C.D.U./U.D.C.: 003.62::681.325.6.004.14 744.423::681.3.056

COMMISSION ÉLECTROTECHNIQUE INTERNATIONALE

NORME DE LA CEI

INTERNATIONAL ELECTROTECHNICAL COMMISSION

IEC STANDARD

Publication 113-7

Première édition — First edition

1981

Schémas, diagrammes, tableaux

Septième partie: Etablissement des logigrammes

Diagrams, charts, tables

Part 7: Preparation of logic diagrams

Mots clés: symboles d'opérateurs logiques; établissement des schémas.

Key words: symbols for logic elements; preparation of diagrams.

Bureau Central de la Commission Electrotechnique Internationale

1, rue de Varembé

Genève, Suisse

Prix Price Fr.s. **65.—**

CONTENTS

Page

FOREWORD 5

PREFACE 5

Clause

1. Scope 7
2. Related publications 7
3. Symbols 7
4. Logic conventions 11
5. Contents of a logic diagram 11
6. Layout of a diagram 11
7. Unused elements 13
8. Distributed connections 13
9. Complex logic units 13
10. Symbols for logic elements 13
11. Labelling of lines 13
12. Use of waveforms 25
13. Tabular information 25
14. Representation of a pin connection acting at different times as either an input or an output . 25
15. Interaction of logic elements and other devices 25
16. Examples of logic diagrams 26

APPENDIX A — Commonly used mnemonics 30

APPENDIX B — Guide to the meaning, interpretation and selection of logic symbols 35

INTERNATIONAL ELECTROTECHNICAL COMMISSION

DIAGRAMS, CHARTS, TABLES

Part 7: Preparation of logic diagrams

FOREWORD

1) The formal decisions or agreements of the IEC on technical matters, prepared by Technical Committees on which all the National Committees having a special interest therein are represented, express, as nearly as possible, an international consensus of opinion on the subjects dealt with.

2) They have the form of recommendations for international use and they are accepted by the National Committees in that sense.

3) In order to promote international unification, the IEC expresses the wish that all National Committees should adopt the text of the IEC recommendation for their national rules in so far as national conditions will permit. Any divergence between the IEC recommendation and the corresponding national rules should, as far as possible, be clearly indicated in the latter.

PREFACE

This standard has been prepared by Sub-Committee 3B: Preparation of Diagrams, Charts and Tables. Item Designation, of IEC Technical Committee No. 3: Graphical Symbols.

Drafts were discussed at the meetings held in Milan in 1975 and in Oslo in 1976. As a result of this latter meeting, a draft, Document 3B(Central Office)19, was submitted to the National Committees for approval under the Six Months' Rule in December 1976.

The National Committees of the following countries voted explicitly in favour of publication:

Australia
Austria
Belgium
Canada
Denmark
Egypt
Finland
France
Israel
Japan
Korea (Republic of)
Netherlands
South Africa (Republic of)
Spain
Sweden
Switzerland
Turkey
United Kingdom
Yugoslavia

The document was not published at that time, however, as, during the preparatory work, it was realized that additional material would be needed. A first draft for this supplement, comprising Clauses 11, 14, 15 and Appendices A and B, was discussed at the meeting held in Paris in 1978. As a result of this meeting, a combined draft, Document 3B(Central Office)22, which included the contents of Document 3B(Central Office)19 and the text of the draft supplement, was submitted to the National Committees for approval under the Six Months' Rule in May 1979.

The National Committees of the following countries voted explicitly in favour of publication of the draft supplement:

Australia
Austria
Belgium
Canada
Egypt
Finland
Germany
Israel
Italy
Japan
Norway
South Africa (Republic of)
Sweden
Switzerland
Turkey
Union of Soviet Socialist Republics
United Kingdom

Other IEC publications quoted in this standard:

Publications Nos. 113-3: Diagrams, Charts, Tables, Part 3: General Recommendations for the Preparation of Diagrams.

113-4: Part 4: Recommendations for the Preparation of Circuit Diagrams.

117-15: Recommended Graphical Symbols, Part 15: Binary Logic Elements.

617-12: Graphical Symbols for Diagrams, Part 12: Binary Logic Elements *(in preparation)*. Until this standard is published reference should be made to Publication 117-15, quoted above.

DIAGRAMS, CHARTS, TABLES

Part 7: Preparation of logic diagrams

1. Scope

This standard gives additional recommendations for the preparation of diagrams with respect to the use of symbols for logic elements.

2. Related publications

The other parts of IEC Publication 113* are concerned with matters of drawing practice and provide additional guidance.

Graphical symbols for diagrams are given in the relevant parts of IEC Publication 117: Recommended Graphical Symbols, and IEC Publication 617: Graphical Symbols for Diagrams*.

3. Symbols

3.1 *General*

Symbols for binary logic elements and some rules for the use of the symbols are given in IEC Publication 617-12.

In accordance with Sub-clause 3.4 of IEC Publication 113-3, their size should be governed by the space necessary for internal annotations and the length of the side needed to accommodate input and output lines at an acceptable spacing.

Input and output lines are preferably placed on opposite sides of the symbol and should join the outline of the symbol at right angles. A symbol for a binary logic element may have any number of inputs and outputs provided that the symbol definition requirements as contained in IEC Publication 617-12 are met.

The special rules for complex elements must be observed.

Most logic diagrams will also require the use of symbols drawn from some of the other parts of IEC Publications 117 and 617.

3.2 *Combination of symbols*

Rules for combining symbols for basic operations are given in IEC Publication 617-12. Symbols representing hardware elements which are contained in one physical package may also be combined providing the interconnections are not accessible to the user and the rules mentioned above are applied.

---

* See inside back cover.

3.3 *Symbol orientation*

The symbols contained in IEC Publication 617-12 have been designed so that inputs appear on the left and outputs on the right, and this orientation is preferred. However, the other symbol orientation in which inputs appear at the top and outputs at the bottom may also be used, provided the following rules are observed:

a) In general, outputs appear on the opposite side of the symbols from the inputs.

b) The rules for the placement of the qualifying symbol(s) for the device function given in IEC Publication 617-12 are obeyed.

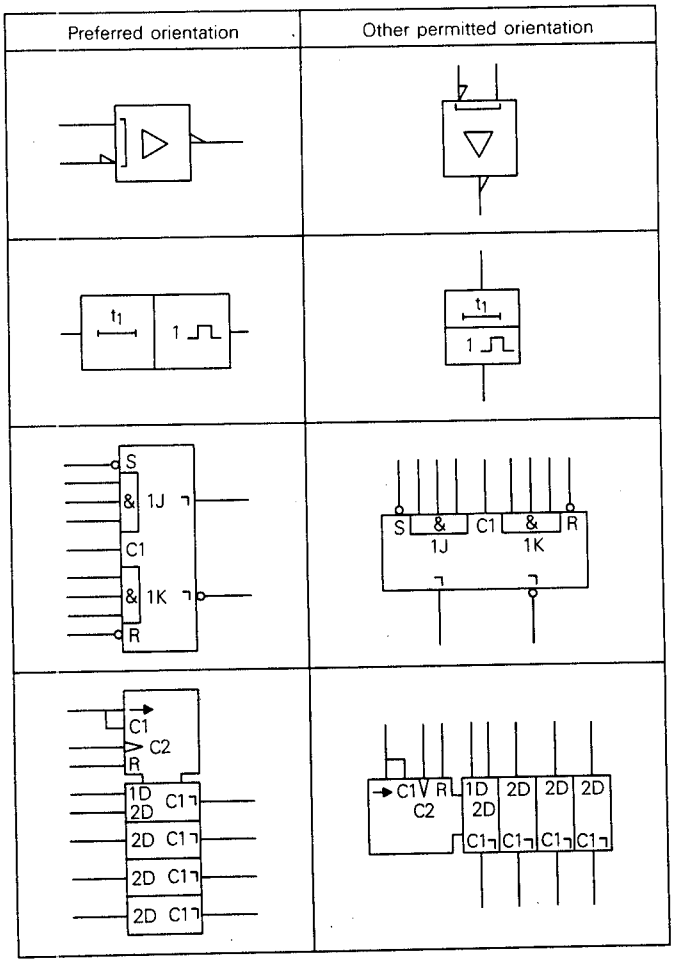
c) Sufficient space is provided to avoid confusion between the qualifying symbol(s) for the function and the input and output labels.

d) Strings of characters, forming qualifying symbols or labels, should preferably appear in normal reading order. Each input (output) label is centred with respect to its input (output) line.

e) The orientation of qualifying symbols for negation, polarity, grouping dynamic input, inhibit and amplification with respect to the connection lines is unchanged.

Table I gives examples of symbol orientation.

TABLE I



4. Logic conventions

4.1 A logic diagram may be prepared using a single logic convention, either positive or negative, to define the relationship between logic states and the nominal values of corresponding physical quantities. In this case, and in the case where no such relationship needs to be established, the symbol for logic negation is used where necessary; the symbol for logic polarity shall not be used. The convention in use should be clearly shown on the diagram or in referenced documentation (see Figure 2, page 28).

Note. — Different logic conventions may be used for different parts of the same diagram, e.g. on either side of an interface between contrasting technologies—the convention applying to each part being clearly shown and the areas of the diagram to which each applies being clearly delineated.

4.2 Alternatively the relationship between the logic state and the nominal value of the corresponding physical quantity at each input and output of every logic element may be given by means of the absence or presence of the symbol for logic polarity (see Figure 3, page 29). In this case the symbol for logic negation shall not be used.

4.3 It is sometimes convenient to represent the logic convention by means of idealized waveforms with indications of the logic states and, if necessary, of the nominal value of corresponding physical quantities. Additional information is provided in Appendix B.

5. Contents of a logic diagram

A logic diagram—which may consist of several sheets—should be prepared for each distinct unit, or assembly of units, intended to fulfil a defined purpose. It may thus relate to a single unit or to several units which together form a functional entity.

In cases where the logic diagram cannot be shown on a single sheet the division into separate sheets should be based on the purpose of the diagram.

6. Layout of a diagram

The most important consideration is the adoption of a layout which best aids understanding of the diagram. The layout should be such that the principal flow of information is from left to right of the diagram—otherwise it should be from top to bottom. Where the direction of information flow is not obvious, lines carrying information should be marked with an arrow-head which must not be placed adjacent to any other symbol or label.

Functionally related symbols should be grouped and placed as close to one another as the requirements of annotation and the avoidance of overcrowding will allow.

Lines should be drawn horizontally or vertically except in those isolated cases where oblique lines aid the clarity of the diagram.

For a signal feeding a multiplicity of elements the use of a single straight line with appropriate indications of T-junctions to the elements aids comprehension of the diagram.

The recommendations for grouping and omission of lines given in Clause 4 of IEC Publication 113-3 apply also to lines representing information flow and connecting lines in logic diagrams.

7. Unused elements

Any hardware elements which are not used—e.g. inputs or outputs or complete elements in a multiple element package—may be shown. It is then important to identify the unused elements and the corresponding pin numbers.

8. Distributed connections

The connection of certain logic elements to achieve the effect of an AND or an OR operation without the use of additional logic elements may be depicted as shown in IEC Publication 617-12. All external components required to implement the facility should be shown or referred to in a table.

Any interrupted line should be appropriately cross-referenced in accordance with IEC Publication 113-4.

9. Complex logic units

The correct way to represent a complex logic unit is to use a symbol developed in accordance with the rules given in IEC Publication 617-12. If this is not desirable owing to the complexity of the function, then the symbol used can be a simple rectangle containing a description of the function or a reference to appropriate documentation. Standard labels should be used for inputs and outputs when possible.

If a symbol for a complex function has to be repeated on a diagram, the repetitions may be shown as simple rectangles with appropriate references.

10. Symbols for logic elements

Preferably each logic element should be shown by that symbol which best depicts the logic function actually performed by the element in the system. Thus, in Figure 2, page 28, the same type of hardware element (designated D3) is represented once by the symbol for an OR element with negated inputs and elsewhere by the symbol for an AND element with negated output (NAND).

Additional information is provided in Appendix B.

11. Labelling of lines

11.1 *General*

Labelling of lines can greatly promote the understanding of a diagram and facilitate the maintenance of a logic system provided that the signals are named intelligently and the name allocation is based on carefully named system functions.

Signal line names should be informative and unambiguous. The space available on the diagram or allocated in the data file usually limits the allowable length of signal names. Identical names shall not be applied to different signal lines, no matter how similar the

function. Every effort should be made to use mnemonic names and standard abbreviations (for examples, see Figures 2 and 3, pages 28 and 29). The mnemonics and abbreviations used should be explained on the diagram or in supporting documentation.

In general, the characters comprising the total signal identification provides three types of information:

the first gives signal cross-reference information;

the second, referred to in this standard as the signal name, gives generally a functional description;

the third (used only on diagrams employing the polarity indicator) gives information which relates the logical truth of a function with its logic level.

Only the second and third types of information are dealt with in this standard.

11.2 *Signal names*

The signal name is an abbreviation of a statement which can be either true or false. For example: the name ALARM is associated with the statement "ALARM IS ACTIVATED".

True and complement signals of the same function are often required on binary logic diagrams. The preferred method to denote the complement of a signal is by use of a negation bar over the signal name, thus: ALARM, $\overline{\text{ALARM}}$.

If only an in-line notation can be used, the notation adopted to designate the complementary signals shall be explained on the diagram or in supporting documentation, for example ALARM-N.

Convention: The relation between the true value of a statement, represented by a signal name, and the logic state of the binary-digital variable assigned to the signal with that signal name, is given by the following general convention:

True always corresponds to the logic *1-state.*

False always corresponds to the logic *0-state.*

For example: the signal name ALARM implies that if ALARM is true, the signal is in its 1-state. Conversely, $\overline{\text{ALARM}}$ implies that if ALARM is true, the signal is in its 0-state.

When a *single logic convention* is used for all inputs and outputs (i.e. positive or negative logic), this defines the relationship between logic states and logic levels. Thus, if the positive logic convention is in force, true always corresponds to the logic H-level and, if the negative logic convention is in force, true corresponds to the logic L-level.

When the *logic polarity indicator* is used for each input and output, the signal name should preferably be followed by the logic level (between brackets) for which the statement represented by the signal name is true, e.g. ALARM(L) or ALARM(H).

11.2.1 *General rules for signal name allocations*

If space permits, easy to understand mnemonics should be used instead of short notations. For example, SELDEV1 rather than SD1 would better convey the meaning (SELECT DEVICE 1).

A signal name must be changed whenever the signal is gated with another, is delayed, chopped, stored, or changes its function in any way; see examples 4 and 7. On the contrary, no signal may change its name as a result of being inverted, amplified, or level shifted (see example 6 in Sub-clause 11.2.11).

Where possible, signals should be named by the function they perform instead of by the signals that generate them. Assume that a signal, PRUN, is gated with a second signal, TP6, to set a bistable element called RUN. If the output signal is called SETRUN, its function is obvious. However, if it is called PRUNTP6 its use is open to speculation (see example 1).

Other signals perform such varied functions that it is not feasible to label them according to their use. For example, the alternative instructions to READ or WRITE may control large blocks of logic. In this event it would be proper to name the signal RD/WR (READ or WRITE); see example 2.

Confusion sometimes occurs in assigning signal names when the true value causes one action and the false value causes a different action. For example, a bistable element may select device A when cleared and device B when set. In this event the signal name should not refer to either device A or B. It should be such as SELDEV with a note on the drawing to explain the selection of device A or B.

Signal names that embody an inherent negative, such as NORUN, sometimes require mental somersaults, and are therefore naturally difficult to understand. If possible, such signal names should be made inherently true; e.g. STOP or HALT can be substituted for NORUN.

11.2.2 *Recommended characters for signal names*

Signal names should be composed from standard character sets, excluding lower-case letters. Also, a single character space may be used where necessary although the use of the dot as a separator often proves advantageous over the space.

Design automation may place certain constraints on the use of characters and the length of signal names.

11.2.3 *Use of serial numbers*

If the same function is generated more than once, each occurrence should have the same signal name and be identified by a different serial number. For example, multiple negators or amplifiers driven by the same signal would have the same signal name, but different serial numbers. Thus, $\overline{\text{STOP-1}}$ and $\overline{\text{STOP-2}}$ may be the output signals of two negators which are both driven by the signal STOP (see examples 5, 6, 9 and 10).

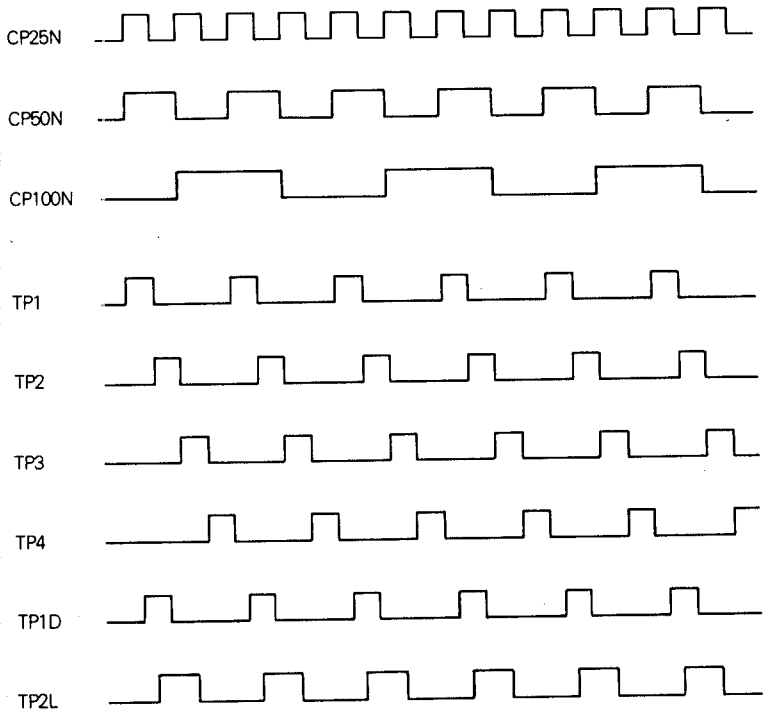
11.2.4 *Language requirements*

There is no restriction as to the use of a language for the formation of signal names. The signal names used in this standard may therefore not necessarily be translated literally when using another language.

11.2.5 *Clock pulses*

The basic clock and all clocks derived therefrom may be designated by CP (Clock Pulse) and identified by the clock period. If the basic clock period is, for example, 25 ns, the mnemonic might be CP25N. Clocks derived from the basic clock might then be termed CP50N, CP100N, and so on.

The timing pulses derived from the clock pulses may be designated by TP and identified by a serial number which indicates their respective order of appearance.



Pulses which are delayed, stretched or shortened may be identified by a suffix D, L or S respectively. TP1D, TP2L, TP2DL, TP4S may represent such pulses.

11.2.6 *Signals of bistable elements*

The output signals of bistable elements usually exhibit the same logic levels as signals from combinative elements. It is therefore not necessary to distinguish them from other outputs; see examples 7 and 8. However, if it is desirable to alert the user to the fact that he is dealing with a signal originating at a bistable element the suffix FF may be used; e.g. STARTFF, etc.; see example 9.

11.2.7 *Representation of logical OR and AND*

In special cases it may be useful to employ the solidus "/" as the logical OR symbol and normal juxtaposition to indicate the logical AND. For example, RD/WR and PRUN may be such signals (see examples 2 and 1).

The use of logic equations in signal names is not recommended.

11.2.8 *Addition and subtraction*

The plus "+" sign denotes algebraic addition and the minus "−" sign denotes algebraic subtraction; for example, MAR + 1 may be the mnemonic for "Memory Address Register Plus 1".

11.2.9 *Bit and byte labelling*

To accord with the normal order of reading, bit and byte labelling should proceed from the least significant (0) to the most significant. Representation should then be in ascending order of significance from top to bottom or from left to right. For example, the 32 lines of an intermediate register may be labelled IR00 to IR31. If any other convention is used, it should be explained on the diagram or in supporting documentation, e.g. IR00 ≙ MSB (Most Significant Bit) and IR31 ≙ LSB (Least Significant Bit).

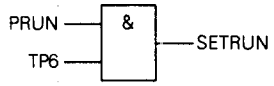
Parity bits and check bits may be designated by "P" and "C" respectively and related to the corresponding byte by a suffix number. For example, MRP3 may represent the parity bit of byte 3 of the memory register.

11.2.10 *Bus lines*

Bus lines may be designated by the term "BUS". For example, CBUS00 may be the signal name for line 00 on bus C.

11.2.11 *Examples of signal name allocation*

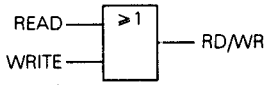
1



Labelling of a non-negated output of a combinative element.

PRUN may stand for PRE RUN and TP6 for TIMING PULSE 6.

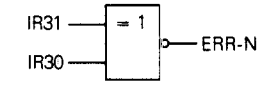
2



Expression of alternatives as part of a signal name.

RD/WR here stands for READ OR WRITE.

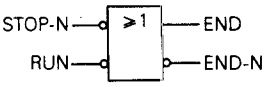
3



Labelling of a negated output of a combinative element in in-line notation.

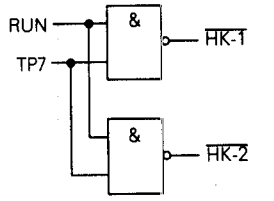
ERR may stand for ERROR and IR for INTERMEDIATE REGISTER.

4



Labelling of the complementary outputs of a combinative element in in-line notation.

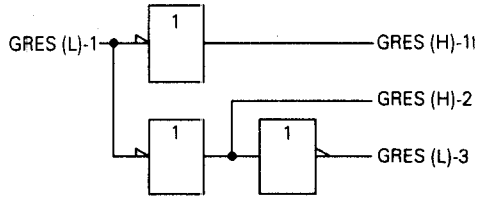
5



Use of serial numbers to identify two outputs carrying the same signal.

HK may stand for HOUSEKEEPING.

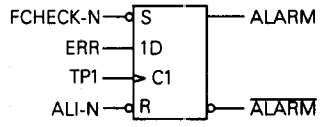
6



Use of serial numbers to identify the various trees of a signal being inverted and amplified.

GRES may stand for GENERAL RESET.

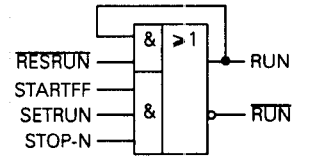
7



Labelling of the complementary outputs of a bistable element.

FCHECK may stand for FUNCTION CHECK and ALI for ALARM INHIBIT.

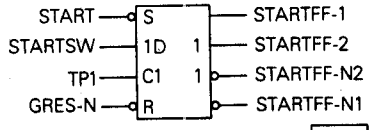
8



Labelling of the complementary outputs of a bistable element which is wired up from an AND-OR-NOT element.

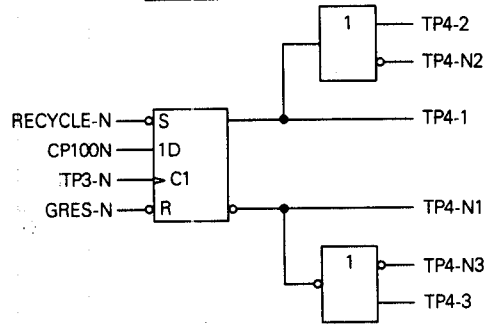
RESRUN may stand for RESET RUN.

9



Labelling of the complementary outputs of both the master and slave stages of a bistable element when attention should be drawn to the fact that these signals originate at a bistable element.

10



Extensive use of serial numbers to identify the various trees of a signal originating at a bistable element.

12. **Use of waveforms**

Waveforms may be shown to facilitate maintenance, adjustment of the equipment or to clarify the function of a circuit.

13. **Tabular information**

A logic diagram may be supplemented by:

a) truth tables;

b) tables containing information on components and packaged elements used to implement functions;

c) tables providing information on signal source, destination, etc.

14. **Representation of a pin connection acting at different times as either an input or an output**

Some devices have pins that can serve as inputs and outputs. Simplification of a logic diagram can result if such a connection is shown as both an input and an output on the symbol outline.

Examples given in IEC Publication 617-12 show how this should be done.

15. **Interaction of logic elements and other devices**

It is sometimes necessary to represent a logic element which operates a device such as a lamp or relay, or which is itself operated by such a device. Detailed logic diagrams usually contain information such as voltage levels which enables the reader to deduce the condition under which the required action will occur and no special notation is necessary. On diagrams dealing only with logic states, or in any case where confusion can arise, the logic state or logic level causing the desired operation may be indicated on the connection line adjacent to the symbol for the device. The use of the symbol for logic negation or logic polarity at terminals of other than logic elements is not allowed.

16. Exemples de logigrammes

Les figures 1 à 3, pages 27 à 29, illustrent la mise en œuvre de la présente norme. Afin de faciliter la comparaison entre les différents types de logigrammes, chacune de ces figures reprend le même élément d'un équipement, en l'espèce un générateur d'impulsions temporelles. Une explication des dénominations des signaux utilisés dans ces figures devrait normalement être fournie par une documentation afférente.

Dans la figure 1, des symboles d'opérateurs binaires sont utilisés pour indiquer les conditions de démarrage et d'arrêt de l'oscillateur. Aucune relation avec une réalisation matérielle précise n'est impliquée. De même, les niveaux logiques convenables et réalisés pour chaque signal peuvent être différents de ceux qui sont figurés. Un symbole fonctionnel est utilisé pour le changeur de fréquence.

La figure 2 est un exemple de logigramme qui utilise une logique positive. Ce type de logique, qui est indiqué par une note du schéma, établit la relation entre les niveaux logiques et les états logiques, ce qui fait que les fonctions logiques et les grandeurs physiques correspondantes sont fournies par le schéma.

Tous les détails, sauf l'alimentation des opérateurs logiques binaires, sont indiqués sur le schéma.

La figure 3 est un autre exemple de logigramme qui utilise le symbole de polarité logique. Les valeurs assignées aux niveaux logiques sont précisées par une note du schéma.

16. Examples of logic diagrams

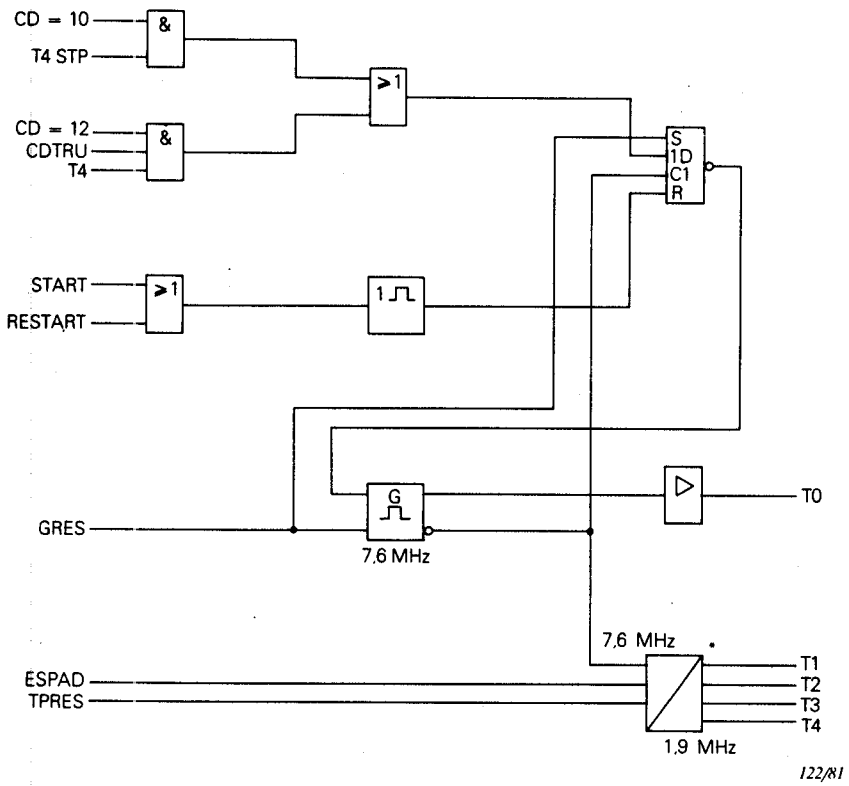
Figures 1 to 3, pages 27 to 29, illustrate the application of the principles given in this standard. To facilitate comparison between the different types of logic diagram, the same part of an equipment, i.e. a timing pulse generator, is shown in each of the figures. An explanation of the signal line names used in these figures would normally appear in supporting documentation.

In Figure 1 symbols for binary logic elements are used to show the conditions which start and stop the oscillator. In this example no relation to the exact physical implementation exists. Also, the actual logic levels available and produced for each signal may be different from those shown. A block symbol is used for the frequency divider (changer).

Figure 2 is an example of a logic diagram using the positive convention. This convention which is stated in a note on the diagram establishes the relationship between logic levels and logic states so that both the logic function and the physical function are represented by the diagram.

Every detail, except the power supply to the binary logic elements, is shown in the diagram.

Figure 3 is an alternative example of a logic diagram using the symbol for logic polarity. The exact logic levels are stated in a note on the diagram.



* Le diviseur de fréquence est en combinaison avec un circuit déphaseur. Pour plus de détails, voir le diagramme de séquence-temps.

* The frequency divider is combined with a phase shifter. For details see the time sequence chart.

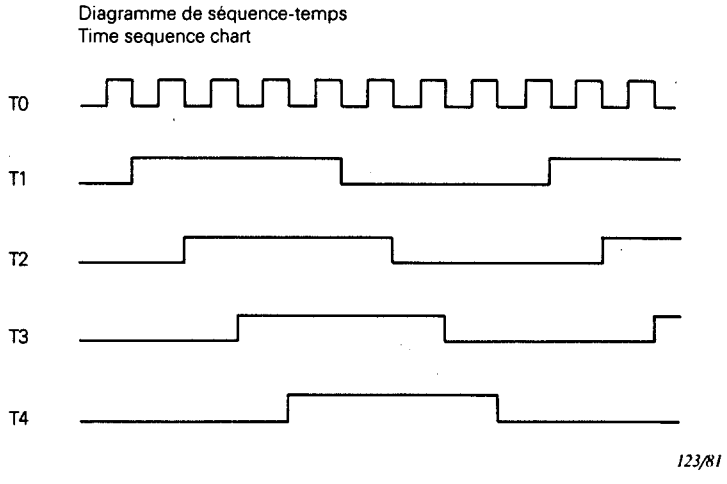
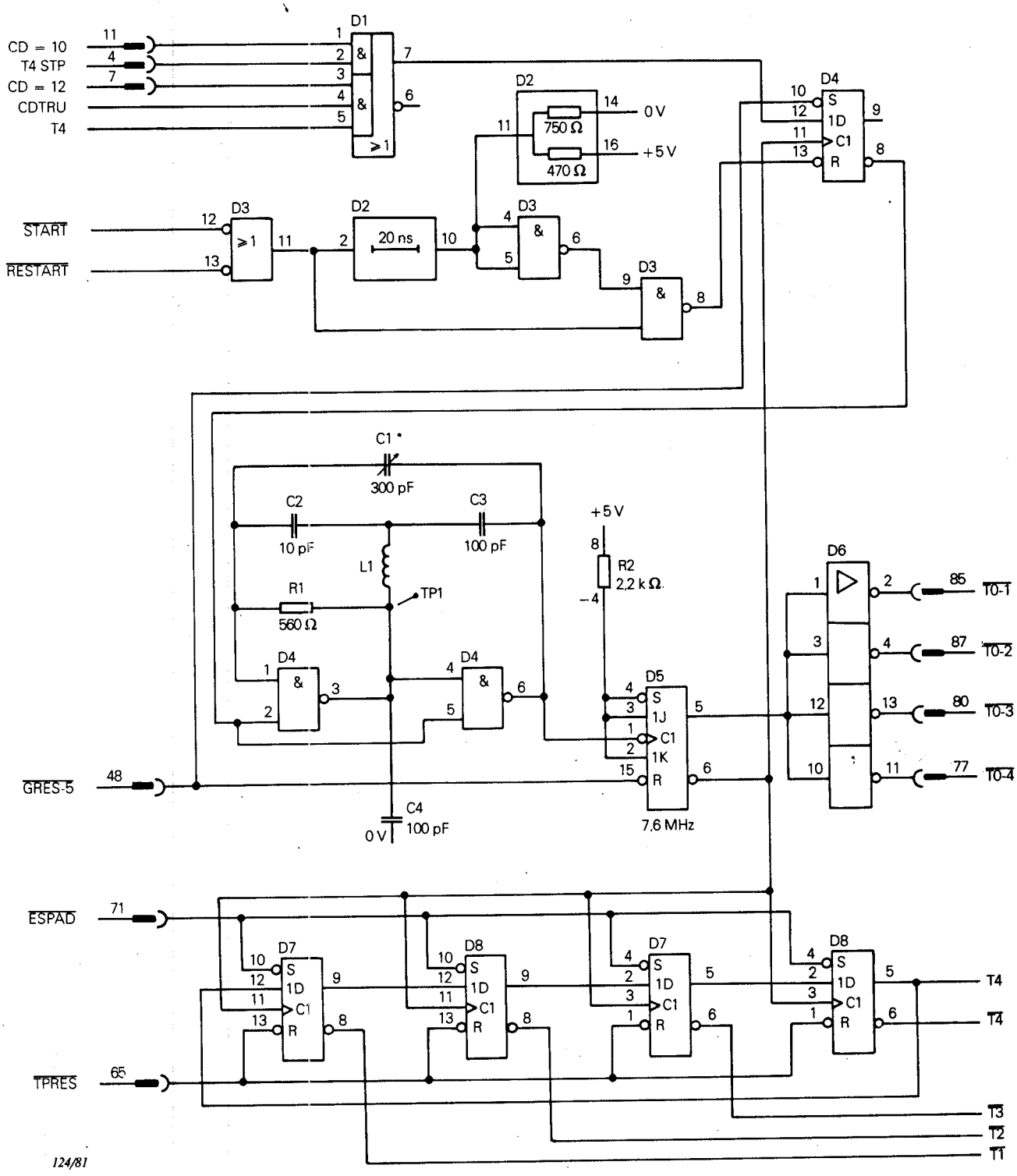


FIG. 1. — Logigramme d'un générateur d'impulsion d'horloge.
Logic diagram for a timing-pulse generator.



Notes. — « H » ⩾ 2,4 V, « L » ⩽ 0,5 V.

* Ajuster pour obtenir une fréquence de 15,20 MHz, la mesure devant se faire au point de contrôle TP1.

Notes. — "H" ⩾ 2.4 V, "L" ⩽ 0.5 V.

* Adjust for 15.20 MHz to be measured at test point TP1.

FIG. 3. — Logigramme détaillé utilisant le symbole de polarité logique.
Detailed logic diagram, using logic polarity indicator.

ANNEXE A

ABRÉVIATIONS MNÉMONIQUES COURANTES

Aucun ensemble de règles ne peut éviter au concepteur d'avoir à déterminer judicieusement les désignations des signaux et à l'utilisateur de bien savoir les interpréter.

Le tableau ci-après des abréviations mnémoniques courantes, essentiellement non exhaustif, est une tentative d'unification des principales désignations de signaux pour des systèmes logiques.

D'autres désignations peuvent être attribuées en accord avec les prescriptions de la présente norme, et d'autres significations peuvent être attribuées aux éléments de ce tableau, si aucune ambiguïté ne peut en résulter.

Les exemples donnés par le tableau ci-après correspondent à des expressions professionnelles typiques en langue anglaise et ne peuvent pas toujours être traduites sans risque de confusion (voir paragraphe 11.2.4 sur le maintien de la langue d'origine).

APPENDIX A

COMMONLY USED MNEMONICS

No set of rules can avoid the necessity for the designer to exercise good judgement and for the user to know how to interpret the significance of signal names.

The following table of commonly used mnemonics, which necessarily cannot be exhaustive, is an effort to unify some of the main signal names of logic systems.

New signal names may be assigned according to the rules set out in this standard and other meanings may be assigned to the ones listed thereafter, if no ambiguity results.

The examples given in the following table represent typical usage in the English language and cannot all be translated without risk of confusion (see Sub-clause 11.2.4 concerning language requirements).

Symbole Symbol	Légende Legend	Description
ACC	ACCEPTATION	ACCEPT
ACK	accusé de réception	ACKNOWLEDGE
ACT	ACTIVATION	ACTIVATE
ADR	ADRESSE	ADDRESS
ALI	inhibition d'alarme	ALARM INHIBIT
ALU	opérateur arithmétique	ARITHMETIC LOGIC UNIT
BP	signal binaire présent	BIT PRESENT
BCD	DÉCIMAL CODE BINAIRE	BINARY CODED DECIMAL
BCTR	compteur d'éléments binaires (BIT)	BIT COUNTER
BIN	BINAIRE	BINARY
BR	BRANCHEMENT	BRANCH
BUF	tampon	BUFFER
BUS	BUS	BUS
BUSY	occupé	BUSY
BY	multiplet	BYTE
BYSEL	sélection de multiplet	BYTE SELECT
CAR	report; retenue	CARRY
CARP	précalcul de report	CARRY PREDICT
CC	signal d'état	CONDITION CODE
CE	validation de circuit	CHIP ENABLE
CHK	vérification; contrôle	CHECK
CL	horloge	CLOCK
CLA	anticipation de retenue	CARRY LOOK AHEAD

Symbole Symbol	Légende Legend	Description
CLR	effacer	CLEAR
CMD	COMMANDE; instruction	COMMAND
COMP	COMPARAISON	COMPARE
CORR	CORRIGÉ	CORRECTED
CP	impulsion d'horloge; signal d'horloge	CLOCK PULSE
CR	registre de commande	CONTROL REGISTER
CRC	CONTRÔLE DE REDONDANCE CYCLIQUE	CYCLE REDUNDANCY CHECK
CS	mémoire de commande	CONTROL STORE
CT	COMPTAGE	COUNT
CTR	COMPTEUR	COUNTER
CY	CYCLE	CYCLE
D	DONNÉES	DATA
DACC	DONNÉES ACCEPTÉES	DATA ACCEPTED
DBY	multiplet de données	DATA BYTE
DD	double retard	DOUBLE DELAY
DEC	DÉCIMAL	DECIMAL
DEL	retard	DELAY
DEV	dispositif	DEVICE
DEVCLR	dispositif vide	DEVICE CLEAR
DIN	données entrées	DATA IN
DOUT	données sorties	DATA OUT
DR	registre de données	DATA REGISTER
DVLD	validation des données	DATA VALID
DWN	immobilisation; bas	DOWN
ECR	registre de contrôle d'erreur	ERROR CONTROL REGISTER
EN	validation	ENABLE
END	fin	END
EO	opération élémentaire	ELEMENTARY OPERATION
ERASE	oblitérer	ERASE
ERR	ERREUR	ERROR
EW	mot d'erreur	ERROR WORD
EXOR	disjonction; ou exclusif	EXCLUSIVE OR
F	FONCTION	FUNCTION
FB	premier bit	FIRST BIT
FBY	premier multiplet	FIRST BYTE
FF	bascule bistable	FLIP-FLOP
FIFO	premier entré — premier sorti	FIRST IN — FIRST OUT
FM	mémoire rapide	FAST MEMORY
FSEL	choix de fonction	FUNCTION SELECT
G	porte	GATE
GEN	GÉNÉRATION	GENERATE
GM	mémoire principale	GENERAL MEMORY
GND	terre	GROUND
GOON	poursuite	GO ON
GRES	remise à zéro générale	GENERAL RESET
H	maintien	HOLDING
HERR	erreur de frappe	HAMMING ERROR
HEX	HEXADÉCIMAL	HEXADECIMAL
HK	gardiennage	HOUSEKEEPING
HO	de poids fort; de rang supérieur	HIGH ORDER
HR	registre de maintien	HOLDING REGISTER
ICAR	report initial; retenue initiale	INITIAL CARRY
ID	IDENTIFICATION	IDENTIFICATION
INH	INHIBITION	INHIBIT
INOP	non exploitable	INOPERABLE
INT	INTERRUPTION	INTERRUPT
I/O	entrées-sorties	INPUT/OUTPUT
IR	registre intermédiaire	INTERMEDIATE REGISTER

Symbole Symbol	Légende Legend	Description
KT	CONTACT	CONTACT
L	à gauche	LEFT
LD	chargement; charger	LOAD
LO	de poids faible; de rang inférieur	LOW ORDER
LOC	emplacement	LOCATION
LOG1	UN LOGIQUE	LOGICAL ONE
LOGZ	ZÉRO LOGIQUE	LOGICAL ZERO
LP	impulsion locale	LOCAL PULSE
LRC	contrôle par redondance longitudinale	LONGITUDINAL REDUNDANCY CHECK
LS	mémoire locale	LOCAL STORAGE
LSB	bit de poids le plus faible	LEAST SIGNIFICANT BIT
LT	voyant	LIGHT
MAR	registre d'adresses mémorisées	MEMORY ADDRESS REGISTER
MCR	registre de microcode	MICROCODE REGISTER
MM	mémoire principale	MAIN MEMORY
MMPE	erreur de parité en mémoire principale	MAIN MEMORY PARITY ERROR
MOD	MODIFICATEUR	MODIFIER
MOT	MOTEUR	MOTOR
MPX	MULTIPLEXAGE	MULTIPLEX
MR	registre mémoire	MEMORY REGISTER
MSB	bit de poids le plus élevé	MOST SIGNIFICANT BIT
MUX	MULTIPLEXEUR	MULTIPLEXER
N	NÉGATION	NEGATION
NC	NORMALEMENT CLOS	NORMALLY CLOSED
NO	NORMALEMENT OUVERT	NORMALLY OPEN
OCT	OCTAL	OCTAL
OP	OPÉRATION	OPERATION
OPER	exploitable	OPERABLE
OPR	registre d'opération	OPERATION REGISTER
P	PRÉ-(antériorité)	PRE-
PAR	PARITÉ	PARITY
PC	compteur du programme	PROGRAM COUNTER
PCI	interruption programmée	PROGRAM CONTROLLED INTERRUPT
PE	erreur de parité	PARITY ERROR
PF	pré-extraction	PRE-FETCH
PON	sous tension	POWER ON
PPON	processeur sous tension	PROCESSOR POWER ON
PS	état du programme	PROGRAM STATUS
PU	extraction	PULL-UP
R	à droite	RIGHT
RD	lecture	READ
RDY	prêt	READY
RE	RE-(répétition)	RE-
REG	REGISTRE	REGISTER
REJ	REJET	REJECT
REQ	interrogation; demande	REQUEST
RES	mise à zéro; remise à l'état initial	RESET
RGM	matrice de régénération	REGENERATION MATRIX
RO	sortie de lecture	READ OUT
ROM	mémoire morte	READ ONLY MEMORY
RUN	marche	RUN
RX	RÉCEPTION	RECEIVE
RZ	RÉSULTAT NUL	RESULT ZERO
SCAR	retenue finale	SUM CARRY
SEL	SÉLECTION	SELECT
SELDEL	dispositif de sélection	SELECT DEVICE
SET	positionnement; mise à « 1 »	SET

Symbole Symbol	Légende Legend	Description
SH	décalage	SHIFT
SIM	SIMULATION	SIMULATION
SIMC	retenue simulée	SIMULATION CARRY
SRQ	demande de service	SERVICE REQUEST
START	mise en marche; début	START
STBY	multiplet d'état	STATUS BYTE
STOP	ARRÊT	STOP
STR	signal d'échantillonnage	STROBE
SW	aiguillage; commutation	SWITCH
SYNC	SYNCHRONISATION	SYNCHRONIZATION
TERM	fin	TERMINATE
TG	bascule bistable	TOGGLE
TO	transfert	TO (transfer)
TP	impulsion temporelle	TIME PULSE
TRAC	commande de transmission	TRANSMIT CONTROL
TRIG	déclencheur	TRIGGER
TX	transmission	TRANSMISSION
U	en haut	UP
UR	registre de service	UTILITY REGISTER
VAR	registre d'adresses virtuelles	VIRTUAL ADDRESS REGISTER
WC	commande d'écriture	WRITE CONTROL
WI	entrée d'écriture	WRITE IN
WR	écriture	WRITE
WRAP	balayage; bouclage	WRAP AROUND

APPENDIX B

GUIDE TO THE MEANING, INTERPRETATION AND SELECTION OF LOGIC SYMBOLS

B1. Object

This appendix gives an explanation of the meaning and interpretation of logic symbols.

B2. Logic states and logic levels

Binary logic is concerned with variables, each of which may take up one of two states. These states may be described by terms such as ON and OFF, YES and NO, TRUE and FALSE or more usually as the logic 1-state and the 0-state.

The qualifying symbols for logic functions in IEC Publication 617-12 represent the relationship between the inputs and outputs in terms of *logic states*. They may therefore be used on diagrams representing abstract logic designs and no difficulties of interpretation should arise. At this stage there need be no knowledge of what type of device (electrical, fluidic, etc.) will eventually be used to implement the design.

As soon as hardware items are selected to carry out the desired logic functions it becomes necessary to decide on the physical quantities which will be used to represent the logic states. With electrical devices it is usual to select electric potential as the physical quantity and to assign values to represent the logic states (for other possibilities see Sub-clause 2.1). Absolute figures are not normally used and the values are merely identified as the more positive (High H) or the less positive (Low L) as the case may be. These two values are known as *logic levels*.

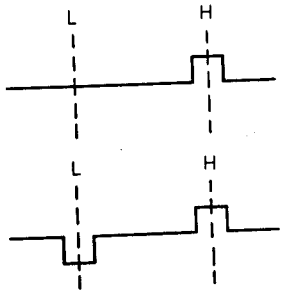
A clear understanding of the distinction between the concepts of *logic state* and *logic level* is vital to the correct interpretation and use of logic symbols and to the discussion which follows.

B2.1 *Assignment of other parameters of a physical quantity to denote H- and L-logic levels*

Parameters other than steady values of a physical quantity may be used to represent logic states and these parameters may be denoted by H and L. The following examples indicate some of the possibilities and show how the correspondence between the desired parameters and the H- and L-logic levels may be depicted.

a) The logic H-level corresponds with the presence of a positive pulse at a specific time.
The logic L-level corresponds with the absence of a pulse at a specific time.

b) The logic H-level corresponds with the presence of a positive pulse at a specific time.
The logic L-level corresponds with the presence of a negative pulse at a specific time.



B3. Correspondence between logic states and logic levels

When logic symbols are used to represent physical devices it is necessary to establish the correspondence between logic states and the values (logic levels) of the physical quantities used to represent these states. There are basically two methods by which this may be done.

The first is the adoption of a single logic convention, either positive or negative, for the whole diagram (see Sub-clause B3.1).

The second consists of the use of the logic polarity indicator, the presence or absence of which indicates the required correspondence at each input and output of every logic symbol on the diagram (indication by the logic polarity indicator, see Sub-clause B3.2).

B3.1 *Single logic convention*

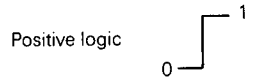
With this method the correspondence between a given logic state and logic level is the same at all inputs and outputs on the diagram.

The symbol for logic negation is used as required, *but the logic polarity indicator must not be used.*

The convention in use, either positive or negative (see Sub-clauses B3.1.1 and B3.1.2), must be clearly stated on the diagram or in supporting documentation. This statement can usefully include a small waveform diagram.

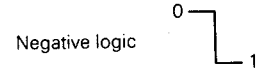
B3.1.1 *Positive logic convention*

The more positive value of the physical quantity (H-level) corresponds to the logic 1-state. The less positive value (L-level) corresponds with the logic 0-state. This may be stated on a diagram thus:



B3.1.2 *Negative logic convention*

The less positive value of the physical quantity (L-level) corresponds to the logic 1-state. The more positive value (H-level) corresponds to the logic 0-state. This may be stated on a diagram thus:



B3.2 *Indication by logic polarity indicator*

With this method the presence of the logic polarity indicator at the specific input or output signifies that the L-level of the physical quantity corresponds with the logic 1-state. Absence of the indicator at a specific input or output indicates that the H-level corresponds to the 1-state at that point. *In this case the logic negation indicator must not be used.*

It will be seen that these two statements correspond respectively with the use of negative logic (with the symbol present) and positive logic (when the symbol is absent). For this reason this method is sometimes described as the use of mixed logic, but, because this term can be confusing, its use is not recommended.

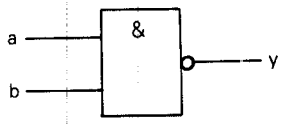
B4. Interpretation of logic symbols

For explanatory purposes it is convenient to employ the term "interconnection" to describe a connecting line between the device symbols on a diagram. Qualifying symbols for logic negation and logic polarity are associated only with symbols for logic elements and are not part of the interconnection.

Sub-clauses B4.1 to B4.3 show how various symbols representing hardware devices may be interpreted on diagrams employing the positive logic convention, the negative logic convention and the logic polarity indicator, respectively.

B4.1 *Examples in positive logic*

B4.1.1 *AND function*



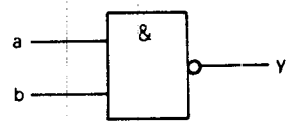
Logic states

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

Logic levels

a	b	c
L	L	L
L	H	L
H	L	L
H	H	H

B4.1.2 *NAND function*



Logic states

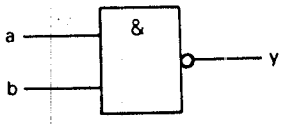
a	b	c
0	0	1
0	1	1
1	0	1
1	1	0

Logic levels

a	b	c
L	L	H
L	H	H
H	L	H
H	H	L

B4.2 *Examples in negative logic*

B4.2.1 *AND function*



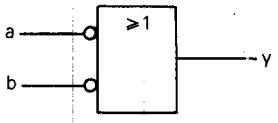
Logic states

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

Logic levels

a	b	c
H	H	H
H	L	H
L	H	H
L	L	L

B4.2.2 *OR function*



Logic states

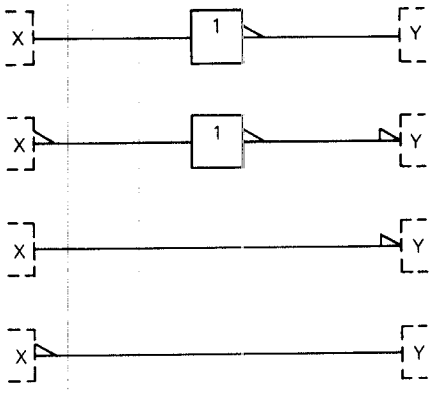
a	b	c
0	0	0
0	1	1
1	0	1
1	1	1

Logic levels

a	b	c
H	H	H
H	L	L
L	H	L
L	L	L

B4.3 *Examples using the logic polarity indicator*

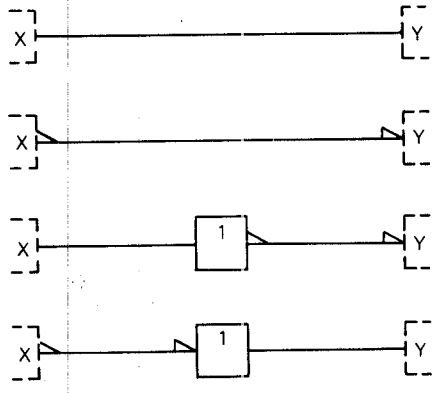
It is important to realize that the presence or absence of the polarity indicator serves only to define which logic convention is in force at the input or output of the basic symbol and that it does not indicate a reversal of logic state. It follows that the logical NOT function, being an inversion of logic state, cannot be represented using the symbol for logic polarity. However, there are four possible ways of depicting an interconnection implying logic negation as shown in the following examples. The elements at each end of the interconnection dictate the way which must be used.



Logic states

X	Y
0	1
1	0

The following figures show the four possible ways of depicting an interconnection without implying logic negation.



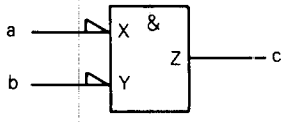
Logic states

X	Y
0	0
1	1

Similarly, the NAND and NOR functions cannot be represented using the polarity indicator. Comparison of the example in Sub-clause B4.1.2 with the example in Sub-clause B4.3.4 shows that although the tables of logic levels are identical, the logic states of the outputs of the elements are different.

When elements have different logic conventions at inputs and outputs it is necessary and sufficient to define such elements in terms of the levels on the interconnections. In the following examples the tables of logic states have been included to indicate the development of the tables of logic levels.

B4.3.1 *AND, inputs active LOW*



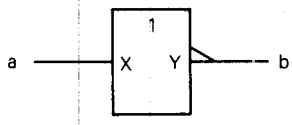
Logic states

X	Y	Z
0	0	0
0	1	0
1	0	0
1	1	1

Logic levels

a	b	c
H	H	L
H	L	L
L	H	L
L	L	H

B4.3.2 *Invertor, output active LOW*



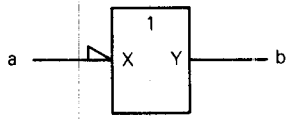
Logic states

X	Y
0	0
1	1

Logic levels

a	b
L	H
H	L

B4.3.3 *Invertor, input active LOW*



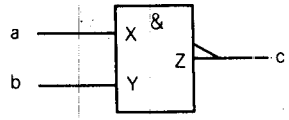
Logic states

X	Y
0	0
1	1

Logic levels

a	b
H	L
L	H

B4.3.4 *AND, output active LOW*



Logic states

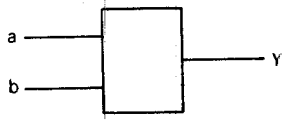
X	Y	Z
0	0	0
0	1	0
1	0	0
1	1	1

Logic levels

a	b	c
L	L	H
L	H	H
H	L	H
H	H	L

B5. Dual functions

Consider a simple binary device having two inputs and one output, the logic level of the output depending on the combination of the logic levels of the inputs in accordance with the table:



Logic levels

a	b	y
L	L	L
L	H	L
H	L	L
H	H	H

The behaviour of this device can be described in either of two ways:

1. The output y will stand at the H-level if and only if input a *and* input b stand at the H-level.
2. The output y will stand at the L-level if input a *or* input b stands at the H-level or if both inputs stand at the L-level.

These statements are both true and each on its own is sufficient to describe the function of the device.

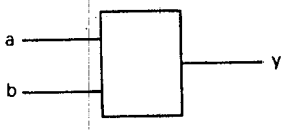
If the H-level at both the inputs and the outputs is chosen to represent the logic 1-state, the device is seen to carry out the AND-function in accordance with statement 1 above.

If the L-level at both the inputs and the outputs is chosen to represent the logic 1-state, the device is seen to carry out the OR-function in accordance with statement 2 above.

It is evident from the foregoing that the function performed by a given binary device depends on the way it is used or more specifically on the significance assigned to the logic levels at its inputs and outputs, i.e. on the convention in force.

B6. Choice of symbols

Complementary representation of functions may be used to simplify the understanding of logic diagrams. Consider for example the symbolization of a hardware element characterized by the following table:



Logic levels

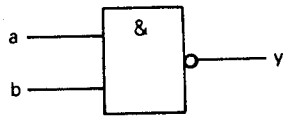
a	b	y
L	L	H
L	H	H
H	L	H
H	H	L

On a diagram drawn in positive logic the following table may be drawn:

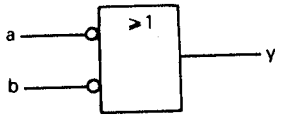
Logic states

a	b	y
0	0	1
0	1	1
1	0	1
1	1	0

The last line of this table summarizes the statement of the AND with negated output (NAND) function and the device may be symbolized thus:



The first three lines of the table, however, indicate that the output will stand at its 1-state if either input a *or* b stands at its 0-state. The symbol for logic negation may be applied to the symbol for an OR function to emphasize this relationship thus:



The choice between these two symbols should be governed by ease of interpretation of the diagram. If it is important for the reader to see what happens to the output if either input signal is in its 0-state, then the second symbol is easier to interpret than the first.

B7. Examples of logic representation

To illustrate the development of logic representation the *exclusive OR* function will be considered.

In a practical diagram the exclusive OR function would normally be represented by a single symbol. The function has been used here as a simple illustration of the principles of logic representation.

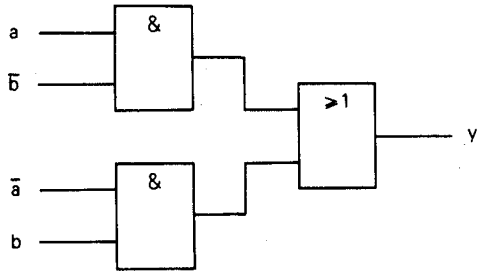
The table and logic equation for this function are as follows:

Logic states

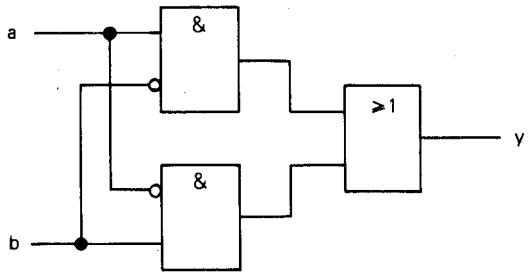
a	b	y
0	0	0
0	1	1
1	0	1
1	1	0

$y = a\bar{b} + \bar{a}b$

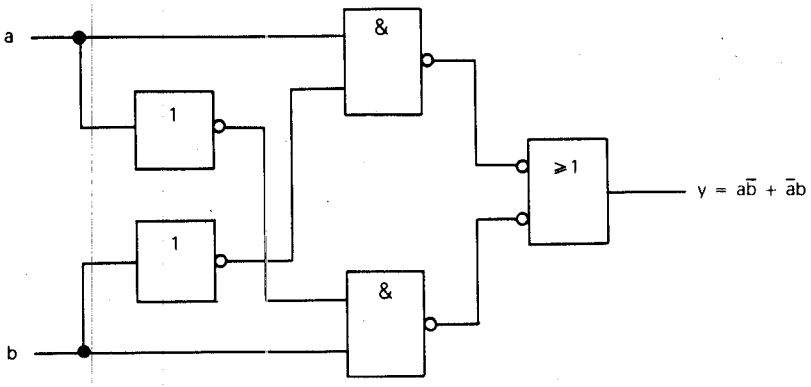
From this equation the required function can be expressed in basic symbols:



The symbol for logic negation can be used to simplify the input connections:



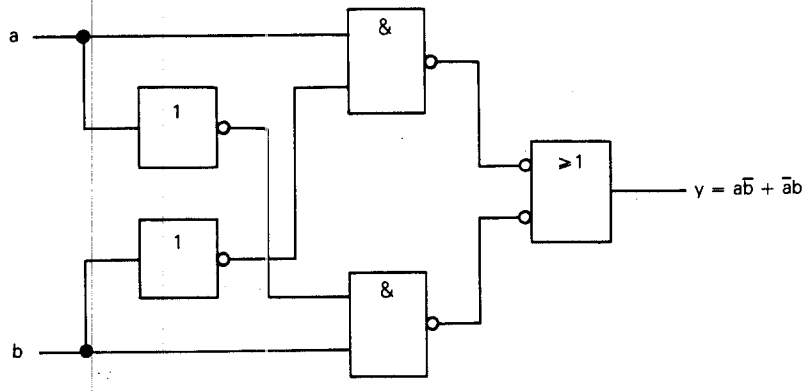
The diagram above is a pure logic diagram and it will be seen that each AND element has a negated and a non-negated input. Such an element is not normally available as a standard hardware item. If the diagram is to be developed further to represent physical levels and hardware then the following expansion can be made using the positive logic convention:



Logic levels

a	b	y
L	L	L
L	H	H
H	L	H
H	H	L

The same function can be represented in negative logic by an identical circuit as follows:

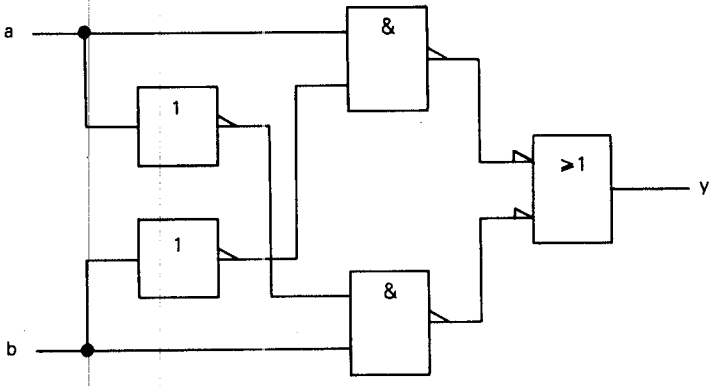


Logic levels

a	b	y
H	H	H
H	L	L
L	H	L
L	L	H

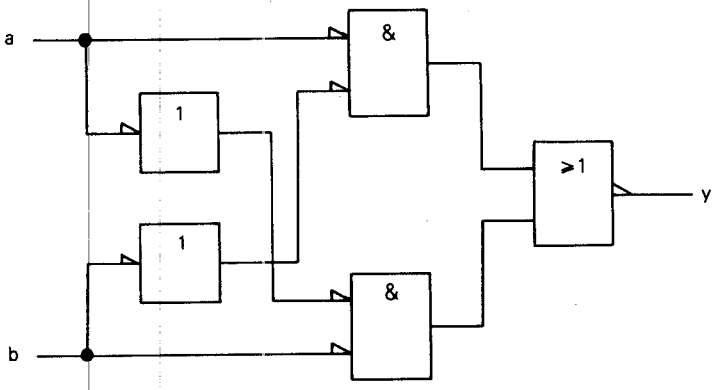
From a comparison of these tables it will be noted that different hardware would be required for the above two cases.

The two examples above may be redrawn using the polarity indicator as follows:



Logic levels

a	b	y
L	L	L
L	H	H
H	L	H
H	H	L



Logic levels

a	b	y
H	H	H
H	L	L
L	H	L
L	L	H

In this case no boolean expression should appear at the outputs of the circuits, because in a system using the polarity indicator, logic states do not exist there, but only inside the logic elements.