ $7 = \overline{n_1} \cdot n_2 \cdot n_3 \cdot n_4$

$8 = n_1 \cdot \overline{n_2} \cdot \overline{n_3} \cdot \overline{n_4}$ $\quad$ $9 = n_1 \cdot \overline{n_2} \cdot \overline{n_3} \cdot n_4$

Nous ne poursuivons pas plus avant l'étude des circuits transcodeurs. A titre d'exercice, le lecteur (patient !) pourra imaginer la table de vérité du circuit d'affichage d'un ordinateur. Rappelons que ce circuit comporte huit entrées (un octet) et soixante-quatre sorties, une pour chacun des pixels de la matrice permettant de dessiner les chiffres et les lettres. Plus simple, mais basée sur le même principe est la matrice 5×7 à trente-cinq points lumineux couramment utilisée pour les « journaux lumineux » ou autres affichages du même genre.

Logique séquentielle.

Nous avons vu, en logique combinatoire, que les états logiques des sorties étaient fixés sans ambiguïté par ceux des entrées. En logique séquentielle, par contre, les états logiques des sorties dépendent non seulement des états des entrées mais aussi de leur succession dans le temps.

Le circuit de base en logique séquentielle est la *bascule* dite aussi *bistable.* En dérivent toute une famille de fonctions et en particulier les mémoires et compteurs.

Bascule.

L'origine des bascules remonte, comme c'est le cas pour la plupart des fonctions de l'électronique, aux tubes à vide (bascule d'ECCLES-JORDAN). Toutefois, la seule technologie utilisée de nos jours est celle des circuits intégrés.

La bascule la plus simple est dite R-S, pour Reset-Set, et peut être réalisée à l'aide de deux opérateurs ET-NON convenablement reliés (fig. 11). Les deux entrées sont traditionnellement désignées par R et S, les deux sorties par Q et $\overline{Q}$, pour des raisons que nous examinerons plus loin.

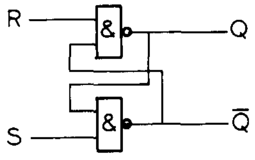

Fig. 11. — Schéma de principe d'une bascule R-S.

A partir de la table de vérité de l'opérateur ET-NON, nous pouvons établir celle de la bascule. Toutefois, lorsque R = 1 et S = 1, nous ne pouvons pas fixer directement les états des sorties. Nous devons faire une hypothèse et vérifier qu'elle ne conduit pas à une contradiction.

Il apparaît immédiatement que deux solutions sont possibles. En fait, dans un circuit réel, une seule d'entre elles est réalisée mais de façon imprévisible, selon les caractéristiques intimes des circuits intégrés qui la constituent. Nous remarquons par ailleurs que, sauf pour R = 0 et S = 0, les états logiques des deux sorties sont opposés. C'est là l'origine des symboles utilisés pour les représenter.

R	S	Q	$\overline{Q}$
0	0	1	1
0	1	0	1
1	0	1	0
1	1	1	0
1	1	0	1

Ceci étant, l'utilisation d'une bascule repose sur les modifications des états logiques des entrées. La table de vérité montre que si R = 0 ou S = 0 toute modification de l'état d'une seule entrée provoque celle de l'une des sorties. Jusque-là, nous n'avons rien de nouveau par rapport à la logique combinatoire. En fait, le cas intéressant est celui où R = 1 et S = 1. Nous partons alors de l'un des deux états indiqués dans la table de vérité, par exemple :

R	S	Q	$\overline{Q}$
1	1	0	1

Portons l'entrée S à l'état logique 0. Les deux sorties vont changer d'état (d'où le nom du circuit).

R	S	Q	$\overline{Q}$
1	0	1	0

Ramenons S à l'état logique 1. L'état précédent des sorties étant compatible avec le nouvel état des entrées, rien ne se passe. Tant que nous nous bornons à agir sur S, l'état des sorties ne dépend plus de celui des entrées. Par contre, si nous agissons alternativement sur R et sur S, les états des sorties changent une fois sur deux.

La bascule R - S est simple, mais présente l'inconvénient d'exiger d'agir sur les deux entrées. Des modèles plus élaborés ont été mis au point, qui suppriment cet inconvénient. L'un des plus utilisés est la bascule J - K dont nous indiquons simplement la table de vérité et le schéma d'utilisation, l'étude de sa réalisation étant hors de notre propos (fig. 12).

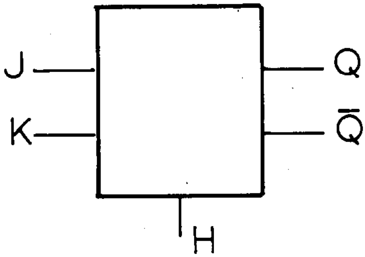

Fig. 12. — Bascule J-K : représentation symbolique.

Outre les deux entrées J et K, la bascule possède une entrée H, dite « entrée horloge » dont le rôle est le suivant. Tant que son état logique est 0, les états des sorties Q et $\overline{Q}$ restent fixés, et ceci quelles que soient les modifications des états des entrées J et K. Quand on porte H au niveau 1, il est possible de changer les états de Q et $\overline{Q}$, conformément à la table de vérité, mais la transformation n'apparaît qu'au moment où l'état de H redevient 0. En pratique, l'entrée H est commandée par des impulsions, aussi dit-on souvent que la bascule change d'état sur le front descendant de l'impulsion.

La table de vérité de la bascule J - K est alors :

J	K	Q(n)	Q($n+1$)
0	0	*	*
1	0	*	1
0	1	*	0
1	1	*	$\overline{*}$

Q(n) est l'état logique de la sortie Q avant d'appliquer une impulsion à l'entrée H et Q ($n+1$) l'état de cette même sortie après que l'état de H soit redevenu 0. L'état logique de $\overline{Q}$ est toujours opposé de celui de Q, et enfin * désigne indifféremment l'état logique 0 ou l'état logique 1.

Deux points intéressants doivent être notés : si J = 0 et K = 0, l'état des sorties de la bascule est fixé ; par contre, si J = 1 et K = 1, il change de valeur à chaque impulsion appliquée à l'entrée H. Nous allons voir à présent que ces particularités sont à la base des principales fonctions de la logique séquentielle.

Mémoires.

En électronique comme ailleurs, le problème se pose de conserver des informations. En logique binaire, il s'agit bien entendu des états des variables et/ou des fonctions logiques.

Nous avons vu lors de l'étude des réalisations pratiques que les informations portées par les fiches perforées ne pouvaient être effacées. Il est équivalent de dire qu'elles sont mises en mémoire, en ce sens qu'elles sont toujours disponibles une fois réalisée la fonction pour laquelle elles ont été enregistrées.

Toutefois l'encombrement et la lenteur d'utilisation des fiches perforées ont rapidement conduit à la mise au point de dispositifs électroniques jouant le même rôle et reposant pour l'essentiel sur les propriétés des bascules.

Le fonctionnement d'une mémoire repose sur trois points essentiels : la possibilité d' « écrire » une information, puis de la conserver et enfin de la « lire » pour l'utiliser.

Dans le cas d'une fiche perforée, l'écriture correspond à l'encochage de la fiche, la lecture à la constatation que la fiche tombe ou non, tandis que la conservation résulte des propriétés physiques du papier dans lequel la fiche est réalisée (un tissu vivant qui se cicatrise ne permettrait pas une mise en mémoire de ce type).

Nous pouvons repérer les mêmes fonctions dans le cas d'une bascule, par exemple la bascule J - K.

Tant que l'entrée H (horloge) est au niveau logique 0, les états logiques des sorties Q et $\overline{Q}$ sont invariables et en particulier indépendants de ceux des entrées J et K. Nous pouvons donc admettre qu'ils ont été mis en mémoire.

Ils sont constamment disponibles et un simple voltmètre nous permettrait de constater la présence ou l'absence d'une tension sur la connexion électrique associée, donc de lire l'information mise en mémoire. En ce qui concerne l'écriture, il suffit de déterminer à partir de la table de vérité les états des entrées en fonction de ceux auxquels on désire porter les sorties, puis de porter l'entrée H au niveau logique 1 pour réaliser l'enregistrement.

Les bascules J - K ou d'un modèle équivalent présentent l'inconvénient d'exiger au moins une entrée et une sortie par information élémentaire et donc d'être encombrantes (tout est relatif...). Des dispositifs plus élaborés permettent un adressage de type matriciel, par lignes et colonnes, et donc une réalisation pratique plus compacte.

Les progrès de la technique aidant, ces mémoires sont actuellement fabriquées en circuits intégrés dont la capacité s'exprime en octets ou groupes de huit informations élémentaires. Le record actuel (en 1986) est de 1 mégaoctet (en toute rigueur 1 048 576) soit environ un million de caractères ou l'équivalent d'un livre de cinq cents pages, mais certains constructeurs annoncent déjà des mémoires de quatre mégaoctets.

Notons à ce propos que la capacité de mémoire est de nos jours l'un des principaux — et quelquefois le seul — argument de vente des constructeurs de micro-ordinateurs.

Compteurs.

Comme leur nom l'indique, les compteurs servent... à compter. Plus précisément, il s'agit de circuits comportant une seule entrée et plusieurs sorties. L'entrée reçoit une suite d'impulsions ou, si l'on préfère, est portée alternativement au niveau logique 0 et au niveau logique 1. Les états des sorties matérialisent, selon un code connu, le nombre d'impulsions reçues par l'entrée. Le code choisi est bien entendu binaire, soit pur soit mixte. Le résultat du comptage est la plupart du temps transposé dans le système décimal à l'aide d'un circuit transcodeur comme ceux que nous avons décrit précédemment.

Nous pouvons citer tout de suite une application courante des circuits de comptage : si les impulsions se suivent à cadence régulière et connue, la détermination de leur nombre équivaut à la mesure du temps écoulé entre l'envoi de la première et celui de la dernière. En d'autres termes, l'association d'un générateur d'impulsions et d'un compteur constitue une horloge. C'est sur ce principe que reposent les montres et pendules digitales modernes, ainsi que les chronomètres électroniques utilisés en travaux pratiques.

La plupart des compteurs sont réalisés à partir de bascules. Nous allons décrire un circuit à quatre sorties permettant de compter jusqu'à 1111 dans le système binaire, soit 15 dans le système décimal. Nous partons de bascules J - K mais d'autres modèles sont utilisables (fig. 13).

Les entrées J et K de toutes les bascules sont maintenues au niveau logique 1. La table de vérité de la bascule J - K nous précise que dans ces conditions les sorties changent d'état après chaque envoi d'une impulsion sur l'entrée H. Nous supposons

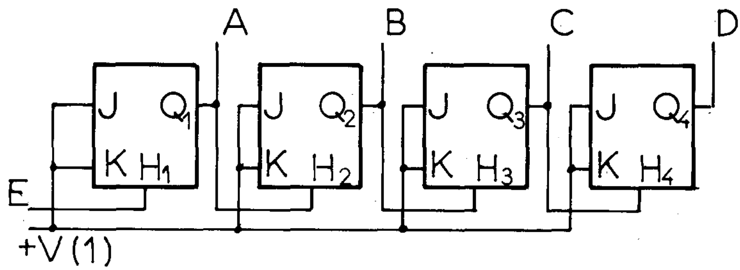

Fig. 13. — Compteur à quatre sorties.

pour fixer les idées que l'état logique initial des quatre sorties A, B, C et D est 0, ce qu'il est possible d'obtenir à l'aide d'un circuit auxiliaire. Portons l'entrée E au niveau 1 puis ramenons-la au niveau 0 (première impulsion). La sortie Q_1 et par suite A et H_2 passent au niveau 1. H_2 étant maintenue au niveau 1, Q_2 conserve son niveau antérieur, soit 0, et de même Q_3 et Q_4. Envoyons sur E une seconde impulsion. Q_1 change à nouveau d'état et revient au niveau logique 0, ce qui provoque le changement d'état de la seconde bascule. Q_2 et H_3 passent au niveau 1, ce qui maintient Q_3 et Q_4 au niveau 0. La troisième impulsion ramène Q_1 au niveau logique 1 mais ne change ni l'état de Q_2 ni, par conséquent, ceux de Q_3 et de Q_4, puisque l'entrée H_2 n'est pas revenue au niveau 0.

La poursuite du raisonnement permet d'établir la table de vérité complète de notre compteur.

N° d'impulsion	Etats des sorties			
	A	B	C	D
0	0	0	0	0
1	1	0	0	0
2	0	1	0	0
3	1	1	0	0
4	0	0	1	0
5	1	0	1	0
6	0	1	1	0
7	1	1	1	0
8	0	0	0	1
9	1	0	0	1
10	0	1	0	1
11	1	1	0	1
12	0	0	1	1
13	1	0	1	1
14	0	1	1	1
15	1	1	1	1
16	0	0	0	0

Cette table de vérité nous permet de noter quelques particularités intéressantes. D'abord la seizième impulsion ramène toutes les sorties au niveau logique 0, ce qui permet de recommencer un nouveau comptage. Puis le nombre formé des quatre chiffres définis par les états des sorties D, C, B, A *prises dans cet ordre* n'est autre que le rang de l'impulsion correspondante, exprimé dans le système binaire.

Par ailleurs, la sortie C, par exemple, ne change d'état que toutes les quatre impulsions. Cette dernière particularité représente un autre aspect important du comptage, la division de fréquence. Nous pouvons nous figurer que la sortie C fournit des impulsions de fréquence égale au quart de celle des impulsions appliquées à l'entrée E. Cette propriété est utilisée entre autres dans les montres à quartz. Le générateur d'impulsions primaires utilise un cristal de quartz oscillant à une fréquence de l'ordre du mégahertz. Par divisions successives, on passe à la fréquence commandant l'affichage, 1 Hz pour les secondes par exemple.

Signalons pour terminer une autre application : la réalisation de programmateurs électroniques pour machines où la mesure du temps est ramenée au comptage d'impulsions.

Registre à décalage.

Le registre à décalage permet de faire « circuler » une information élémentaire parmi plusieurs sorties. Nous prenons à nouveau l'exemple d'un circuit constitué de bascules J - K et comportant deux entrées et quatre sorties. L'une des deux entrées commande simultanément les entrées « horloge » de toutes les bascules. L'autre est reliée directement à l'entrée J_1 de la première bascule et par l'intermédiaire d'un opérateur logique NON à l'entrée K_1 de la même bascule, ce qui assure que les états logiques de J_1 et de K_1 sont toujours opposés (fig. 14).

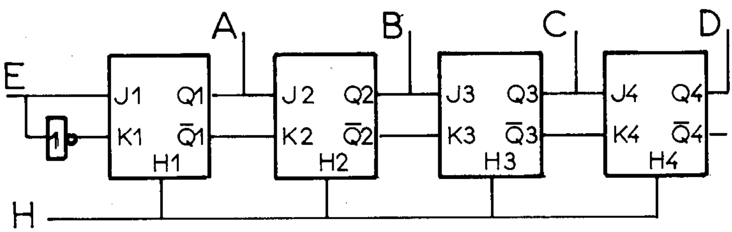

Fig. 14. — Registre à décalage à quatre sorties.

Nous supposons encore que l'état logique initial de toutes les sorties est 0.

Portons l'entrée E au niveau 1 puis appliquons une impulsion à l'entrée horloge commune H. Q_1 passe au niveau 1 et $\overline{Q}_1$ au niveau 0 comme le montre la table de vérité de la bascule J - K. Au moment où H passe au niveau logique 1, Q_1 est au niveau 0. Par suite la seconde bascule ne change pas d'état lorsque H revient au niveau 0.

Ramenons E au niveau 0 et appliquons une nouvelle impulsion à l'entrée H. Lorsque H passe au niveau 1, Q_1 et J_2 sont au niveau 1, $\overline{Q}_1$ et K_2 au niveau 0 et *Q_2 au niveau 0* ainsi que nous venons de le voir. Quand H revient à 0, Q_1 repasse également au niveau 0 et Q_2 prend l'état 1. La répétition du raisonnement montre qu'à chaque nouvelle impulsion appliquée sur l'entrée H, l'état logique 1 est décalé d'un cran, d'où le nom du circuit.

Il est bien entendu possible de modifier l'état de E entre deux envois d'impulsions vers l'entrée H. Les états correspondants sont décalés d'un cran à chaque impulsion et sont perdus au fur et à mesure qu'ils passent la dernière bascule.

Il faut noter que les états logiques introduits en série (l'un après l'autre) sur l'entrée E se retrouvent en parallèle (l'un à côté de l'autre) sur les sorties A, B, C et D. Cette transformation est d'utilisation courante dans les calculettes où les données sont par force introduites en série.

Si les sorties de la dernière bascule sont reliées aux entrées de la première, les informations introduites en E ne sont plus perdues mais circulent indéfiniment entre les sorties du registre. Celui-ci porte alors le nom de « compteur en anneau ». Citons, parmi les applications des compteurs en anneau, la réalisation de distributeurs (Delco) électroniques pour moteurs d'automobiles, plus fiables que les distributeurs électromécaniques actuels.

Générateurs de signaux.

Ces circuits, toujours réalisés à partir d'opérateurs logiques, changent périodiquement d'état sans intervention extérieure, contrairement aux circuits de logique combinatoire ou de logique séquentielle. Ce changement « spontané » résulte d'un couplage entre deux parties du circuit. De même que les bascules, il s'agit d'une adaptation de circuits en composants discrets.

Circuit monostable.

Sous sa forme la plus simple, il est formé de deux opérateurs ET-NON reliés par un condensateur (fig. 15).

Nous supposons qu'à l'origine des temps, le condensateur est déchargé et que l'entrée E_1 est au niveau logique 1. L'entrée E_2 du second opérateur est au niveau logique 0, par suite la sor-

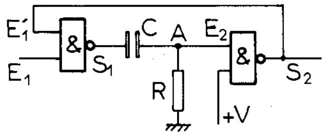

Fig. 15. — Schéma de principe d'un circuit monostable.

tie S_2 est au niveau logique 1 et la sortie S_1 est au niveau logique 0. La situation est stable et rien ne se passe.

Portons l'entrée E_1 au niveau logique 0. La sortie S_1 passe au niveau logique 1, c'est-à-dire que son potentiel passe à $+V$. Nous avons à partir de ce moment la séquence d'événements suivante.

Le condensateur, déchargé, se comporte comme un court circuit, le potentiel des points A et E_2 prend la valeur $+V$ si bien que la sortie S_2 et par voie de conséquence l'entrée E'_1 prennent l'état logique 0. De ce fait, la sortie S_1 se maintient au niveau logique 1 même si l'entrée E_1 est ramenée à 1. L'impulsion d'entrée peut donc être très courte.

Le condensateur se charge à travers la résistance R et le potentiel du point A diminue. Quand il est revenu à zéro, la sortie S_2 repasse au niveau logique 1 et S_1 au niveau logique 0. Le condensateur se décharge alors dans la résistance R sans provoquer de nouveau changement.

Le diagramme ci-après (fig. 16) résume le principe de fonctionnement du circuit monostable.

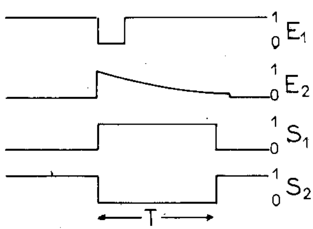

Fig. 16. — Evolution au cours du temps des états logiques de divers points du circuit monostable de la fig. 15.

La durée T pendant laquelle S_2 reste dans l'état logique 0 dépend de la rapidité avec laquelle se charge le condensateur, c'est-à-dire de la constante de temps $R \cdot C$ et il est possible de la modifier en jouant sur les valeurs de R et/ou de C.

D'autres types de circuits monostables permettent d'obtenir en sortie un « créneau » positif, mais le principe de fonctionnement reste le même.

L'application principale des circuits monostables est la réalisation de temporisateurs, qui permettent de faire fonctionner un appareil pendant un temps déterminé, ou de ne le mettre en marche qu'après un certain temps...

Circuit astable.

Au lieu de relier directement la sortie S_2 et l'entrée E'_1, nous pouvons les coupler par l'intermédiaire d'un second condensateur et d'une seconde résistance (fig. 17). L'état défini par $S_1 = 0$, $S_2 = 1$ et $E'_1 = 1$ n'est plus stable et ne se maintient que pendant un temps T_2 défini par R_2 et C_2.

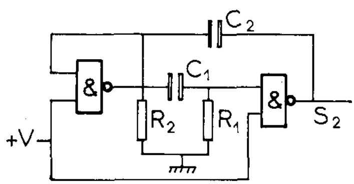

Fig. 17. — Schéma de principe d'un circuit astable ou générateur de signaux « carrés ».

La sortie S_2 passe périodiquement du niveau 0 au niveau 1 et *vice versa.* Elle reste au niveau 0 pendant une durée T_1 définie par R_1 et C_1, puis au niveau 1 pendant une durée T_2 définie par R_2 et C_2. Il est facile de faire varier le rapport T_1/T_2 dit *rapport cyclique.*

Le circuit astable est couramment utilisé pour fournir des signaux d'horloge, lorsqu'une grande précision n'est pas nécessaire. Il sert également dans certains instruments de musique électronique dits synthétiseurs. Notons à ce propos que le signal de sortie est plutôt « carré » que sinusoïdal, donc plus riche en harmoniques, ce qui explique la sonorité particulière de la musique électronique.

EXPERIENCES DE DEMONSTRATION.

Au niveau du collège, les seules expériences envisageables sont la vérification des tables de vérité des opérateurs fondamentaux : ET, OU, NON, ET-NON, OU-NON. Il suffit de suivre les recommandations des fabricants, en ce qui concerne les tensions et intensités admissibles par les circuits intégrés correspondants.

Il est commode de « visualiser » les états des entrées et des sorties à l'aide de diodes électroluminescentes. Or les sorties ne débitent pas, sauf exception, une intensité suffisante. Il convient alors de passer par l'intermédiaire d'un transistor amplificateur de courant et fonctionnant en mode bloqué/saturé. Le schéma ci-après donne un exemple de montage, les valeurs des résistances pas plus que le type de transistor n'étant très critiques (fig. 18).

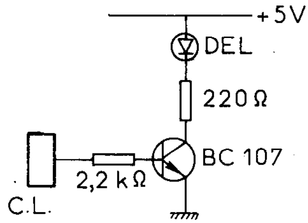

Fig. 18. — Indicateur d'état logique de la sortie d'un circuit logique (C. L.).

En classe de 6e, il est possible de montrer que les opérations de logique interviennent dans des situations somme toute assez courantes. Voici un exemple parmi d'autres : deux bicyclettes sont liées par une chaîne et deux cadenas. Quand les deux cadenas sont « en série » avec les extrémités de la chaîne l'un OU l'autre des détenteurs des clefs peut libérer les engins. Par contre, si les deux cadenas sont « en parallèle », il faut que l'un ET l'autre des propriétaires soient présents. L'extension à un nombre quelconque de variables logiques (cadenas) est immédiate. Vous pourrez en profiter pour faire remarquer, en comparant avec des interrupteurs électriques, que la définition d'une opération logique est étroitement liée à sa réalisation matérielle.

BIBLIOGRAPHIE

F. Huré. — *Initiation pratique à l'emploi des circuits intégrés digitaux.* E.T.S.F. Paris 1977.

F. Huré. — *Traité expérimental de logique digitale.* E.T.S.F. Paris 1979.

Ces deux ouvrages sont plutôt orientés vers l'aspect pratique et comportent une partie expérimentale importante.

G. McWhorter. — *Understanding digital electronics.* Texas Instruments Dallas 1984.

Destiné aux élèves du centre de formation de la société Texas. Très pragmatique et très intéressant.

H. Ney. — *Eléments d'automatismes.* Nathan Paris 1985.

Destiné à l'enseignement technique. Quelques chapitres traitent des opérateurs logiques.

J.-M. Rolando. — *Initiation à la conception du matériel en informatique.* B.U.P. n° 686 p. 1093. Juillet 1986.

P. Tougné. — *Jeux mathématiques.* Pour la Science. Septembre 1986 p. 11.

B. Woollard. — *Conquérir la logique.* Pratiguide Dunod Paris 1980.

Assez clair et complet.

The TTL Data Book. Texas Instruments Inc. Dallas 1976.

L'un des nombreux (et indispensables) recueils de données, qui vous permettront de tout savoir sur les circuits intégrés logiques : désignation, caractéristiques, connexions électriques...