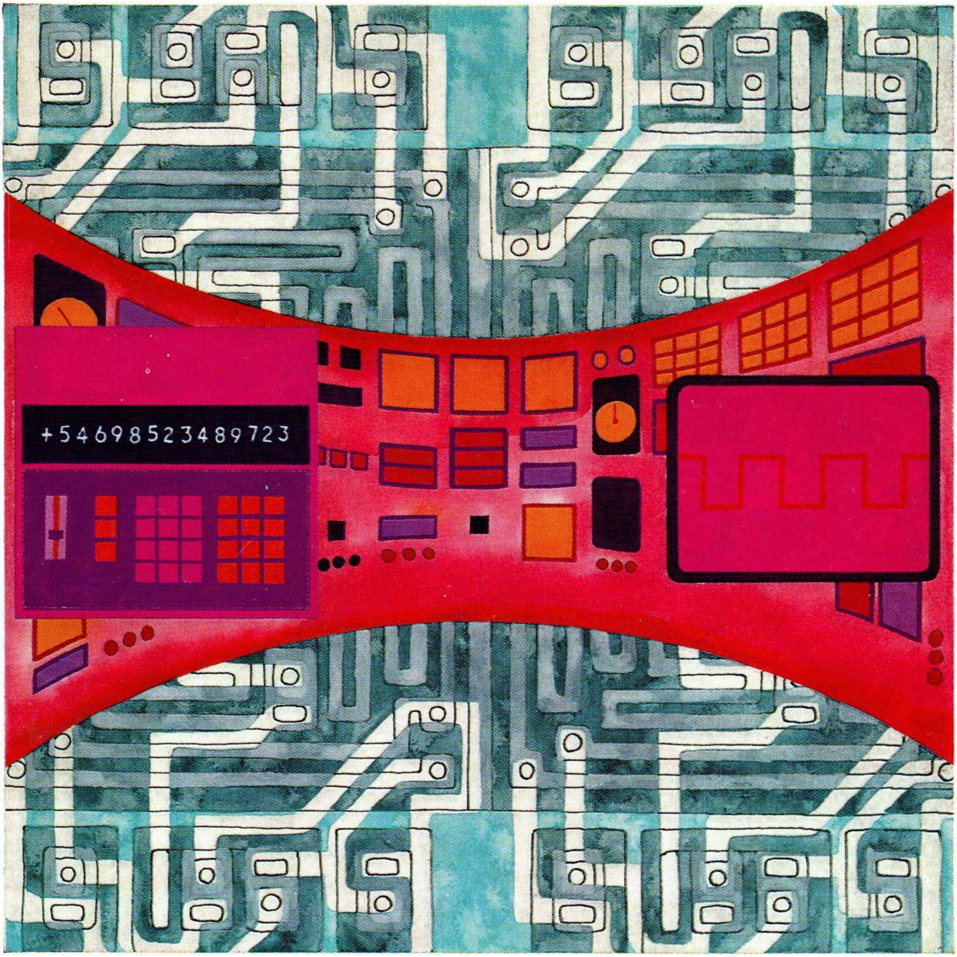
PHILIPS

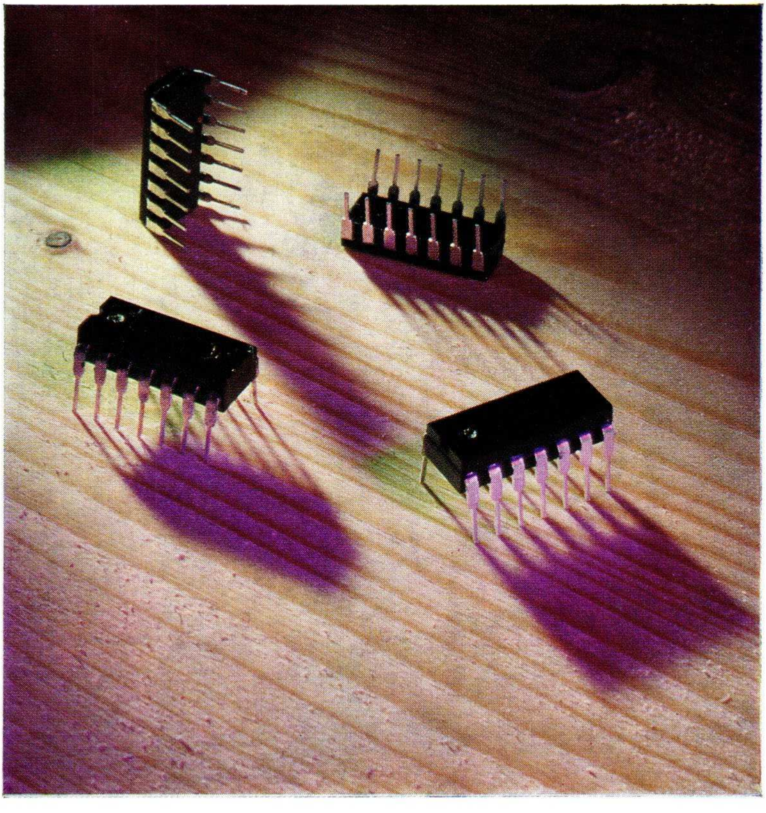
APPLICATION BOOK

ELECTRONIC COMPONENTS
AND MATERIALS DIVISION

FC FAMILY OF DTL INTEGRATED CIRCUITS



FC Family of DTL Integrated Circuits



FC Family
# of DTL Integrated Circuits

edited by **J. Deerson**

PUBLICATIONS DEPARTMENT
ELECTRONIC COMPONENTS AND MATERIALS DIVISION

EINDHOVEN - The Netherlands

April 1970

The publication of this document does not imply a licence under any patent

Contents

Preface

The purpose of this book is to summarize, in clear and easily accessible form, the information that system design engineers need to know in order to make the best possible use of the FC family of DTL integrated circuits.

By way of introduction it describes the circuits comprising the range and compares their performance with that of other kinds of logic circuitry. Emphasizing practical design methods, it describes in detail a number of typical and widely used applications that exploit the advantages of the family while taking proper account of its limitations. Besides that, it offers such general application information as rules for designing sequential and combinational logic systems, and rules to ensure satisfactory mechanical design.

It is hoped that by consolidating information in convenient format this book will help to enlarge the already wide acceptance and use of the FC family of integrated circuits.

Acknowledgement

For their valuable contributions and assistance in preparing the book acknowledgement is due to:

W. M. Adriaans	Eindhoven
E. J. van Barneveld	Nijmegen
A. J. M. Dorn	Eindhoven
P. A. Neeteson	Eindhoven
M. P. Smit	Eindhoven
Ph. de Weger	Nijmegen

1 The FC Family of DTL Digital Integrated Circuits

1.1 Introduction

Several ranges of integrated circuits are available to the digital applications field. The main difference between these ranges is in their operating speed, but they also differ in logic levels, power dissipation, supply voltage and other respects.

Many applications require devices which operate with propagation delays of the order of 5 to 50 ns. These medium-speed devices are used in small and medium-sized computers, peripheral equipment, telephone exchanges, instrumentation and control, and in many items of digital equipment associated with telecommunications.

In this speed range two configurations are common:

DTL (Diode-Transistor-Logic), represented by the FC series, and

TTL (Transistor-Transistor-Logic), represented by the FJ and FH series.

Of these, DTL is currently the more familiar and has become established as a solution to many problems.

1.2 General Description of FC Family

1.2.1 IMPORTANT FACETS

Integrated circuit techniques allow a number of transistors, diodes, resistors and capacitors to be made on a single chip of silicon. The various components are connected to form the required circuit. The DTL gate of Fig. 1–1*a* forms the basis of the complete FC family of digital integrated circuits.

This gate performs the NAND-function for positive logic*. The left-hand part of the circuit acts as an AND gate: only if all inputs G_1, G_2, G_3 have a HIGH voltage will point X have a HIGH voltage. If one or more inputs are LOW there will flow current through R_1 and the voltage at X will be LOW. This implies that the circuit functions as an OR gate for negative logic. The right-hand part of the circuit, involving D_4, D_5, TR, R_C

* Positive logic has the logic 1-level at a more positive voltage than the logic 0-level. See Appendix "Explanation of Drawing Symbols and Notation".

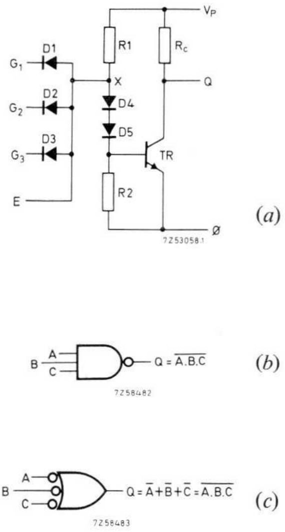


Fig. 1–1. The basic NAND gate.

and R_2, represents an inverter. As soon as the base-emitter voltage going HIGH reaches a certain level, transistor TR will conduct and output Q will go LOW.

The hardware shown in Fig. 1–1 can be represented by either of two logic symbols: that of Fig. 1–1*b* (AND) or that of Fig. 1–1*c* (OR). Although a more detailed description of the gate is given in chapter 2, a few observations in general will not be out of place here.

- The number of inputs can be easily and cheaply increased by means of external diodes connected at pount E (but not all gates are provided with this expander facility).
- The level shifting diodes D_4 and D_5 create an additional voltage "hurdle" to be exceeded before the transistor switches on, thus giving a high noise immunity for the DTL gate.
- By simply connecting the outputs of two or more DTL gates, an extra function will be obtained, usually called "Wired-Output". To be able to do this within wide limits, nearly all gates are available in two versions, with and without collector resistor R_C (see section 2.1.4).

On the basis of the NAND gate a variety of gate functions, bistable circuits and complex elements are available in the FC family. Table 1.1 shows the currently available circuits. Our Data Handbook System gives complete specifications of the various circuits.

Table 1.1(a) General Data of Gates

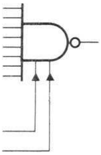
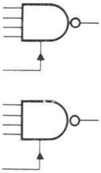
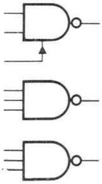
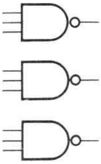
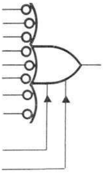
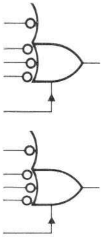
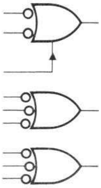
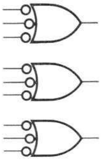
	gates, inverters and expanders with resistive pull-up		line drivers (active pull-up)		
	all types		type FCH221	type FCH222	type FCH231
operating temperature range	standard	extended	standard	extended	standard
propagation delay	31 ns	32 ns	35 ns	53 ns	35 ns
available fan-out	8	11	14	20	20
noise margin:					
typical	1.2 V	1.2 V	1.2 V	1.2 V	1.2 V
min. (T=25°C)	0.60 V	0.55 V	0.60 V	0.55 V	0.60 V
min. (T=$T_{max.}$)	0.40 V	0.25 V	0.40 V	0.25 V	0.40 V
power consumption*					
without R_C	7 mW	7 mW			
with R_C	11 mW	10.5 mW			
with active pull-up			11 mW	11 mW	11 mW

* per logic function.

Temperature range for military types:

operating	−55 to +125 °C
storage	−65 to +125 °C

Table 1.1(a) General Data of Gates.

description	single gate	dual gate	triple gate		
AND symbols					
OR symbols					
type number without R_C	FCH101 FCH102	FCH121 FCH122	FCH141 FCH142	FCH151 FCH152	
with R_C	FCH111 FCH112	FCH131 FCH132	FCH161 FCH162	FCH171 FCH172	
with active pull-up					

* Collector resistor R_C not applicable.

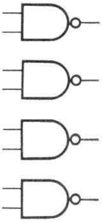
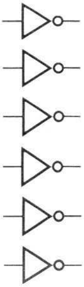
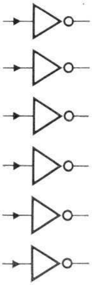
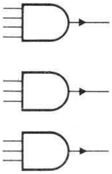
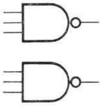
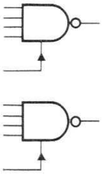
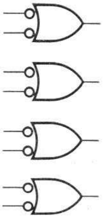
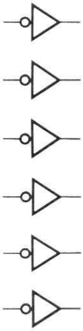
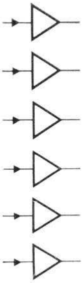
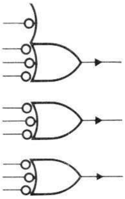
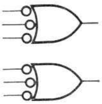
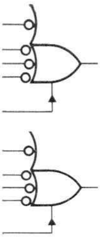
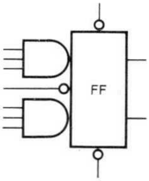
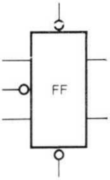
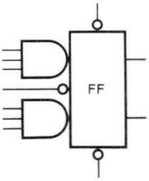
	quadruple gate	sextuple inverter	sextuple expandable inverter	triple input-expander	dual line-driver gate	
						
						
	FCH181	FCH201	FCH311			
	FCH182	FCH202		FCY101*		
	FCH191	FCH211	FCH321	FCY102*		
	FCH192	FCH212				
					FCH221	FCH231
					FCH222	

Table 1.1(b) General Data of Flip-Flops

description	single JK flip-flop		single JK master-slave flip-flop		
symbol					
type number	FCJ101	FCJ102	FCJ111	FCJ201	
available fan-out	8	10	8	8	
power consumption (per logic function, $T = 25\,^{\circ}\mathrm{C}$)	36 mW	55 mW	67 mW	67 mW	
maximum clock rate (typical)	10 mHz	10 MHz	5 MHz	5 MHz	

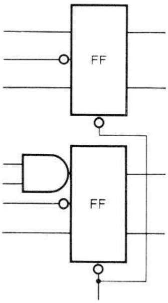
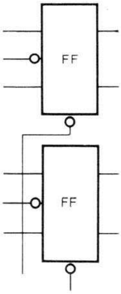
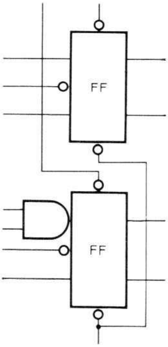
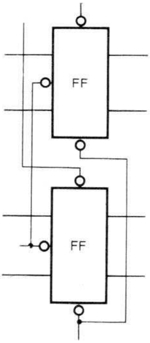
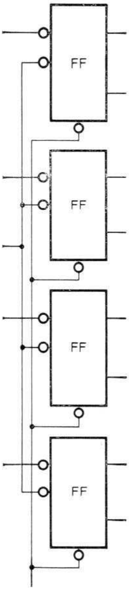
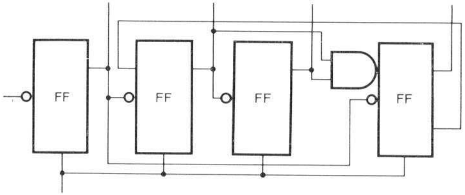
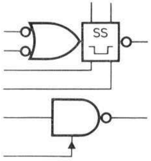
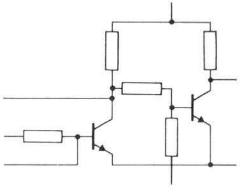
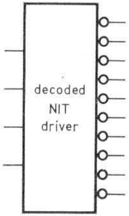
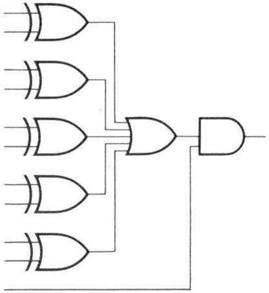
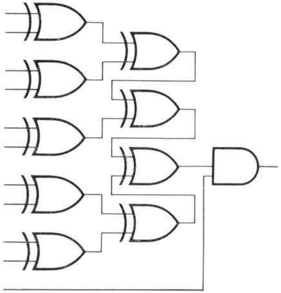
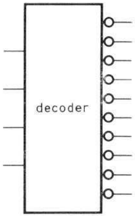
	dual JK master-slave flip-flop				quadruple latch flip-flop
					
	FCJ121	FCJ131	FCJ191	FCJ211	FCJ221
	8	8	8	8	8
	50 mW	50 mW	50 mW	50 mW	200 mW
	7 MHz	7 MHz	7 MHz	7 MHz	5 MHz

Table 1.1(c) General Data - Miscellaneous.

description	single asynchronous counter	monostable multivibrator		level detector (Schmitt trigger)	
					
type number	FCJ141	FCK101	FCK102	FCL101	FCL102
available fan-out	6	8	11	3	3
propagation delay					
power consumption	200 mW	110 mW	110 mW	12 mW	14 mW
maximum clock rate	5 MHz	2.5 MHz	2.5 MHz	3.0 MHz	1.0 MHz

description	single decoder N.I.T.* driver	single 5-bit comparator	single 10-bit parity checker	gated 4-bit decoder
				
type number	FCL111	FCH281	FCH291	FCH 301
available fan-out	one N.I.T.*	7	7	8
propagation delay		150 s	150 s	70 s
power consumption	190 mW	50 mW	50 mW	250 mW
maximum clock rate				

* N.I.T. = Numerical Indicator Tube.

1.2.2 SUMMARY OF MAJOR DATA

Table 1–2 gives a summary of the major data applicable to the FC family.

Table 1–2 Summary of Data for the FC Family

absolute maximum supply voltage:	8.0 V
operating supply voltage:	5.7–6.3 V
absolute maximum input voltage:	8.0 V
maximum input current requirements:	
LOW state	—1.65 mA/input
HIGH state	1 μA/input
FAN-out over full operating temperature range:	
for buffer gates	14 or 20
for other circuits	8
temperature range for industrial types:	
operating	0 to 75 °C
storage	—55 to + 125 °C

1.3 Manufacturing techniques

The process for the manufacture of monolithic integrated circuits is basically an extension of that which is used in making silicon planar transistors. On an N-type silicon wafer a thin oxide layer is grown. With a photo-lithographic process, windows are etched in the oxide on the places where transistors, diodes or resistors are to be made. Impurity diffusion into the silicon can only occur through the windows, thus giving the required p–n junctions. This process of oxidizing, etching and diffusing is repeated several times to obtain the required configuration of components with the required characteristics. After the last oxide etching step, an aluminium evaporation is done to connect the components to each other. The whole process of making the masks and aligning them exactly with the pattern on the slice left after the previous diffusion requires a high degree of workmanship. To illustrate the accuracy: every silicon slice gives after breaking several hundreds of integrated-circuit chips.

The size of a single IC-chip, containing large numbers of components, is in the region of 1 mm^2. Fig. 1–2 gives a cross section through part of a silicon chip showing an integrated n-p-n transistor, a resistor and a diode.

Components or groups of components must be insulated from each other. This is done by means of the epitaxial layer and the isolation diffusion. Prior to the photo-lithographic process an epitaxial N layer

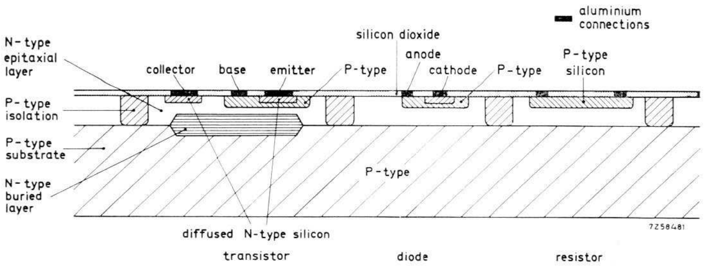


Fig. 1–2. Cross-section through part of a silicon chip.

is diffused over the entire P-type substrate. The first photo-lithographic step is the dividing of the epitaxial layer into "islands" by means of isolation diffusion. The P–type substrate is connected to the negative supply; thus reverse-biased p-n junctions isolate the islands from each other. The collector contacts must be brought out on the top of the slice. Compared with discrete transistors this leads to increased values of the collector resistance and saturation voltage of the transistor because of the lateral resistance of the epitaxial layer. To counteract this, a heavily doped (low-ohmic) layer, the *buried layer*, has been diffused into the substrate before epitaxy, to provide a lateral low-resistance path.

To insure proper low-resistance collector contacts, an N^+ collector contact diffusion is carried out. This N^+ diffusion can also be used in interconnections.

The base diffusion is a P diffusion into the collector (into an island of the epitaxial layer) whilst the emitter diffusion again is an N diffusion in the base region.

The sheet resistance of any diffused layer, depending on impurity concentration and diffusion depth, can be used to determine the geometry of required resistors. However, most resistors are made compatible with transistor base diffusion. Capacitances are obtained by using reverse-biased diodes.

After the diffusion procedures and the scribing and breaking of the slice, the separate IC chips are bonded to a header (die bonding), and the wires between the ends of the aluminium patterns of the chip and the header are connected (wire bonding).

Finally the encapsulation takes place, either in flat pack encapsulation (mainly for military types) or in the "dual-in-line" (D.I.L.) package.

TESTING AND QUALITY CONTROL

Very important aspects of the production of integrated circuits are the continual testing and controlling operations. This implies not only a 100 % electrical measurement and inspection of the circuit at several stages of their production, but also a close and continual control of workmanship and factory cleanliness. Highly sophisticated computerised measuring equipment and extensive climatic, environmental, and life test programmes insure the quality and reliability that has become standard in our FC family of digital integrated circuits.

1.4 Comparison of DTL and TTL Circuits

The integrated circuit user is often confronted with the task of deciding which of the two technologies he should use in a system, DTL or TTL. (A range of TTL integrated circuits is available; it is referred to by the type letters *FJ**.) Table 1–3 lists main characteristic data for each.

For comparison the circuits of a DTL and a TTL gate are given in Figs 1–3 and 1–4.

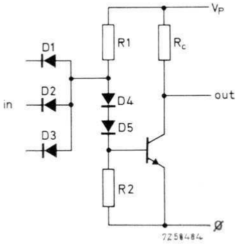


Fig. 1–3. The DTL NAND gate.

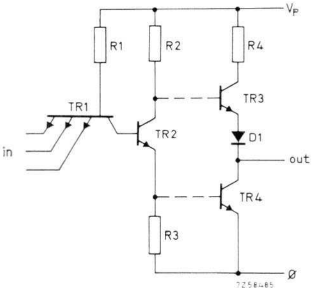


Fig. 1–4. The TTL NAND gate.

* See our Application Book "FJ Range of TTL Integrated Circuits".

Table 1–3 Comparison of Main Characteristics of DTL and TTL Gates

characteristic	DTL	TTL
fan-out, maximum	8	10
fan-in, maximum	8	8
input expansion facility	yes	no
propagation delay	25 to 30 ns	13 ns
signal noise immunity		
LOW state	0.8 V ($\geqslant$ 0.6 V)	0.8 V ($\geqslant$ 0.4 V)
HIGH state	3.0 V* ($\geqslant$ 1.8 V)	1.9 V** ($\geqslant$ 0.4 V)
power dissipation per gate	11 mW*	
(50 % duty cycle)	10 mW**	10 mW**
wired-output capability	yes	not normally
output drive		
LOW state	good	good
HIGH state	poor	fair
short-circuit protection	yes	yes
dual-in line package	yes	yes

* 6 V supply
** 5 V supply.

2 General Application Information

This chapter deals with general points of interest regarding the complete range of FC-family ICs. Sections 2.1, 2.2 and 2.3 (Gates, Flip-Flops and Miscellaneous circuits respectively) discuss particular aspects such as principles of operation and application hints.

2.1 Gates

2.1.1 PRINCIPLES OF OPERATION

A simple form of gate is the combination of diodes and resistor shown in Fig. 2–1. For positive logic, that is a HIGH voltage denoting logic "1" level, this gate performs an AND function. In Boolean algebra this can be expressed as:

$$G_1 \cdot G_2 \cdot G_3 = Q.$$

For negative logic, in which a LOW voltage denotes logic "1" level, this gate performs the OR function, expressed as:

$$G_1 + G_2 + G_3 = Q.$$

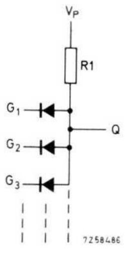


Fig. 2–1. The simplest form of gate, performing the logic function "AND".

Loading this gate with similar circuits would however have considerable influence on the operation, and fan-out would be low. In the circuit of Fig. 2–2 an (inverting) amplifier has been added; the function performed is NOT-AND or NAND, expressed as:

$$\overline{G_1 \cdot G_2 \cdot G_3} = Q.$$

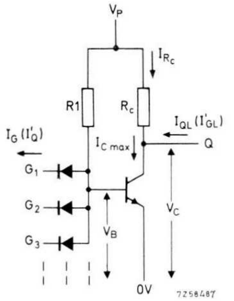


Fig. 2–2. A simple form of NAND gate.

By using a basic rule in Boolean algebra this can be written as:

$$\bar{G}_1 + \bar{G}_2 + \bar{G}_3 = Q,$$

which is clearly an OR function, demonstrating the dual nature of the logic gate. This is the basic form of the DTL (FC-family) gates, types FCH101 to FCH211 and types FCH102 to FCH212 (standard and extended temperature ranges respectively). Fan-out of these types is 8; by using two additional transistors in an "active pull-up" arrangement, fan-out can again be appreciably increased. This has been done in the case of the "Line Driver" types FCH221 and FCH231 (standard temperature range) and type FCH222 (extended temperature range) with fan-outs of 14,20 and 20 respectively. Further information about these types is given in section 2.2.2; note that a "Wired-Output" connection must not be used with gates having active pull-up.

Assume that the diodes and transistor in Fig. 2–2 are ideal components having zero resistance when conducting and infinite resistance when blocked. Then if all the inputs to the diodes were at HIGH or "1" level ($+V_P$) the base current of the transistor would be:

$$I_B = V_P/R_1,$$

and the base voltage

$$V_B = 0.$$

The transistor when conducting (saturated condition) would have a collector resistor current of:

$$I_{R_C} = V_P/R_C,$$

and a collector voltage

$$V_C = 0.$$

If one or more of the inputs become LOW or "0" level (0 V) the base voltage follows and base current stops. The transistor blocks current (cut-off condition) and collector voltage becomes equal to V_P. The current flowing out of the diode at LOW level, I_{GL}, is then:

$$I_{GL} = -V_P/R_1,$$

by definition negative, because current flows out. Its magnitude is equal to the base current in the previous case, i.e. when all diode inputs were at HIGH level.

To determine the loading allowable in both cases it should be remembered that the maximum collector current is:

$$I_{C\max} = \beta I_B,$$

where β is the d.c. amplification factor of the transistor, defining the saturation level of the collector current as depicted idealized in Fig. 2–3.

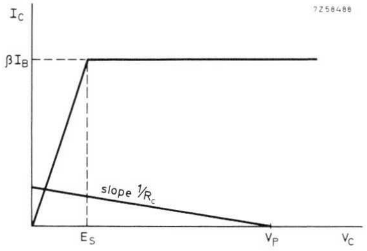


Fig. 2–3. Idealised representation of the variation of collector current I_C with collector voltage V_C.

Let us now consider the loadability of the gate when its output is in the LOW state. If we load the circuit with N_a gates of the same type, the total LOW output loading current I_{QL} would be:

$$I_{QL} = N_a |I_{GL}'|,$$

I_{GL}' being the current supplied by each gate at LOW level connected to Q. The dash indicates that the gates do not necessarily belong to the same crystal chip. The current through the collector resistor, I_{R_C}, is:

$$I_{R_C} = V_P/R_C.$$

The transistor must remain in saturation, therefore:

$$I_{C\max} \geqslant I_{QL} + I_{R_C},$$

so that:

$$\beta I_B \geqslant N_a|I_{GL}'| + V_P/R_C.$$

Assuming $|I_B|$ equal to $|I_{GL}|$, the allowable LOW state fan-out N_a is then given by:

$$N_a = \beta - R_1/R_C.$$

The above assumes that the diodes and transistors are ideal. However, a saturated transistor in practice will exhibit a small but definite potential difference between collector and emitter, as does a conducting diode between anode and cathode (we shall call these potentials E_S and E_D respectively). In general E_D is considerably greater than E_S in integrated circuits. This, however, changes the picture of gate behaviour completely, as it is now no longer possible to obtain the cut-off condition.

Let us assume that an input terminal G is at LOW level in Fig. 2–2, that is, at voltage E_S: the LOW level output voltage of the preceding gate. Then the current I_{GL} out of the G-terminal if no base current were flowing would be:

$$I_{GL} = -(V_P - E_D - E_S)/R_1,$$

and the base potential would be $E_D + E_S$; this, is, however above the value E_D necessary to make the transistor conducting. To get over this difficulty, a bias voltage must be inserted in the base circuit; this is most easily done by incorporating a diode in the base circuit. In fact two diodes have been incorporated (see Fig. 2–4) so as to increase the noise margin in the input "LOW" situation, the two diodes now making a pull-down resistance R_2 necessary. R_2 allows the base-emitter charge to flow away after cut-off; if this did not happen, speed would be greatly reduced. In addition it takes up the collector-base leakage current.

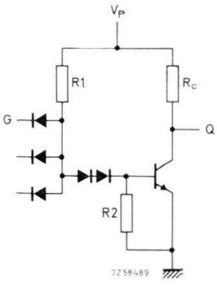


Fig. 2–4. The circuit of the DTL gates in the FC family of integrated circuits.

Having investigated the saturated transistor conditions, let us now look at the practical situation which exists when the preceding transistor (or transistors) is inversely biased (cut-off), i.e. when all inputs are at HIGH level. In this case the diodes and transistor are again not ideal because they draw leakage currents which, although in the microamp region (compared to milliamps for the LOW level currents), are still of significance as regards fan-out.

The base current I_B for HIGH level conditions will now be:

$$I_B = (V_P - 3\,E_D)/R_1 - E_D/R_2,$$

and the collector resistor current is:

$$I_{R_C} = (V_P - E_S)/R_C,$$

these expressions being comparable to those for the LOW level output conditions discussed earlier. The fan-out is again given by:

$$N_a = (\beta I_B - I_{R_C})/I_{GL},$$

or, by substituting for I_B, I_{R_C} and I_{GL}:

$$N_a \geqslant \frac{\beta - R_1/R_C - 3\beta E_D/V_P + R_1 \cdot E_s/R_C V_P - R_1 \beta E_D/V_P R_C}{1 - (E_D' + E_S')/V_P}$$

where E_D' and E_S' need not be equal to E_D and E_S.

Inserting minima and maxima for the variables in such a way that the worst (lowest) value for N_a results, will still give a value higher than that given in the Data Sheets, however. This is because an extra safety margin has been incorporated in the published value of N_a, being that V_{QL} must never exceed 0.4 V. This limits the maximum available collector current $I_{CL\max}$ to a value below $\beta_{\min} I_{B\min}$. This is shown graphically in Fig. 2-5,

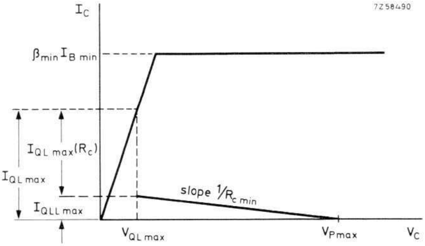


Fig. 2-5. Idealised representation of the variation of I_C with V_C, after considering diode and transistor voltage drops.

Table 2–1

type		number of inputs per function		number of functions per package	type of output
without R_C	with R_C	normal (G)	expander(E)		
FCH101, 102	FCH111, 112	8	2	1	resistive pull-up
FCH121, 122	FCH131, 132	4	1	2	
FCH141, 142	FCH161, 162	2*	1*	3	
FCH151, 152	FCH171, 172	3	0	3	
FCH181, 182	FCH191, 192	2	0	4	
FCH201, 202	FCH211, 212	1	0	6	
FCH311	FCH321	0	1	6	
FCH221, 222		3	0	2	active pull-up
FCH231		4	1	2	

* For one function only; the two remaining functions have 3 normal inputs.

where it can also be seen that $I_{QL\max}$ is further reduced by $I_{RC\max}$ to the value $I_{QL\max}(R_C)=I_{QL\max}-I_{QLL\max}$ for an internal collector resistance of R_C. Values of the above-mentioned quantities are given in the Data Sheets ("Static Data") for 0 °C, +25 °C and +75 °C and for ± 5% tolerance on V_P. The fan-out in the input "HIGH" state for a gate with internal collector resistor R_C will be:

$$N_a = I_{QL\max}/I_{GL\max}.$$

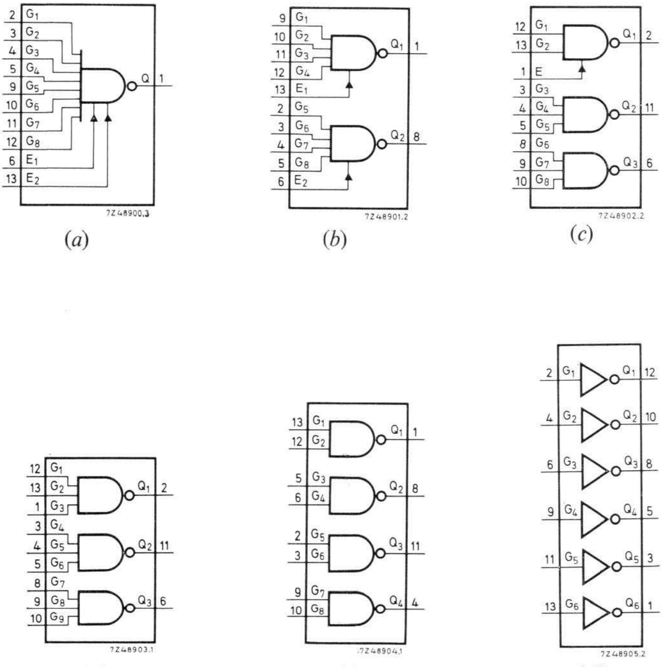
2.1.2 REVIEW OF TYPES

There are sixteen gates in the FC family. Fourteen use resistive pull-up. Seven of these have internal collector resistors, the other seven do not. Of these seven basic types, five have multiple (2 to 10) input terminals and perform the NAND function, one has a single input and performs a simple inverting function, and the last has an expander input only, enabling it to be used as a gate or as an inverter*. The remaining two types use active pull-up. The various types are listed above (Table 2–1) together with their distinguishing features.

* Dependent on the number of diodes connected to the expander input.

A 10-diode array, the “Triple Gate Expander”, is also available for use as an input expander with gates having an “*E*” input. The array has the type number FCY101 (FCY102 in the extended temperature range version).

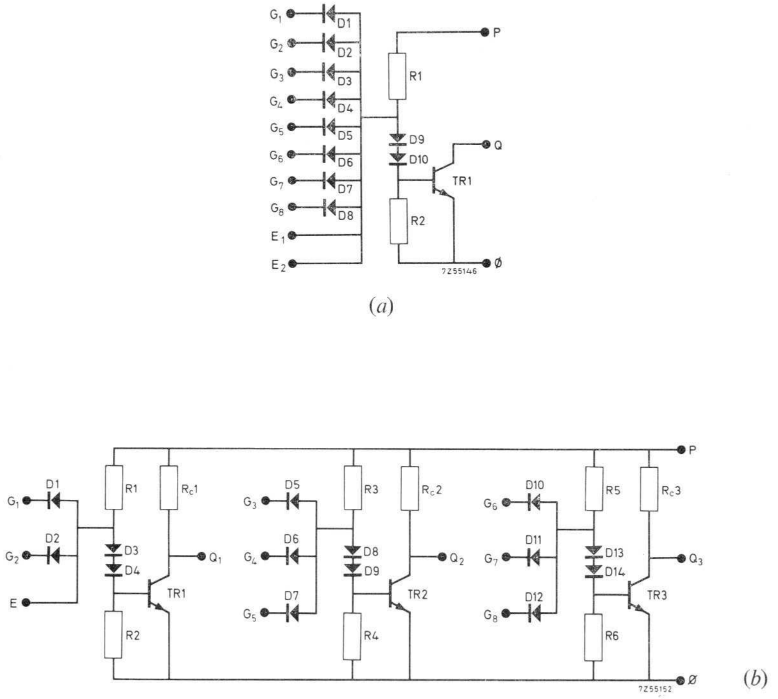
The standard temperature range package is the dual-in-line, 14-pin or 16-pin type XG14 or XG16 (extended temperature package is the flat-pack type); outline and dimensions are given in section 7.1.1; 7.1.2. Logic diagrams are shown in Fig. 2–6, and two representative circuit diagrams in Fig. 2–7, 2–8. To facilitate sketching of circuit diagrams, transparent self-adhesive “stickers” will become available. They are supplied in sheets, the stickers being printed with package outline showing terminal positions, numbers and symbols for each gate type.



(*a*) (*b*) (*c*)

(*d*) (*e*) (*f*)

Fig. 2–6. Logic diagrams of FC family gate packages. (*a*) *FCH101, FCH111;* (*b*) *FCH121, FCH131;* (*c*) *FCH141, FCH161;* (*d*) *FCH151, FCH171;* (*e*) *FCH181, FCH191,* (*f*) *FCH201, FCH211.*



(*a*)

(*b*)

Fig. 2–7. Circuit diagrams of representative FC family gates using resistive pull-up. (a) FCH101, (b) FCH161.

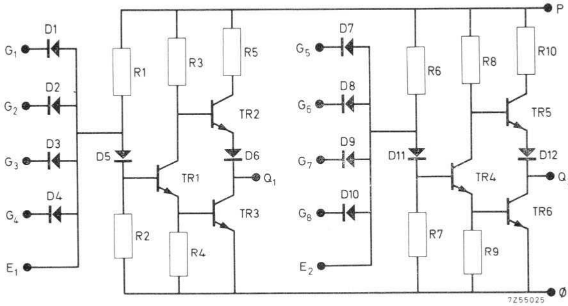


Fig. 2–8. Circuit diagram of FCH231 (2-gate package using active pull-up).

2.1.3 PROPAGATION DELAY

All gates in the FC family have the same rise propagation delay (t_{pdr}) and fall propagation delay (t_{pdf}), given similar operating conditions. The terms are illustrated in Fig. 2–9.

The available fan-out N_a and the wiring capacitance C_w are two factors which strongly influence the propagation delays. Figs 2–10 and 2–11 give values of t_{pdr} and t_{pdf} respectively for various situations.

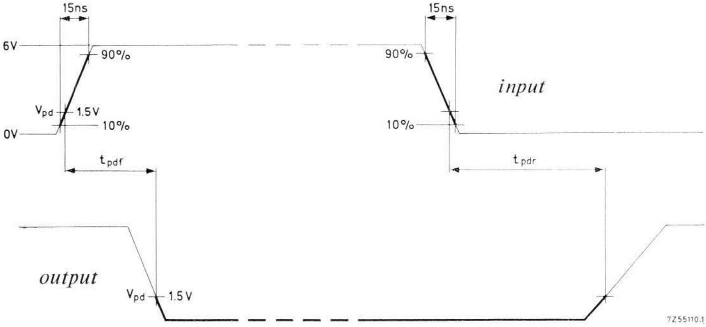


Fig. 2-9. Timing diagram illustrating the terms t_{pdf} and t_{pdr}.

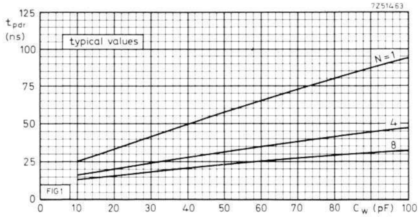


Fig. 2–10. Relationship between t_{pdr} and wiring capacitance C_w for various values of fan-out N.

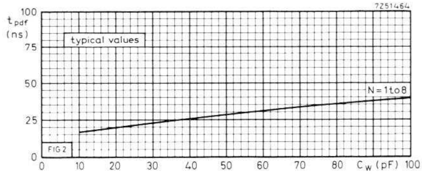
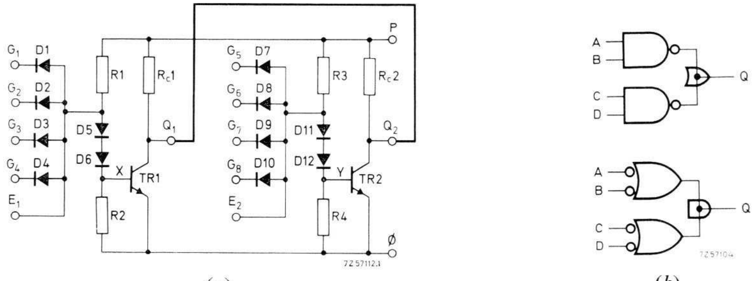


Fig. 2–11. Relationship between t_{pdf} and wiring capacitance C_w for fan-outs of from 1 to 8.

2.1.4 WIRED-OUTPUT LOGIC FUNCTIONS

General

It often occurs in logic circuits that the outputs of two logic gates must be combined in a further logic function (AND or OR). This function could be performed by using another logic gate (or two or more); however, the same effect can be achieved by simply strapping the two outputs together. This latter method is given the name "Wired-Output".* In this section the limitations and benefits of the method are given, so far as the FC family of integrated circuits is concerned. At the end are some design rules and nomograms.



(*a*) (*b*)

Fig. 2–12. Wired-Output configuration. (*a*) *Two NAND gates (the FCH131 circuit) as a Wired-Output function;* (*b*) *Symbolic representation of two NAND gates in a Wired-Output function.*

Fig. 2–12*a* shows the circuit diagram of two 4-input NAND gates with their outputs connected together, as described above. Q will fall to the LOW or "0" level if either TR_1 or TR_2 go into conduction, that is, if one of the transistor bases (X or Y) is brought to HIGH or "1" level. G_1, G_2, G_3 and G_4, or G_5, G_6, G_7 and G_8 must thus be at "1" level. In Boolean Algebra Q can be expressed as:

$$X + Y = \bar{Q},$$

or, in terms of G:

$$G_1 \cdot G_2 \cdot G_3 \cdot G_4 + G_5 \cdot G_6 \cdot G_7 \cdot G_8 = \bar{Q}.$$

[*] Also known as "Wired-OR", "output-OR" or "collector-OR".

Alternatively, this can be expressed as an AND function:

$$(\bar{G}_1 + \bar{G}_2 + \bar{G}_3 + \bar{G}_4) \cdot (\bar{G}_5 + \bar{G}_6 + \bar{G}_7 + \bar{G}_8) = Q.$$

Fig. 2–12*b* shows the symbols which are used to indicate a Wired-Output function.

The limitation of this method is that it is only possible to use it with gates having "resistive pull-up" (resistors R_{C1} and R_{C2} in Fig. 2–12*a*). Wired-Output cannot be used if the gate employs "active pull-up", as in the line-driver NAND gates FCH221 and FCH231 (transistors TR_2 and TR_3 in Fig. 2–13). With active pull-up, either TR_2 or TR_3 in Fig. 2–13 will be conducting, dependent on the state of the gate output. If the outputs Q of two such gates were to be connected together in Wired-Output, a short-circuit might occur between TR_2 of one gate and TR_3 of the other.

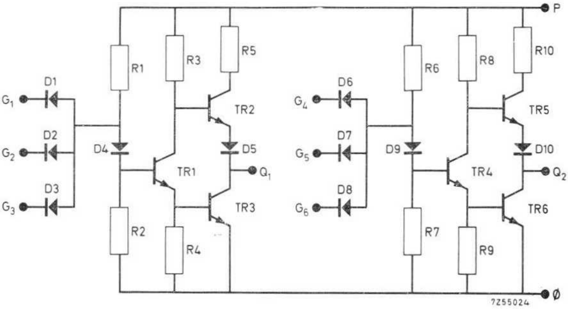


Fig. 2-13. Two NAND gates with active pull-up (the FCH221 circuit).

At the moment, the two line-drivers are the only gates of the FC family employing active pull-up.

The advantages of using Wired-Output are twofold:

— a smaller number of gates is necessary;

— the propagation delay time of the circuit may be reduced.

The circuit shown in Figs 2–14, 15 and 16 exemplify these advantages. Fig. 2–14a shows a typical logic circuit performing the function (positive logic):

$$\overline{P \cdot A + P \cdot B + P \cdot C} = Q.$$

Using gates with resistive pull-up, one gate and the inverter can be saved by employing Wired-Output as shown in Fig. 2–14*b*, this arrangement producing the same function as Fig. 2–14*a*.

Fig. 2–15*a* shows a two-bit comparator. This performs the function:

$$A_1 \cdot \bar{B} + B_1 \cdot \bar{A}_1 + A_2 \cdot \bar{B}_2 + B_2 \cdot \bar{A}_2 = Q,$$

the inverse of which can be obtained using Wired-Output (Fig. 2–15*b*) with a subsequent saving of three gates.

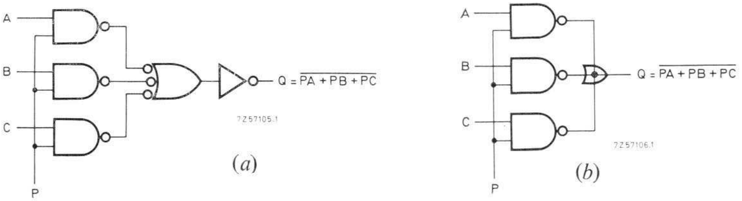


Fig. 2–14. Logic circuit using (a) four NAND gates, and (b) three NAND gates in Wired-Output. (equation for positive logic)

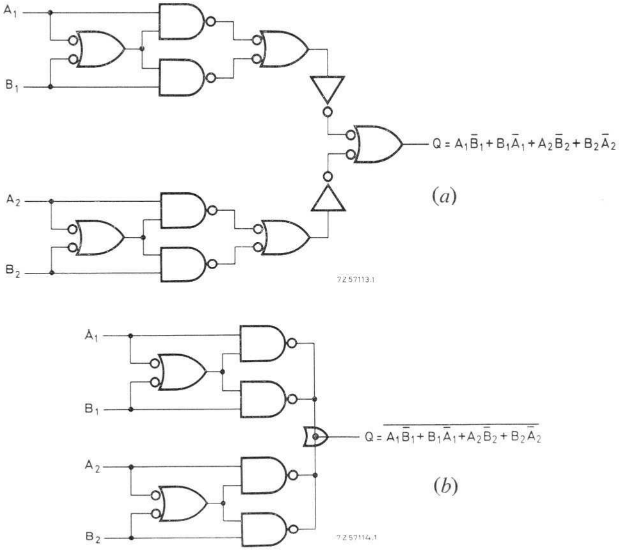

A_1
B_1
A_2
B_2
$Q = A_1\bar{B}_1 + B_1\bar{A}_1 + A_2\bar{B}_2 + B_2\bar{A}_2$
(a)
7Z57113.1
A_1
B_1
A_2
B_2
$Q = \overline{A_1\bar{B}_1 + B_1\bar{A}_1 + A_2\bar{B}_2 + B_2\bar{A}_2}$
(b)
7Z57114.1

Fig. 2–15. Two-bit comparator (a) nine NAND gates, and (b) six NAND gates in Wired-Output. (equation for positive logic)

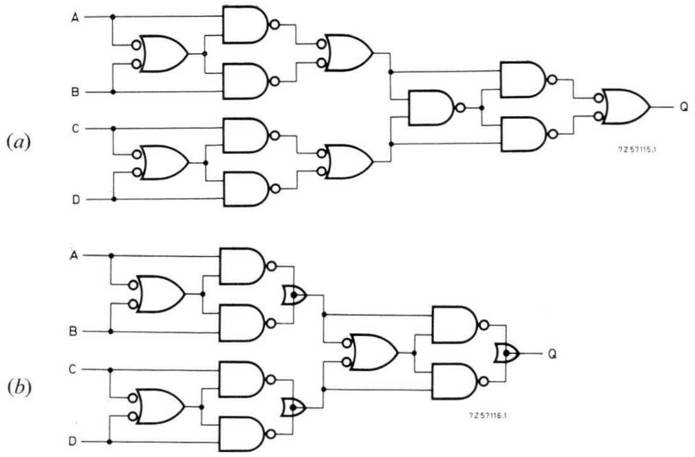


Fig. 2–16. Four-bit parity checker using (a) twelve NAND gates, and (b) nine NAND gates in Wired-Output.

Fig. 2–16*a* is a four-bit parity checker. In Fig. 2–16*b* it is apparent that three gates have been saved by using Wired-Output.

Where the number of stages is decreased by the saving of gates after employing Wired-Output, the propagation delay time is also decreased. For example, in Fig. 2–14*b* the total propagation delay time is reduced by a single stage delay time, in Fig. 2–15*b* by three stage delay times, and in Fig. 2–16*b* by two stage delay times (the propagation delay of the stage itself is also affected; see under "Dynamic Characteristics", p. 32).

Practical Aspects

There are two aspects which must be considered when gate outputs are connected in Wired-Output: static loading, and the dynamic characteristics.

STATIC LOADING

The influence of the Wired-Output configuration on both the fan-out and the noise margin must be investigated.

The FC family contains gates both with and without an internal collector resistor, and as the fan-out is largely determined by the collector resistor there are three cases to consider:

case 1: gates having a collector resistor;
case 2: some gates having, and others not having, a collector resistor;
case 3: gates not having a collector resistor (i.e. having only a single external common collector resistor R_{CC}).

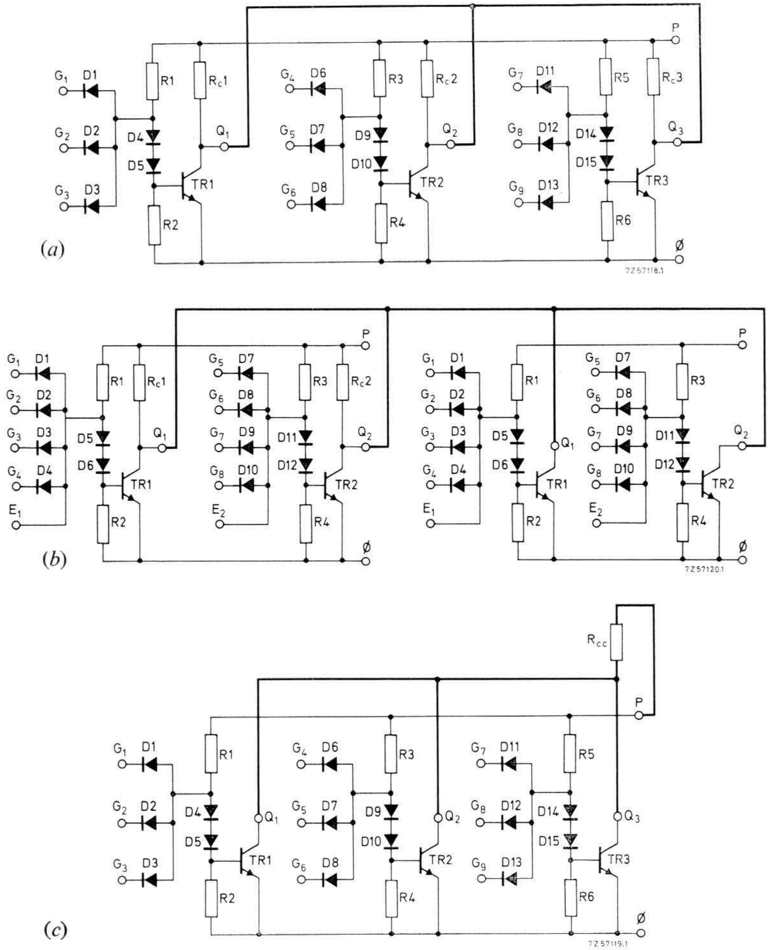


Fig. 2-17. Gate circuits with and without internal collector resistor R_C. (a) Three NAND gates (the FCH171 circuit), each with R_C, in Wired-Output. (b) Four NAND gates (the FCH131, 121 circuit), two with R_C and two without, in Wired-Output. (c) Three NAND gates (the FCH151 circuit), all without R_C, in Wired-Output with a common collector resistor R_{CC}.

If all gates have an internal collector resistor (Fig. 2–17*a*), the maximum number of Wired-Output functions is limited by the available output current in the LOW state (current I_{QL}) of the driving gate.

If only some of the gates have an internal collector resistor (Fig. 2–17*b*) the maximum number of such gates is again determined by I_{QL}. However, in this case there is also a minimum allowable number of gates having an internal collector resistor; this is determined by the voltage drop caused by leakage currents across the common (paralleled) collector resistors.

If none of the gates involved have an internal collector resistor (Fig. 2–17*c*), the Wired-Output configuration can be applied to a greater number of gates than in either of the other two cases. This is because the single common collector resistor R_{CC} which is necessary, can be accurately chosen for the optimal value to suit the actual number of gates concerned.

Convenient methods of determining the permissible number of gates for each of the three cases are given below.

Case 1. (All gates contain an internal collector resistor.) Available fan-out of a single standard-temperature-range FC gate is 8. This figure must, however, be reduced when employing Wired-Output to several gates, because the collector resistors are paralleled. The fan-out N must be decreased by 1.15 (the ratio between $I_{QL L\mathrm{max}}$* and $I_{GL\mathrm{max}}$* for each additional gate connected. The maximum gate output current I_{QL} should never be exceeded.

If only one of the transistors in the Wired-Output configuration conducts and all the loading gates need full gate current (the most unfavourable loading situation), the maximum allowable number of gates (A) is given by:

$$A < \frac{I_{QL} - NI_{GL\mathrm{max}}}{I_{QL L\mathrm{max}}} + 1.$$

Values of A for various values of N are worked out in Table 2–2. The values are for the standard-temperature-range FC gates, using I_{QL}, $I_{QL L\mathrm{max}}$ and $I_{GL\mathrm{max}}$ values as given in the Data Sheets.

Table 2–2

N	1	2	3	4	5	6	7	8
A	7	6	5	4	3	2	1	1

* $I_{QL L\mathrm{max}}$ = maximum current which may be taken by the collector resistor. $I_{GL\mathrm{max}}$ = maximum LOW input current per gate.

Case 2. (Some gates with and some without an internal collector resistor.) Gates without collector resistor do not load he driving gate(s), so that in this case a larger number of gates can be connected in the Wired-Output configuration.

The maximum number of internal-R_C gates that may be connected is determined by I_{QL} (LOW state output current), as in Case 1.

The minimum number of internal-R_C gates that may be connected is determined by the HIGH output state of the driving gates. The total leakage current of outputs and inputs can cause such a voltage drop across the parallel collector resistors that the HIGH state noise margin M_H is worsened. The minimum number of internal-R_C gates (A) is:

$$A > \frac{(BI_{QH\text{max}} + NI_{GH\text{max}})R_{C\text{max}}}{V_{P\text{min}} - (V_{GH} + M_H)},$$

where B = total number of Wired-Output gates (both internal-R_C and non-internal-R_C types);
$I_{QH\text{max}}$ = maximum state gate output current;
$I_{GH\text{max}}$ = maximum state gate input current;
$R_{C\text{max}}$ = maximum value of an internal collector resistor;
$V_{P\text{min}}$ = minimum supply voltage;
V_{GH} = HIGH-state threshold voltage;
M_H = HIGH-state d.c. noise margin.

Worst-case values occur at 75 °C, when $I_{QH\text{max}} = 70$ μA,
$I_{GH\text{max}} = 25$ μA, $R_{C\text{max}} = 7$ kΩ, and $V_{GH} = 2.1$ V.
$V_{P\text{min}} = 5.7$ V. M_H should be taken as 1.8 V.

Table 2–3 gives both minimum and maximum values of A for various values of B and N, assuming worst-case values for the other variables. Taking Fig. 2–18 as an example, assuming $B = 7$ and $N = 4$, it can be seen from Table 2–3 that a minimum of 3 and a maximum of 4 gates with internal R_C can be connected in Wired-Output configuration.

Case 3. (Gates not having an internal collector resistor.) A further extension of the permissible number of gates in Wire-Output configuration can be achieved by using only gates without internal collector resistor. This necessitates the use of a common external collector resistor which then can be chosen to exactly suit the prevailing conditions.

If we substitute R_{CC} for $R_{C\text{max}}/A$ in the expression for A in Case 2, the following expression is obtained:

$$R_{CC} \leqslant \frac{V_{P\text{min}} - (V_{GH} + M_H)}{BI_{QH\text{max}} + NI_{GH\text{max}}}.$$

Substituting actual data gives:

$$R_{CC} \leqslant 1.8/(0.07B + 0.025N),$$

where R_{CC} is expressed in kΩ.

The minimum value of R_{CC} can be stated as:

$$R_{CC} \geqslant \frac{V_P - V_{QL\mathrm{max}}}{I_{QLL\mathrm{max}}},$$

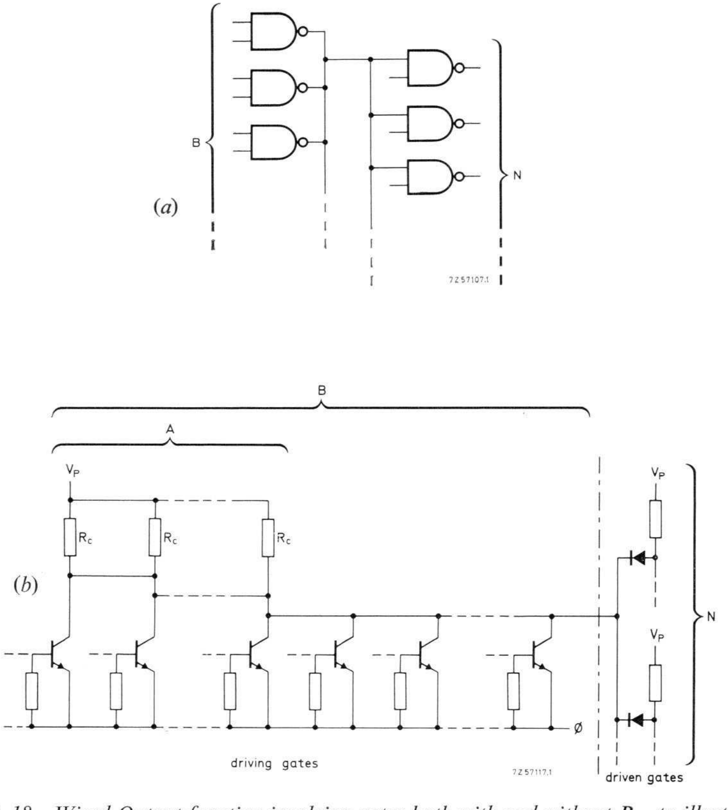


Fig. 2–18. Wired-Output function involving gates both with and without R_C, to illustrate the terms used. (a) Logic diagram, and (b) electrical circuit.

Table 2–3

maximum number of wired-output gates, B	minimum and maximum numbers of internal R_C gates for various values of N (number of driven gates)															
	N = 1		N = 2		N = 3		N = 4		N = 5		N = 6		N = 7		N = 8	
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
2	1	2	1	2	1	2	1	2	1	2	2	2				
3	1	3	1	3	2	3	2	3	2	3	2	2				
4	2	4	2	4	2	4	2	4	2	3	2	2				
5	2	5	2	5	2	5	2	4	2	3	2	2				
6	2	6	2	6	2	5	2	4	3	3						
7	2	7	3	6	3	5	3	4	3	3						
8	3	7	3	6	3	5	3	4	3	3						
9	3	7	3	6	3	5	3	4	3	3						
10	3	7	3	6	3	5	4	4								
11	4	7	4	6	4	5	4	4								
12	4	7	4	6	4	5	4	4								
13	4	7	4	6	4	5	4	4								
14	4	7	4	6	5	5										
15	5	7	5	6	5	5										
16	5	7	5	6	5	5										
17	5	7	5	6	5	5										
18	5	7	6	6												
19	6	7	6	6												
20	6	7	6	6												
21	6	7	6	6												
22	7	7														
23	7	7														
24	7	7														
25	7	7														

or alternatively:

$$R_{CC} \geqslant \frac{V_P - V_{QL\max}}{I_{QL\max} - N|I_{GL\max}|}.$$

Substituting actual data then gives:

$$R_{CC} \geqslant 5.9/(16 - 1.8N),$$

where R_{CC} is expressed in kΩ.

Table 2–4 gives the maximum and minimum values of R_{CC} necessary for various values of N. The maximum value of B for each case is also given.

Taking Fig. 2–18 as an example ($B = 7$, $N = 4$), from Table 2–4 it can be seen that a common collector resistor R_{CC} of value between 0.68 kΩ and 0.82 kΩ will be necessary if all the Wired-Output gates are to be non-internal-R_C types. However, the table gives worst-case values ($B = 30$), and from the formula for R_{CC} (maximum value) it appears that R_{CC} can be a maximum of 3.0 kΩ when $B = 7$.

Table 2–4

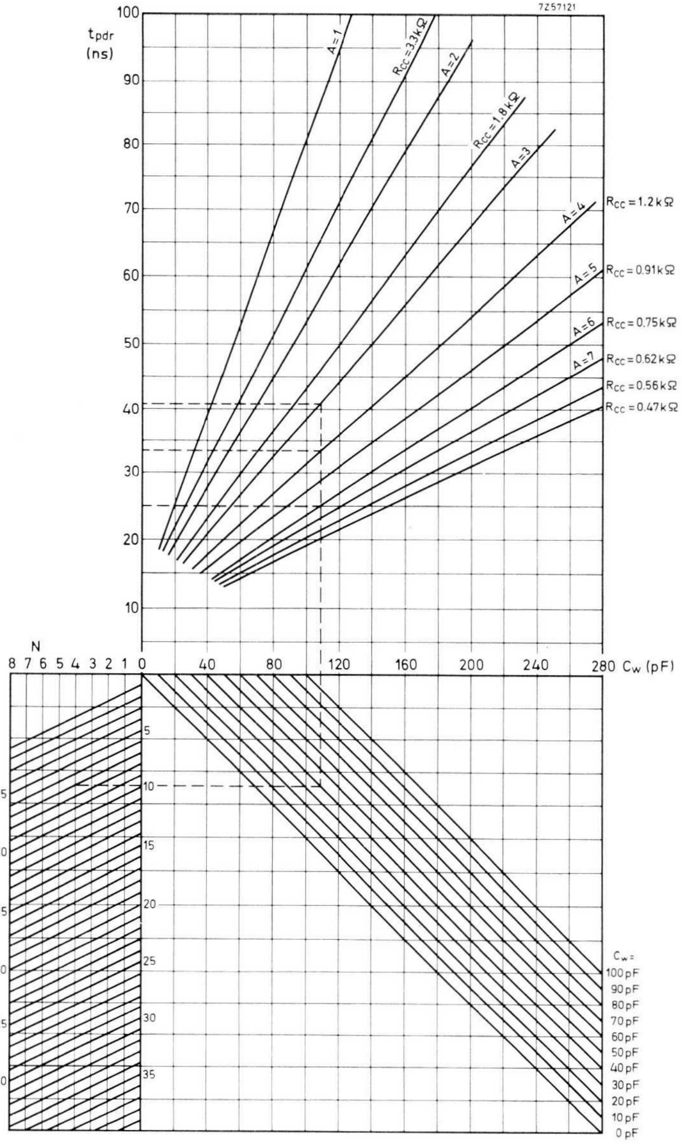
maximum fan-out	value of common collector resistance		max. no. of Wired-Output gates, B
	R_{CC}* (maximum) kΩ	R_{CC}* (minimum) kΩ	
1	0.51	0.43	49
2	0.62	0.51	41
3	0.62	0.56	37
4	0.82	0.68	30
5	1.0	0.91	24
6	1.3	1.2	17
7	2.0	1.8	10
8	3.7	3.7	4

* The E24 standard range of resistors (± 5%) is suitable.

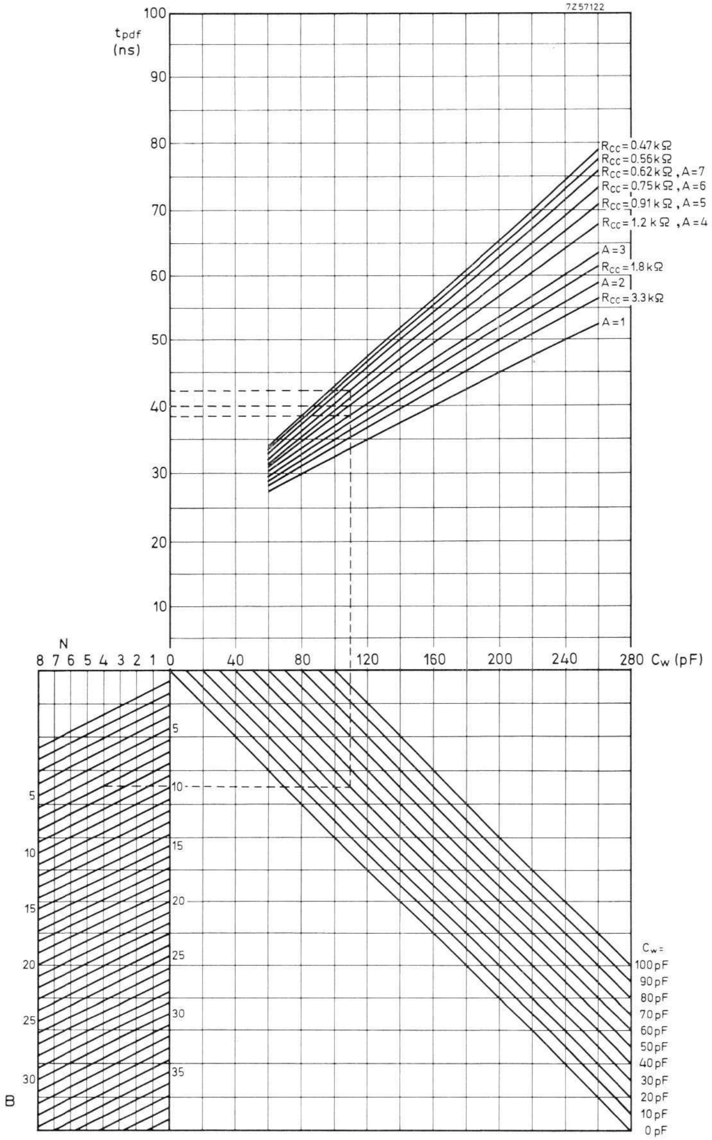
DYNAMIC CHARACTERISTICS

The propagation delay time of a circuit is affected in two ways when gates are connected in Wired-Output. Firstly, use of Wired-Output may save several stages of logic, and thus reduce overall propagation delay by several stage delays, as noted in "General". Secondly, the propagation delay of the stage itself may be reduced *or increased* by using Wired-Output. This second point is discussed further below, under the headings "rise propagation delay" and "fall propagation delay".

The typical rise propagation delay t_{pdr} for various conditions can be found in Nomogram 2–1. The delays were measured with the aid of circuits similar to that of Fig. 2–19*a*, and are worst-case (i.e. only one of the connected gates in Wired-Output active and only one of the loading gates supplying current to the driving stage). The delay can be decreased by decreasing R_{CC}.



Nomogram 2-1. Rise propagation delays t_{pdr} in gates of the FC family for various conditions.



Nomogram 2-2. Fall propagation delays t_{pdf} in gates of the FC family for various conditions.

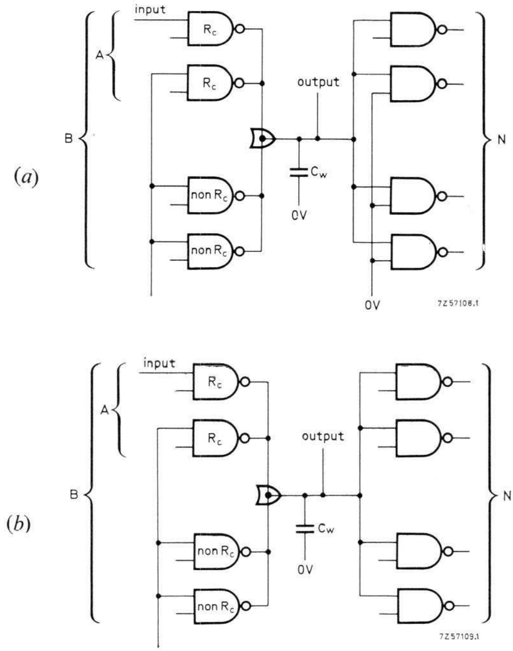


Fig. 2-19. Propagation delay measurement circuits for (a) t_{pdr}, and (b) t_{pdf} (see Fig. 2.9).

Taking Fig. 2–19*a* as an example, with $B = 7$, $N = 4$ and $C_w = 40$ pF, we find from Nomogram 2–1 three values of t_{pdr} as given in the list below.

If $A = 3$, $t_{pdr} = 41$ ns

If $A = 4$, $t_{pdr} = 33$ ns.

If all the gates in Wired-Output are non-internal-R_C types (external collector resistor of 750 Ω), $t_{pdr} = 25$ ns.

The typical fall propagation delay t_{pdr} can be found from Nomogram 2–2. Circuits similar to that of Fig. 2–19*b* were used, with the most unfavourable drive and load conditions. The delay is decreased by increasing R_{CC}.

Again taking Fig. 2–19*b* as an example ($B = 7$, $N = 4$, $C_w = 40$ pF), we obtain the following values for t_{pdf}.

If $A = 3$, $t_{pdf} = 38$ ns

If $A = 4$, $t_{pdf} = 40$ ns.

If all the gates in Wired-Output are non-internal-R_C types ($R_{CC} = 750$ Ω), $t_{pdf} = 42$ ns.

2.1.5. EXPANDER INPUT DIODES AND UNUSED INPUTS

Expander Input Diodes

Gates equipped with one or more "Expander" inputs — denoted E in the terminology — make expansion of the fan-in of the gates an easy matter. No diode is incorporated in the E inputs, so that an external diode must be added, one per extra input signal, as discrete components. In Fig. 2–20 two extra inputs X and Y have been added to the basic 4-input NAND gate (as contained in the FCH121 for example). Output Q is then equal to $\overline{A \cdot B \cdot C \cdot D \cdot X \cdot Y}$.

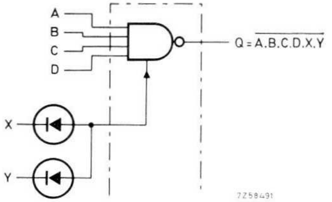


Fig. 2–20. Half of the FCH121 package, showing extra input signals X and Y connected to the expander input E of the gate.

A special integrated circuit in the FC family, the Triple Gate Expander FCY101, has been developed for use with E inputs of other gates. It contains three AND gates (one 4-input and two 3-input) as shown in Fig. 2–21.

If separate diodes are to be used, type BAX13 is recommended. To reduce noise entry, it is advisable to mount the expander diodes or gate as close as possible to the expander input terminal.

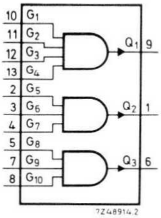


Fig. 2–21. The FCY101 package, containing three AND gates.

Unused Inputs

It will frequently be found that some gates in the system have one or more inputs which are not used. These inputs can either be left unconnected or connected to supply voltage (i.e. at HIGH). The former alternative is recommended, as with the latter method interference in the power supply may affect the gate. Another method is to connect an unused input to one or other used input of the gate, but this increases the gates propagation delay because of the extra diode capacitance of the unused input.

2.1.6 LINE DRIVERS (GATES WITH LARGE FAN-OUT)

FC family Line Drivers (FCH221, FCH231)

The types FCH101-FCH211 (FCH102-FCH212) have a guaranteed fan-out of 8, and most of these will in practice have a larger capability. Nevertheless, the driving of many FC inputs at a time for example can require still larger fan-out, and to meet this requirement the "Line Driver" types FCH221 ($N_a = 14$) and FCH231 ($N_a = 20$) have been introduced. The inputs of these gates are matched (regarding input threshold voltage and current) to the outputs of the other circuits in the FC family; further information is contained in section 1.2.1.

Discrete Component Line Driver

Even the Line Driver gates may not have sufficient output for some circuits however, as when triggering a large capacity shift register synchronously. In such cases a driver made up of discrete components, such as the circuit shown in Fig. 2–22, will be a solution. The fan-out capability

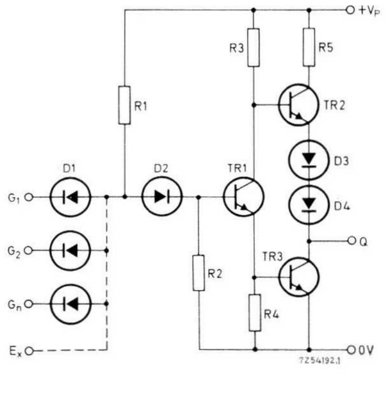


Fig. 2–22. "Heavy-duty" line driver circuit from discrete components.

can be varied by changing resistors R_1 and R_3. Table 2–5 shows the quantity of various integrated circuits from the FC family which can be driven by the output of the heavy duty line driver. Table 2–6 gives the static and dynamic characteristics of the circuit (p. 40, 41).

Table 2–5 Quantity of ICs of any one type that can be driven by the heavy duty line driver.

IC type and input to be driven		quantity that can be driven			
type	input	$R_1 = 3.9$ kΩ $R_3 = 470$ Ω	$R_1 = 3.3$ kΩ $R_3 = 390$ Ω	$R_1 = 2.7$ kΩ $R_3 = 270$ Ω	$R_1 = 2.2$ kΩ $R_3 = 220$ Ω
FCH101 to FCH211	G, E	28	56	84	112
FCJ101	J-K	28	56	84	112
	S	14	28	42	56
	T	8	16	24	32
FCJ111, FCJ201	J-K	28	56	84	112
	S	18	36	54	72
	T	14	28	42	56
FCJ121, FCJ131 FCJ191, FCJ211	J-K	35	70	105	140
	S	8	16	24	32
	T	12	24	36	48

2.1.7 STABILITY CONSIDERATIONS

Many logic circuits require that several gates be connected in series. This is always permissible when driving the gates by pulses from other members of the FC family, but caution is necessary when the driving pulses are generated by an external source (for example the mains supply) having relatively long rise and fall times. In such cases the gates remain in the active linear region for a relatively long time, and if feedback is present, conditions exist which could cause oscillation upon a change of state.

We shall consider two cases:

(a) three NAND gates in series with capacitive feedback from output of the second to input of the first;

(b) four NAND gates in series with capacitive feedback from output of the third to input of the first.

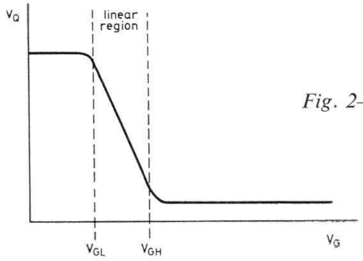


Fig. 2–23. General input-output characteristic of a gate.

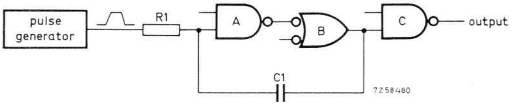
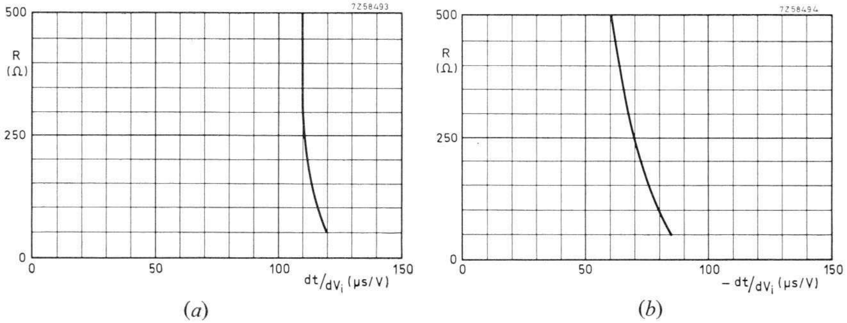


Fig. 2–24. Three gates in series in a stability test.

In Fig. 2–23 the output voltage V_Q is plotted against the input voltage V_G of an odd number of gates. If V_G is changed slowly between 0.8 V and 2.1 V (the linear amplifying region) and feedback is present, oscillations could occur before the output reaches a steady state.

Case a: Three gates with capacitive feedback from the second to the first.

The test set-up is shown in Fig. 2–24. Capacitor C_1 simulates wiring capacitance. To obtain highest internal feedback the three gates are all on the same I.C. (in this case the FCH191). R_1 is the internal source resistance of the pulse generator. Rise time t_r and fall time t_f of the input signal (voltage magnitude V_i) were adjusted to just give stable conditions for various values of R_1 with C_1 at 5 pF. The allowable times t_r and t_f can be found from the time gradients plotted in Figs 2–25*a* and *b*;



(*a*)

(*b*)

Fig. 2–25. Allowable rise and fall time gradients of the input pulse with respect to the value of R_1, with C_1 = 5 pF, (a) Rise time gradient dt/dV_i *vs.* R_1*; (b) fall time gradient* -dt/dV_i *vs.* R_1.

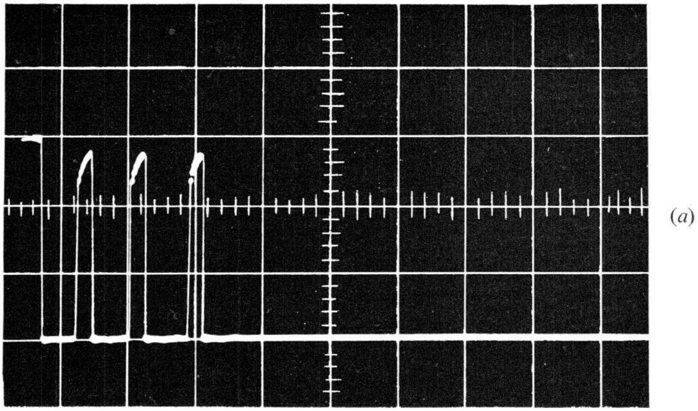
Table 2–6

characteristics			fan-out = 28		
	condition: V_P	symbol	0 °C	25 °C	75 °C
Static characteristics (each gate)					
LOW input current at $V_G = 0.4$ V	5.7 V	$-I_{GL\mathrm{max}}$	1.38	1.4	1.42
	6.3 V	$-I_{GL\mathrm{max}}$	1.58	1.6	1.61
HIGH input current at $V_G = V_{QH\mathrm{min}}$; other inputs at 0 V	5.7 V	$I_{GH\mathrm{max}}$	1.0	1.0	25
LOW output voltage at		$V_{QL\mathrm{max}}$	0.4	0.4	0.4
LOW output current	5.7 V	$I_{QL\mathrm{max}}$	49	46.2	43.4
and all inputs at	6.3 V	$I_{QL\mathrm{max}}$	56	53.2	50.4
HIGH input voltage		$V_{GH\mathrm{min}}$	2.4	2.2	2.0
HIGH output voltage; $R_{Q\Phi} = 1$ MΩ	5.7 V	$V_{QH\mathrm{min}}$	3.6	3.7	3.9
HIGH output voltage drop with loading (I_Q in μA)	5.7 V	$\Delta V_{QH\mathrm{max}}$	$0.04 I_Q$	$0.04 I_Q$	$0.04 I_Q$
LOW input voltage		$V_{GL\mathrm{max}}$	1.0	1.0	1.0
Dynamic characteristics					
time reference level		V_{Pd}	1.5	1.5	1.5
rise propagation delay*	6.0 V	t_{pdr}	—	45	—
fall propagation delay*	6.0 V	t_{pdf}	—	25	—
input capacitance (equivalent)	6.0 V	C_G	—	7	—
supply current					
output LOW	6.0 V	I_{PL}	—	12	—
output HIGH	6.0 V	I_{PH}	—	1.2	—
D.C. noise margin**					
from FCH gate to line driver	5.7 V	M_L	0.6	0.6	0.6
	5.7 V	M_H	2.9	3.1	3.3
from line driver to FCH gate	5.7 V	M_L	0.6	0.6	0.6
	5.7 V	M_H	1.3	1.5	1.8
power consumption † (50 % duty cycle)			—	40	—

* typical, with wiring capacitance $C_w = 6.4$ pF per driven gate.

** minimum. † typical.

fan-out = 56			fan-out = 84			fan-out = 112			unit
0 °C	25 °C	75 °C	0 °C	25 °C	75 °C	0 °C	25 °C	75 °C	
1.63	1.65	1.67	2.0	2.02	2.04	2.45	2.47	2.49	mA
1.87	1.89	1.9	2.28	2.3	2.32	2.8	2.83	2.85	mA
1.0	1.0	25	1.0	1.0	25	1.0	1.0	25	μA
0.4	0.4	0.4	0.4	0.4	0.4	0.4	0.4	0.4	V
98	92.4	86.8	147	138.6	130.2	196	184.8	173.6	mA
112	106.4	100.8	168	159.6	151.2	224	212.48	201.6	mA
2.5	2.2	2.0	2.5	2.3	2.1	2.6	2.4	2.2	V
3.6	3.7	3.9	3.6	3.7	3.9	3.6	3.7	3.9	V
$0.04I_Q$	$0.04I_Q$	$0.04I_Q$	$0.04I_Q$	$0.04I_Q$	$0.04I_Q$	$0.04I_Q$	$0.04I_Q$	$0.04I_Q$	mV
1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	V
1.5	1.5	1.5	1.5	1.5	1.5	1.5	1.5	1.5	V
—	60	—	—	75	—	—	90	—	ns
—	25	—	—	25	—	—	25	—	ns
—	7	—	—	7	—	—	7	—	pF
—	14	—	—	18	—	—	20	—	mA
—	1.4	—	—	1.8	—	—	2.0	—	mA
0.6	0.6	0.6	0.6	0.6	0.6	0.6	0.6	0.6	V
2.8	3.1	3.3	2.8	3.0	3.2	2.7	2.9	3.1	V
0.6	0.6	0.6	0.6	0.6	0.6	0.6	0.6	0.6	V
1.3	1.5	1.8	1.3	1.5	1.8	1.3	1.5	1.8	V
—	50	—	—	60	—	—	70	—	mW



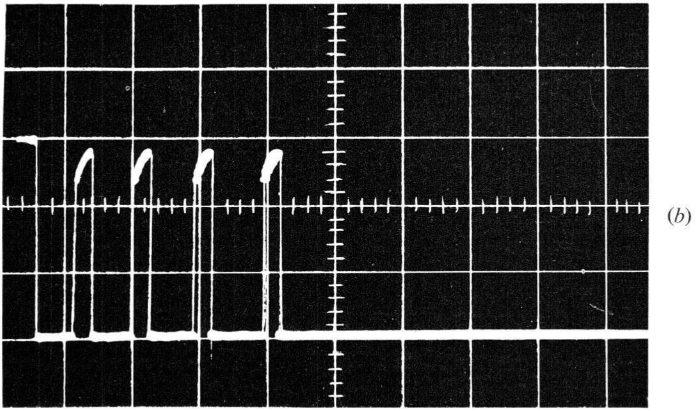


Fig. 2–26. Oscillograms of the output signal in the three-gate test, with $\mathrm{d}t/\mathrm{d}V_i$ *equal to 500* $\mu s/V$ *(rising); (a)* $C_1 = 5$ *pF; (b)* $C_1 = 10$ *pF,*

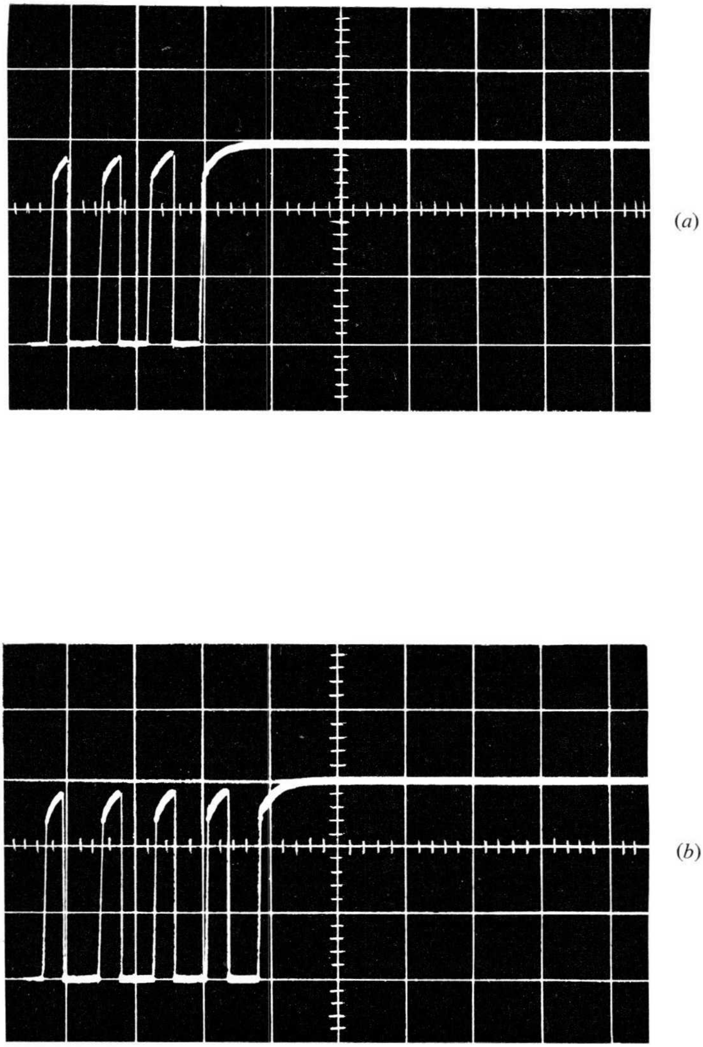


Fig. 2–27. Oscillograms of the output signal in the three-gate test, with dt/dV_i *equal to −150 µs/V (falling); (a)* $C_1 = 5$ *pF; (b)* $C_1 = 10$ *pF.*

they are different because the amount of feedback depends on the $\mathrm{d}t/\mathrm{d}V$ of the output of the second gate ($t_f < t_r$). Figs 2–26*a* and *b* are oscillograms of the third gate output signal for $C_1 = 5$ pF and 10 pF respectively, with $\mathrm{d}t/\mathrm{d}V_i$ of the input signal equal to 500 μs/V (rising). Figs 2–27*a* and *b* show the output for $C_1 = 5$ pF and 10 pF respectively, but with $\mathrm{d}t/\mathrm{d}V_i$ of input signal equal to -150 μs/V (falling).

Case b: Four gates with capacitive feedback from the third to the first.

Fig. 2–28 shows the test set-up. With this configuration the feedback is negative.

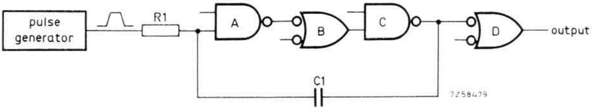
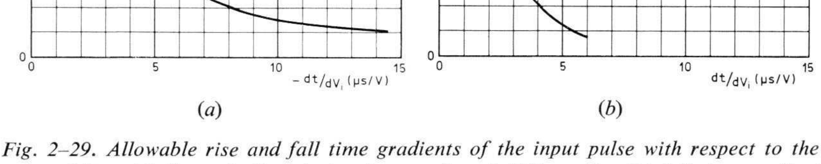


Fig. 2–28. Four gates in series in a stability test.

Fig. 2–29*a* and *b* show maximum rise and fall time gradients respectively, which cause no oscillation of the output, for various values of R_1. Figs 2–30*a* and *b* show the output signal resulting from a rising input of 30 μs/V ($C_1 = 5$ pF and 10 pF respectively), and Figs 2–31*a* and *b* show the output for a falling input of 4,5 μs/V ($C_1 = 5$ pF and 10 pF respectively).

In general, if the input time gradient $\mathrm{d}t/\mathrm{d}V_i$ exceeds 1 μs/V, an interface circuit such as the FCL101 (Level Detector) will be advisable.



(*a*)

(*b*)

Fig. 2–29. Allowable rise and fall time gradients of the input pulse with respect to the value of R_1, with $C_1 = 5$ pF; (a) rise time gradient $\mathrm{d}t/\mathrm{d}V_i$ vs. R_1; (b) fall time gradient -$\mathrm{d}t/\mathrm{d}V_i$ vs. R_1.

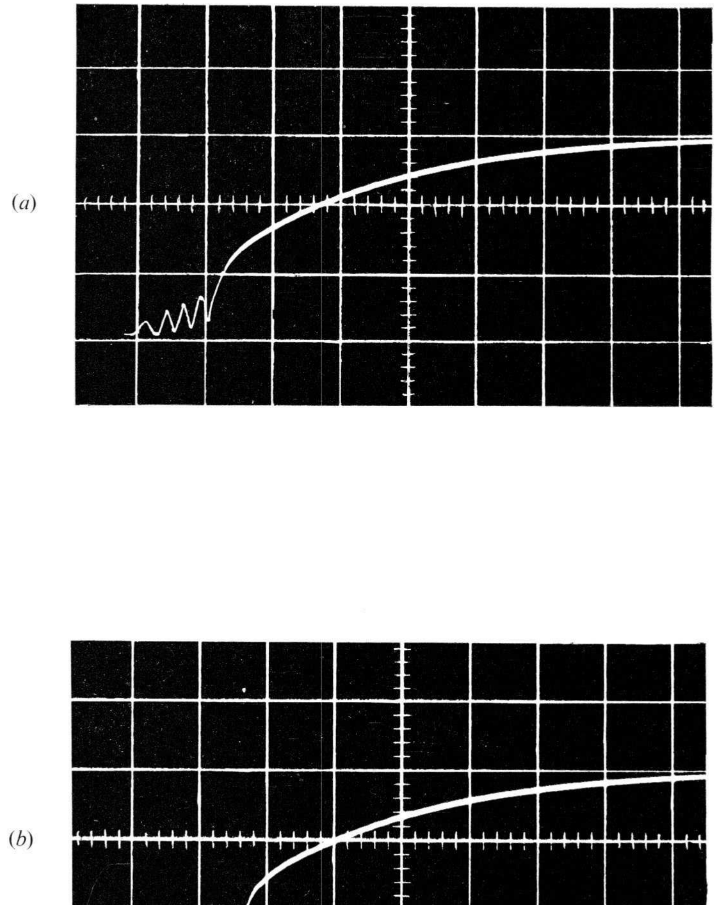


Fig. 2–30. Oscillograms of the output signal in the four-gate test, with $\mathrm{d}t/\mathrm{d}V_i$ *equal to 30 μs/V (rising); (a)* $C_1 = 5$ *pF; (b)* $C_1 = 10$ *pF.*

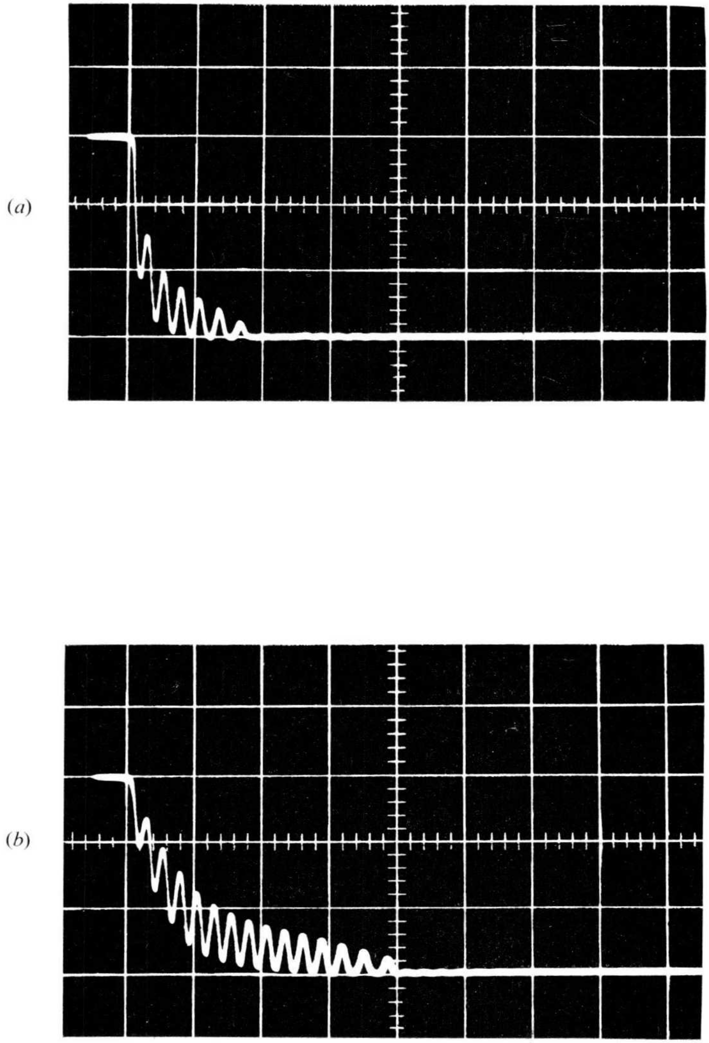


Fig. 2–31. Oscillograms of the output signal in the four-gate test, with $\mathrm{d}t/\mathrm{d}V_i$ *equal to* -4.5 *μs/V (falling), (a)* $C_1 = 5$ *pF; (b)* $C_1 = 10$ *pF.*

2.1.8 NOISE CONSIDERATIONS

The FC family gates are relatively insensitive to noise because of the extra diodes inserted between the input and the transistor base. The diodes raise the voltage level which must be applied at the input before the transistor conducts, i.e. before its output switches to LOW.

Nevertheless, high noise levels in the system may cause undesired switching of gates. We can consider noise as falling into one of two categories, "line" or "ground", and these are now discussed further.

Line noise occurs in the interconnection between the output of a driving gate and the input of the driven gate, the situation being shown in Fig. 2–32. We shall use this to define the noise margin M.

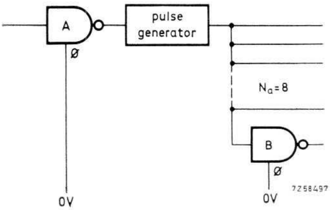


Fig. 2–32. Circuit for testing the influence of line noise (noise occurring on gate interconnections).

Two conditions are possible. The first is that gate A output is LOW and so of course must gate B input. The noise margin M_L is the maximum voltage on the interconnection between A and B that will just not cause gate B to switch to the HIGH state. It is given by:

$$M_L = |V_{GL\mathrm{max}} - V_{QL\mathrm{max}}|,$$

where $V_{GL\mathrm{max}}$ is the maximum LOW input voltage to B,

$V_{QL\mathrm{max}}$ is the maximum LOW output voltage from A.

Substituting worst case values ($V_{GL\mathrm{max}} = 0.8$ V, $V_{QL\mathrm{max}} = 0.4$ V) gives a value of 0.4 V for M_L.

The second condition is that gate A output (and thus gate B input) is HIGH. The noise margin M_H is the maximum voltage on the interconnection between A and B that will just not cause gate B to switch to the LOW state. M_H can be expressed as:

$$M_H = |V_{QH\mathrm{min}} - N_a I_{GH\mathrm{max}} R_{C\mathrm{max}} - V_{GH\mathrm{min}}|,$$

where $V_{QH\min}$ = minimum output HIGH voltage from A (unloaded), 5.3 V
N_a = fan-out (maximum of 8)
$I_{GH\max}$ = maximum input HIGH current of B (diode leakage), 25 μA
$R_{C\max}$ = maximum collector resistance of A, 7 kΩ
$V_{GH\min}$ = minimum input HIGH voltage of B, 2.1 V.
With these values substituted, $M_H = 1.8$ V.

Ground noise is noise appearing in the line connecting the 0 V or ground pin (Ø) of gate A to the common (0 V) point (see Fig. 2–33). The two possibilities are that gate A transistor is either heavily or lightly saturated. When heavily saturated, ground noise has little effect, and cannot drive the transistor out of saturation. In this circumstance the ground noise immunity is equal to the LOW line noise margin M_L.

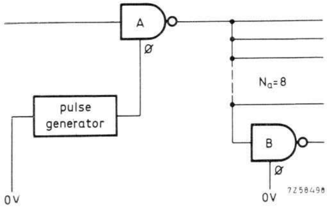


Fig. 2–33. Circuit for testing the influence of ground noise (noise entering the gate via the 0 V or ground connection).

If the transistor is only lightly saturated, ground noise may drive it out of saturation and into the linear amplification region. The transistor then operates as an active device and large voltage changes may appear at the output, expecially at low V_P and low temperatures. It is thus recommended to design logic circuitry in such a way as to minimize the length of ground connection lines.

2.2 Flip-Flops

The flip-flop*, a bistable element, finds application in most logic circuits and indeed plays a predominant role in sequential logic. There are eight

* Officially designated "toggle" or "bi-stable trigger circuit" in England (B.S. 204: 1960, No. 32043). "Flip-flop" was chosen in U.S.A. by the I.R.E. Standards on Computers, 1950.

types in the FC family, distinguished by the type letters FCJ followed by three digits. Section 2.2.1 discusses basic flip-flop design, and sections 2.2.2 to 2.2.5 describe the individual flip-flops in the FC family.

2.2.1 PRINCIPLES OF OPERATION

The S-R Flip-Flop

The simplest way of making the flip-flop is to cross-couple two logic gates, and this has been shown in Fig. 2–34 using two OR gates. The outputs Q_1 and Q_2 are dependent on the inputs S_1 and S_2 in the way described by the function table of Table 2–7.

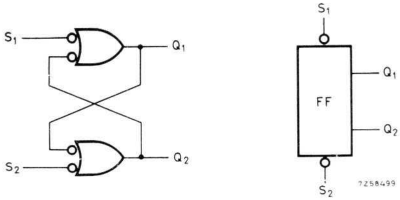


Fig. 2–34. A simple flip-flop formed from two cross-coupled gates, commonly named the "set-reset" or S-R flip-flop.

Table 2–7 S-R Flip-Flop Function Table

S_1	S_2	Q_1	Q_2
L	H	H	L
H	L	L	H
L	L	H	H
H	H	no change	

The outputs can be set in any combination, except double L, by suitable levels at S_1 and S_2. However for binary logic only two combinations are necessary: $Q_1 = \text{L}$, $Q_2 = \text{H}$ and $Q_1 = \text{H}$, $Q_2 = \text{L}$. A convenient property of flip-flops is that if S_1 and S_2 are both brought to HIGH, the outputs do not change state. If however S_1 (say) is then changed from H to L, Q_1 will go H and Q_2 to L, and vice-versa if S_2 is changed. We could consider the $S_1 = S_2 = \text{H}$ condition as the normal, non-activated condition, and changing S_1 to L (making $Q_1 = \text{H}$, $Q_2 = \text{L}$) as a "setting" action. Similarly, changing S_2 to L ($Q_1 = \text{L}$, $Q_2 = \text{H}$) could be considered as a "resetting" action.

If the outputs of this flip-flop circuit are needed to drive the inputs of a second flip-flop and so on, then the contents of the first (whether Q_1 = H, Q_2 = L or Q_1 = L, Q_2 = H) will be entered into the others. To enable other information — any combination of HIGHs and LOWs — to be written into any desired flip-flop, the circuit of Fig. 2–35 could be used instead of Fig. 2–34. This has an extra input T (trigger) which is brought to H only at the instant it is desired to set or reset the circuit.

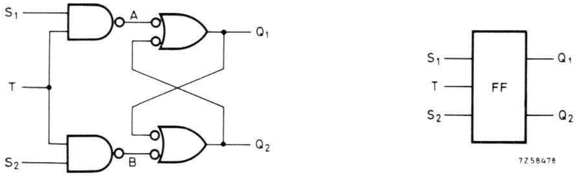


Fig. 2–35. A flip-flop circuit with triggering possibility.

Normally it is left at L. Points A and B, which correspond exactly with inputs S_1 and S_2 in Fig. 2–34 determine the output levels; in Boolean algebra this can be expressed:

$$\overline{T \cdot S_1} = \bar{T} + \bar{S}_1 = A.$$

$$\overline{T \cdot S_2} = \bar{T} + \bar{S}_2 = B.$$

A and B will remain at HIGH as long as T remains at LOW level, no matter what levels S_1 and S_2 are at. Only when T = H can S_1 and S_2 influence the outputs.

In a chain of such flip-flops, assume that the input of the first is (say) S_1 = L, S_2 = H. Then Q_1 = L and Q_2 = H after the trigger pulses, this appears at the inputs S_1 and S_2 of the second flip-flop in the chain. Applying a trigger pulse to this second flip-flop causes its outputs to assume the levels Q_1 = L ,Q_2 = H. In this way a shift register could be made, but requiring the trigger pulse to be delivered separately to each T input.

Fig. 2–37 shows the block diagram of the "master-slave" type of flip-flop that is not afflicted with this drawback*. The trigger pulse at T in Fig. 2–37 activates the master flip-flop M directly and the slave flip-flop S indirectly via the inverter I. As long as there is no trigger pulse (T =

* Introduction of a delay element (D_1, D_2 in Fig. 2–36) aims at the same effect.

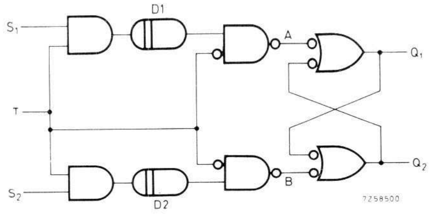


Fig. 2–36. A flip-flop incorporating delay elements D which ensure that the outputs only assume the states dictated by the inputs after the trigger pulse has returned to LOW level.

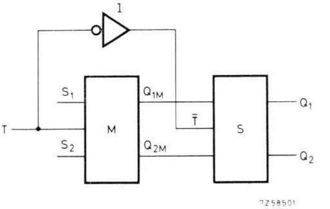


Fig. 2–37. Block diagram of the typical "master-slave" flip-flop.

L) M remains inactive and S_1 and S_2 may change without influencing the outputs Q_{1M} and Q_{2M}. Due to the inverter however, $\bar{T}$ = H and forces Q_1 and Q_2 (slave outputs) to take over the states Q_{1M} and Q_{2M} respectively.

Fig. 2–38 shows an arrangement using the master-slave principle; this is actually the basic circuit of some of the flip-flops in the FC family.

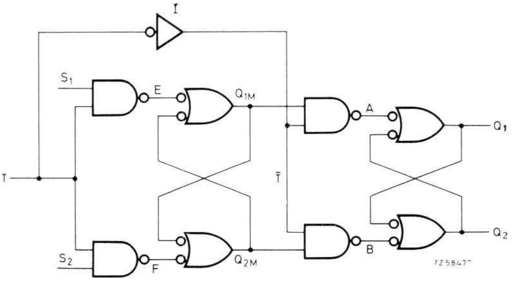


Fig. 2–38. Typical master-slave flip-flop circuit.

The function table of the master is given in Table 2–8.

Table 2–8 Input/Output Function Table for the Master-Slave Flip-Flop

S_1	S_2	Q_{1M}	Q_{2M}
L	H	L	H
H	L	H	L
H	H	H	H
L	L	no change	

The operation of the master-slave flip-flop is now described with the aid of Table 2–9. This table indicates the logic levels present at various points in the circuit at three times, i.e. before, during and after application of a HIGH trigger pulse to T. It is assumed that $S_1 = \bar{S}_2$.

Table 2–9 Complete Function Table of the Master-Slave Flip-Flop

inputs			master				slave			
T	S_1	S_2	E	F	Q_{1M}	Q_{3M}	A	B	Q_1	Q_2
L	S_1	S_2	H	H	L H	H L	$\bar{Q}_{1M}$	$\bar{Q}_{2M}$	Q_{1M}	Q_{2M}
H	S_1	S_2	$\bar{S}_1$	$\bar{S}_2$	S_1*	S_2*	H	H	no change**	
L	S_1	S_2	H	H	S_1	S_2	$\bar{S}_1$	$\bar{S}_2$	S_1	S_2

* see Table 2–8. ** see Table 2–7.

Table 2–9 shows that finally the outputs Q_1 and Q_2 assume the levels of S_1 and S_2 respectively, that is, no inversion takes place. To sum up the operation of this circuit, the binary information at S_1 and S_2 is transferred during the time that input T is at HIGH to the outputs Q_{1M} and Q_{2M}, and when T goes to LOW again the slave outputs Q_1 and Q_2 take over the information.

The "D" Flip-Flop

The circuits discussed so far are commonly termed "*S-R flip-flops*" (having a Set and a Reset terminal). This principle of operation is used again in the so-called "*D* flip-flop" shown in the block diagram of Fig. 2–39. The *FF* block in this diagram is simply the S-R flip-flop of Fig. 2–38. The inverter connected between S_1 and S_2 ensures that $S_1 = \bar{S}_2$, meaning that a HIGH applied to D (= S_1) will set the flip-flop to Q_1 = H, Q_2 = L (after a trigger pulse on T) and a LOW applied to D will reset it to Q_1 = L, Q_2 = H.

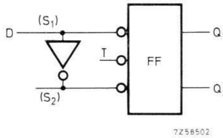


Fig. 2–39. Block diagram of the typical "D" flip-flop.

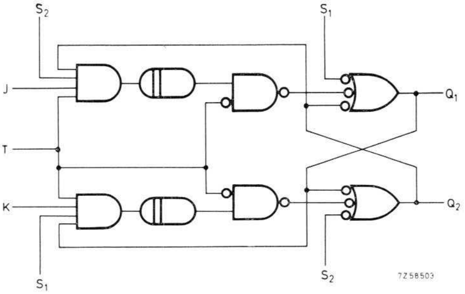


Fig. 2–40. Typical "J-K" flip-flop circuit, based on the flip-flop with delay elements shown in Fig. 2–36 (type FCJ101 uses this configuration).

The J-K Flip-Flop

A disadvantage inherent in S-R flip-flops is that if S_1 and S_2 are both brought to HIGH level then the output levels may not be defined after a change in input levels. Apart from the D flip-flop, a solution is to use "crossed feedback", as shown in the circuit of Fig. 2–41 (modified Fig. 2–38). Two other inputs J and K are provided in this circuit, commonly called a "J-K flip-flop". The original S_1 and S_2 inputs are retained, but now directly connected to master and slave sections, in order to set the circuit in one or other condition before the logic operation is commenced. Normally S_1 and S_2 will be held at HIGH or left floating; bringing S_1 to LOW sets Q_1 = H, Q_2 = L; and bringing S_2 to LOW resets Q_1 = L, Q_2 = H, overriding any other input signals which may be present. We now have three types of input terminals: direct set inputs S_1 and S_2, preparatory inputs J and K, and the trigger input T.

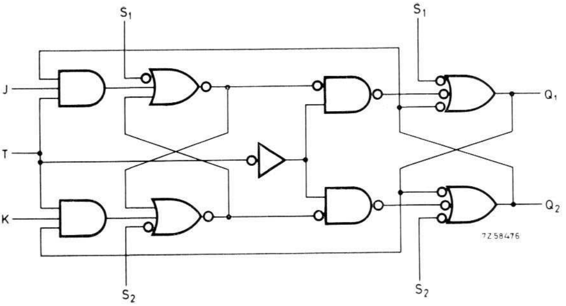


Fig. 2–41. Typical "J-K" flip-flop, based on the "master-slave" flip-flop of Fig. 2–38 (types FCJ121, FCJ131, FCJ191 and FCJ211 use this principle).

The function table of Table 2–10 describes the operation of the circuit of Fig. 2–41.

Table 2–10 Function Table for the Flip-Flop of Fig. 2–41

input levels		output levels after triggering	
J	K	Q_1^{n+1}	Q_2^{n+1}
L	H	L	H
H	L	H	L
H	H	$\overline{Q}_1^{\,n}$	$\overline{Q}_2^{\,n}$
L	L	$Q_1^{\,n}$	$Q_2^{\,n}$

In the table the superscripts $n + 1$ and n denote the post- and pre-triggering states respectively. When $J = K =$ H, the output levels assume the inverse to those prevailing before the trigger pulse. The operation of a *J-K* flip in this situation is often termed a "toggle" action.

The *J-K* flip-flops of Fig. 2–40 (compare Fig. 2–36) and 2–41 are of the "volatile" and "non-volatile" memory types respectively. This will be elucidated further.

In the flip-flop of Fig. 2–40, the "volatile memory" type, the levels of the *J* and *K* inputs may vary and still have no effect on the output up to a certain time before *T* goes to LOW. After this time the outputs will be governed by the inputs, as given in Table 2–10.

In the flip-flop of 2–41 however (the "non-volatile" type), there are two combinations of *J-K* levels which, if applied before and during a certain minimum time before *T* goes to LOW, will cause the outputs not to change state.

These "no-change" combinations are:

$$J = 0 \qquad K = 0$$
$$J = Q_1{}^n, \qquad K = \overline{Q_1{}^n}.$$

$Q_1{}^n$ is the Q_1 output level just before applying the trigger signal T = H. Two other combinations can cause the outputs to change when T goes to LOW; these "change" combinations are:

$$J = 1 \qquad K = 1;$$
$$J = \overline{Q_1{}^n}, \qquad K = Q_1{}^n$$

When any one of the four combinations is present during the whole of the time T is at HIGH, the outputs will assume the states dictated by the combination. If the combination is changed during this time however, the output levels are not so easily predictable. Let us examine what happens in the following case.

Assume that a "no-change" combination is presented to the *J-K* inputs when T goes to HIGH, and that a "change" combination replaces it previous to the critical time before T goes to LOW. The master flip-flop will change state and the slave outputs will follow. If it was indeed the intention to produce a change, then this "last moment" change of *J-K* combination was all that was necessary. If the intention is that no change is to occur however, the "no-change" combination must be applied to the *J-K* inputs previous to a certain minimum time before the trigger signal, i.e. before T goes to HIGH, and remain until a certain minimum time before T returns to LOW.

The *J-K* flip-flops can be further modified so that it can be triggered on the *J-K* inputs with T held at HIGH; the block diagram for this type is given in Fig. 2–42. There are three modes of operation possible: J and K connected and used as a trigger input will cause a toggle operation; bringing J from HIGH to LOW will set the outputs to Q_1 = H, Q_2 = L; and bringing K from HIGH to LOW will reset the outputs to Q_1 = L,

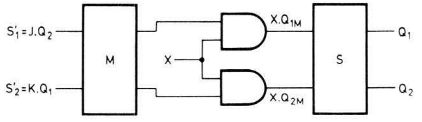


Fig 2–42. Block diagram of special J–K flip-flop. This type can be triggered on the J and K inputs as well as on the normal T input.

$Q_2 = \mathrm{H}$. In the latter two cases, the output levels stated will be attained irrespective of the level of the K or the J input (respectively). The X input in Fig. 2–42 is an internal connection. It is the output of the gate shown in the centre of the circuit diagram of Fig. 2–43, where $X = \overline{J \cdot Q_2} \cdot \overline{K \cdot Q_1}$. Both the master and the slave flip-flops operate according to Table 2–11, S_1' and S_2' in the table being equivalent to $J \cdot Q_2$ and $K \cdot Q_1$ respectively for the master, or $X \cdot Q_{1M}$ and $X \cdot Q_{2M}$ respectively for the slave. The circuit is completely symmetrical, and the following description of operation for the J terminal also applies to the K terminal. Table 2–11 gives the four possibilities for J to go from H to L under the various conditions of K and Q_1.

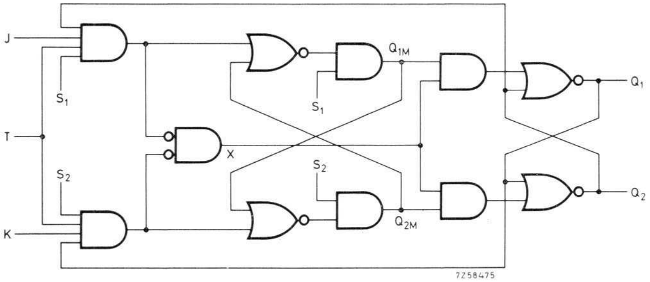


Fig. 2–43. Circuit diagram of the special J–K flip-flop of Fig. 2–42 (types FCJ111 and FCJ201 are based on this configuration).

Table 2–11 Function Table, Flip-Flop of Fig. 2–43

J	K	Q_1	S_1'	S_2'	X	Q_{1M}	Q_{2M}
H	L	L	H	L	L	L	H
L	L	L	L	L	H	*)	*)
H	H	L	H	L	L	L	H
L	H	L	L	L	H	*)	*)
H	L	H	L	L	H	*)	*)
L	L	H	L	L	H	*)	*)
H	H	H	L	H	L	H	L
L	H	H	L	H	L	H	L

*) indeterminate

In the first possibility, J going from H to L will force Q_1 from L to H, because when $J = \mathrm{H}$, the master output is $Q_{1M} = \mathrm{L}$, $Q_{2M} = \mathrm{H}$. This is maintained when $J = \mathrm{L}$, because the master output is insensitive to changes in that position. In addition $X = \mathrm{H}$, enabling the slave to take over the inverse output of the master, i.e. $Q_1 = \mathrm{H}$, $Q_2 = \mathrm{L}$.

In the second possibility, K being at H level has no effect on the operation, and the results are the same.

In the third possibility, K remains at LOW and Q_1 starts at HIGH. J changing from H to L causes no change in the master outputs and therefore no change in the slave, Q_1 thus remaining at HIGH.

The fourth possibility also causes no change in the master. The X signal being at LOW prevents the slave from taking over the inverse information from the master, and so Q_1 does not change level.

Sections 2.2.2 to 2.2.5 give the more important data concerning the actual flip-flop types included in the FC family, based on the types described above.

2.2.2 TYPE DESCRIPTION: FCJ121, FCJ131, FCJ191 and FCJ211

(Dual J-K Master-Slave Flip-Flop)

Each type in this series comprises two independent direct-coupled *J-K* flip-flops operating on the master-slave principle. Operation depends on voltage levels at the inputs only. Set inputs (if applicable) are active at LOW level.

Table 2–12 lists the kind and number of inputs characterizing each of the four types.

Table 2–12

package	*J* inputs		*K* inputs		set inputs (S_1, S_3)		set inputs (S_2, S_4)		*T* inputs	
type	1st FF	2nd FF	1st FF	2nd FF	1st FF	2nd FF	1st FF	2nd FF	1st FF	2nd FF
FCJ121	1	2	1	1	0	0	common		1	1
FCJ131	1	1	1	1	0	0	1	1	1	1
FCJ191	1	2	1	1	1	1	common		1	1
FCJ211	1	1	1	1	1	1	common		common	

Figs 2–46, 2–47, 2–48 and 2–49 give the diagrams relevant to the four types.

Table 2–13 illustrates the output states of the first flip-flop which result upon T going LOW. (Information is transferred to the master from the

J-K inputs when *T* is HIGH.) Set inputs where available must be HIGH or floating. A similar table is valid for the second flip-flop, but where two *J* inputs are available, the effect on the flip-flop input is logical-AND or, symbolically, $J_2 \cdot J_3$.

Table 2–13

J	K	$Q_1(Q_3)$	$Q_2(Q_4)$
L	H	L	H
H	L	H	L
L	L	no change	
H	H	reversed	

The action of the Set (S_1, S_3) and (S_2, S_4) inputs is shown in Table 2–14. Logic levels of the other inputs are immaterial.

Table 2–14a Function Table for Set Inputs of Type FCJ121, FCJ131 Flip-Flops

$S_2(S_4)$	$Q_1(Q_3)$	$Q_2(Q_4)$
L	L	H
H	no change	

Table 2–14b Function Table for Set Inputs of Type FCJ191, FCJ211 Flip-Flops

$S_1(S_3)$	S_2	$Q_1(Q_3)$	$Q_2(Q_4)$
H	L	L	H
L	H	H	L
L	L	H	H
H	H	no change	

2.2.3 TYPE DESCRIPTION: FCJ111 AND FCJ201

(*Single J-K Master-Slave Flip-Flops*)

This type of flip-flop operates on the master-slave principle. Operation depends on voltage levels only, i.e. rise and fall times of all input signals, including the trigger signal, are immaterial. The *J*, *K* and *T* inputs are logically equivalent, allowing the use of *J* and *K* for triggering. The Set inputs, which override the other inputs, are active at LOW.

The FCJ111 has a single J and a single K input, while the FCJ201 is provided with AND input gates giving three J and three K inputs. For the FCJ201, the three J or K gate inputs must all be HIGH before the relevant J or K flip-flop input can be considered as being HIGH.

Figs 2–44*a*, *b* and *c* show the logic, pin layout and numbering, and block diagrams respectively for the FCJ111. Figs 2–45*a*, *b*, *c* show the diagrams for the FCJ201.

Table 2–15 shows how information on the J and K terminals is transferred to the outputs Q_1 and Q_2 when T goes LOW. Set inputs S_1 and S_2 should be HIGH or floating.

Table 2–15 Function Table, Type FCJ111 and FCJ201 Flip-Flops.

J	K	Q_1^{n+1}	Q_2^{n+1}
L	H	L	H
H	L	H	L
L	L	no change	
H	H	reversed	

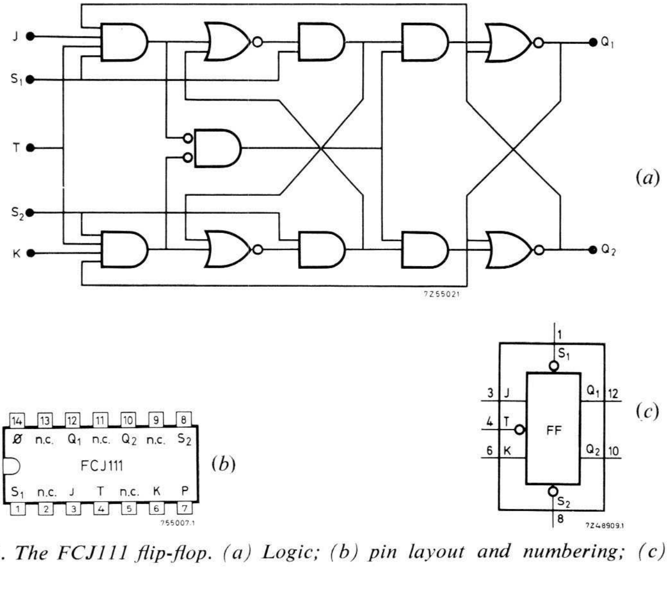


Fig. 2–44. The FCJ111 flip-flop. (a) Logic; (b) pin layout and numbering; (c) block diagram.

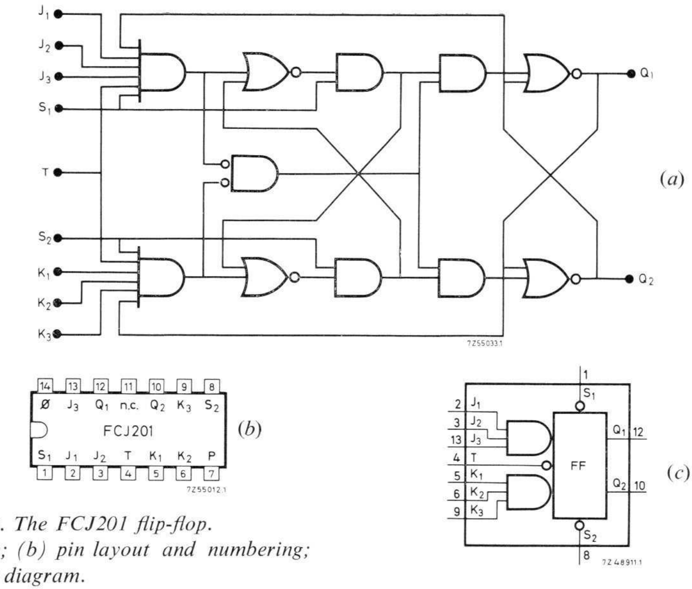


Fig. 2–45. The FCJ201 flip-flop.
(a) Logic; (b) pin layout and numbering;
(c) block diagram.

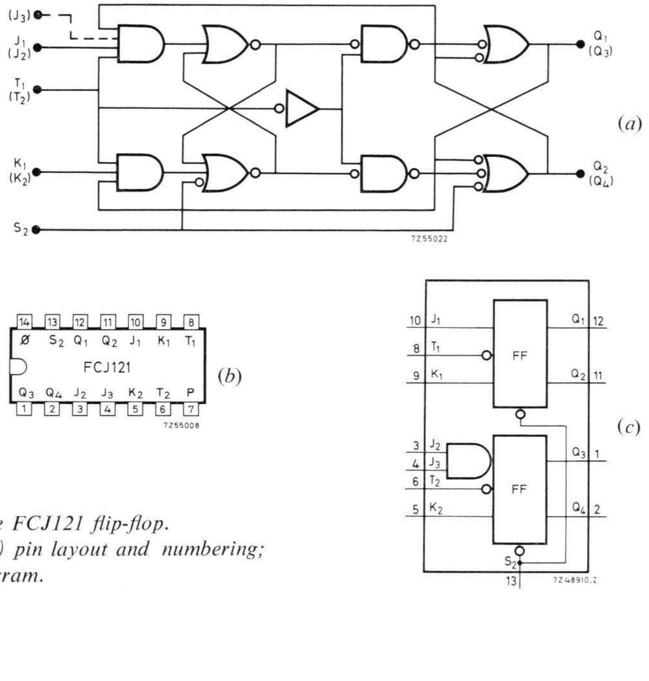


Fig. 2–46. The FCJ121 flip-flop.
(a) Logic; (b) pin layout and numbering;
(c) block diagram.

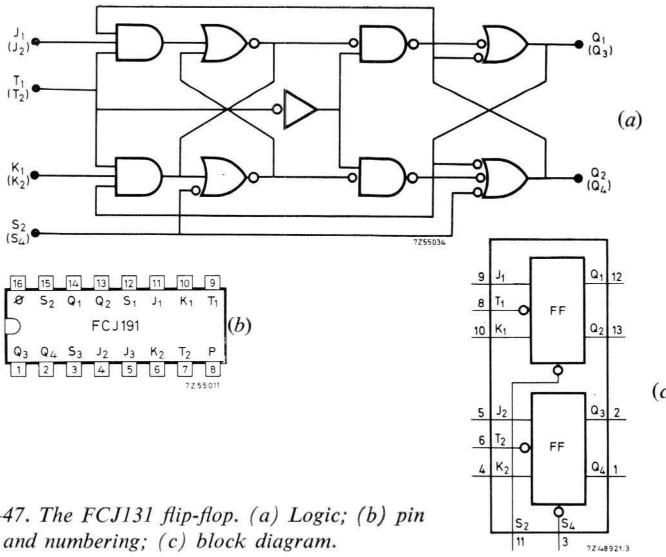


Fig. 2–47. The FCJ131 flip-flop. (a) Logic; (b) pin layout and numbering; (c) block diagram.

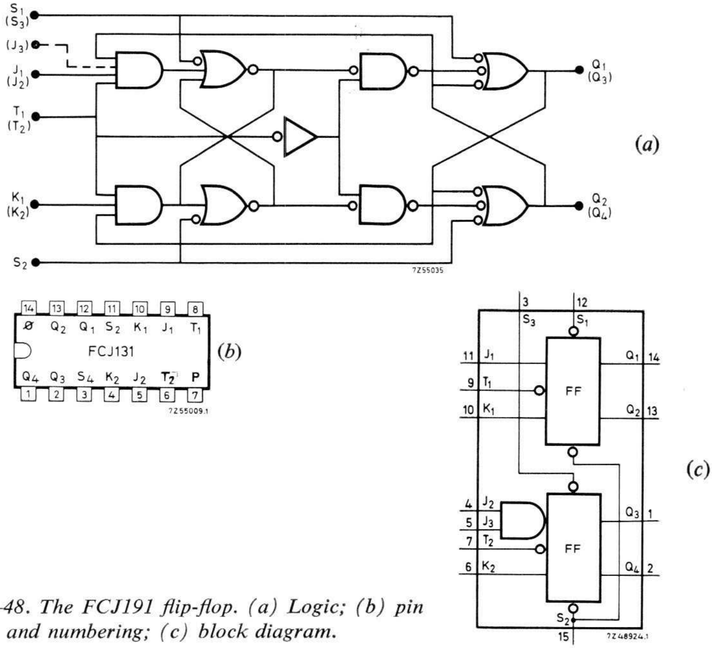


Fig. 2–48. The FCJ191 flip-flop. (a) Logic; (b) pin layout and numbering; (c) block diagram.

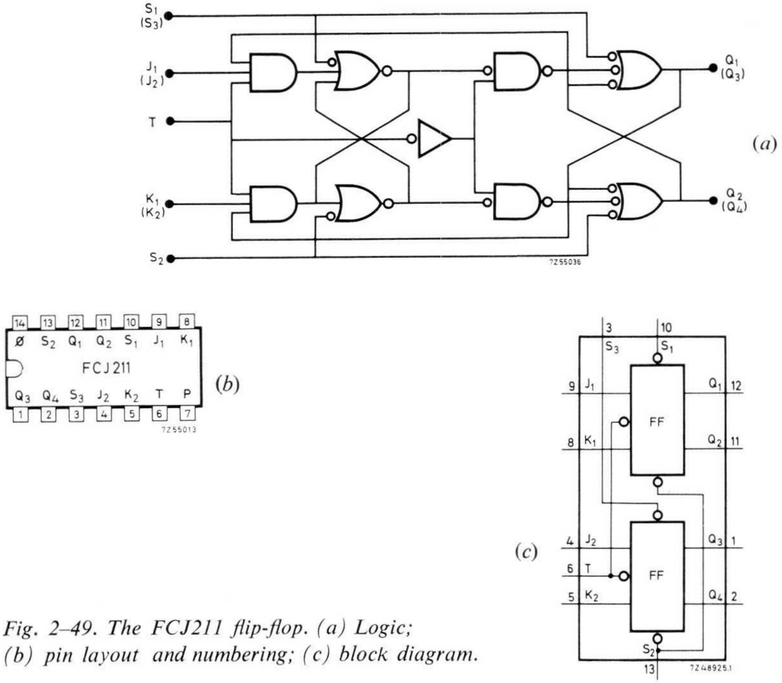


Fig. 2–49. The FCJ211 flip-flop. (a) Logic; (b) pin layout and numbering; (c) block diagram.

2.2.4 TYPE DESCRIPTION: FCJ101

(Single J-K Flip-Flop)

The type FCJ101 is a *J-K* flip-flop which is triggered at the falling edge of a voltage applied to the *T* input. The Set and Reset inputs, which override all other inputs, are active at LOW level. Both flip-flop inputs are preceded by AND gates, providing three *J* and three *K* inputs. Figs. 2–50*a*, *b* and *c* show the logic, pin numbering, and block diagrams for the FCJ101.

Table 2–16 shows the output states which result when *T* goes LOW (information on *J* and *K* applied when *T* was HIGH). S_1 and S_2 should be HIGH or floating. In this table, *J* is the logical AND of $J_1 \cdot J_2 \cdot J_3$ (similarly $K = K_1 \cdot K_2 \cdot K_3$ in Boolean algebra).

Table 2–16

J	K	Q_1^{n+1}	Q_2^{n+1}
L	H	L	H
H	L	H	L
L	L	no change	
H	H	reversed	

The setting and resetting action of S_1 and S_2 respectively is illustrated in Table 2–17. Logic levels of the other inputs are immaterial.

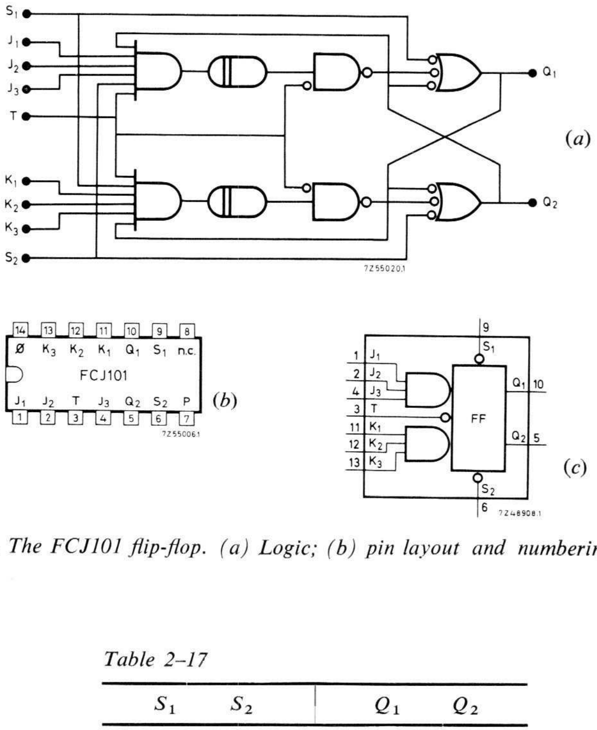


Fig. 2–50. The FCJ101 flip-flop. (a) Logic; (b) pin layout and numbering; (c) block diagram.

Table 2–17

S_1	S_2	Q_1	Q_2
H	L	L	H
L	H	H	L
L	L	H	H
H	H	no change	

2.2.5 TYPE DESCRIPTION: FCJ221

(*Quadruple Latch Flip-Flop*)

The FCJ221 package contains four flip-flops which are triggered by a common T input. One input (D) per flip-flop is available. The mode of operation is as follows, taking the top flip-flop as an example. When T goes HIGH, the information present at D_1 just before the time of transition is retained at Q_2 (and the inverse at Q_1) until T goes LOW again. D cannot influence the outputs when T is HIGH. As long as T is LOW, output Q_2 will follow any level changes at input D_1. Operation of the other three flip-flops is similar to the above; note that Trigger (T) and Set (S_2) signals are common to all. Resetting can be done by applying a LOW signal to input S_2. The principle of operation differs from that discussed in section 2.2.1.

Figs 2–51*a* and *b* show the relevant diagrams for the FCJ221.

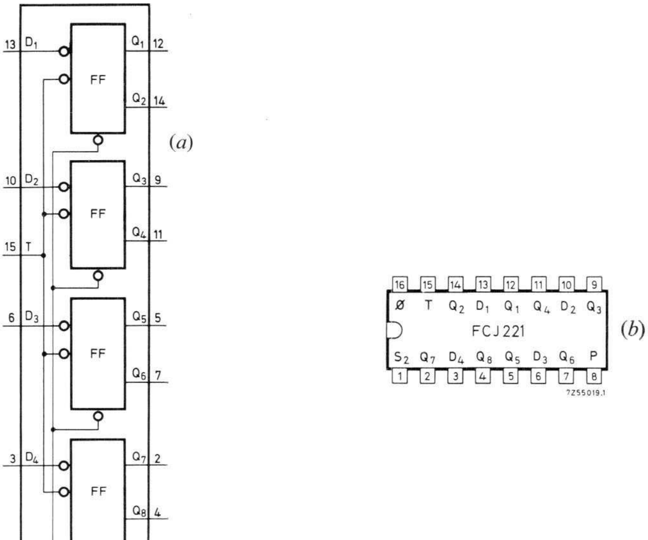


Fig. 2–51. The FCJ221 flip-flop. (a) Block diagram; (b) pin layout and numbering.

2.3 Miscellaneous Types

In this section are grouped the members of the FC family of integrated circuits which fall under neither “gates” nor “flip-flops”. There are seven such members:

— the 5-bit Comparator, FCH281;
— the 10-bit Parity Checker, FCH291;
— the Decoder Circuit, FCH301;
— the Asynchronous Decade, FCJ141;
— the Level Detector, FCL101;
— the Numerical Indicator Tube Driver, FCL111;
— the Monostable Multivibrator, FCK101.

The seven circuits are discussed further in the following sections.

2.3.1 THE 5-BIT COMPARATOR, FCH281

Figs 2–52*a* and *b* show the relevant diagrams respectively. The FCH281 gives a HIGH output signal if two 5-bit binary words presented to its inputs are not equal. It can thus be used for checking errors made when punching data into tape, for example.

Corresponding bits of the same significance in each word are presented to the inputs which each Exclusive-OR gate has, there being five such Exclusive-OR gates. The outputs are applied to an OR-gate, whose output will only become HIGH if the inputs to one or more Exclusive-OR gates are unequal. A final AND gate allows the result to appear at the output terminal of the circuit after application of a strobing signal to a second input of this AND gate.

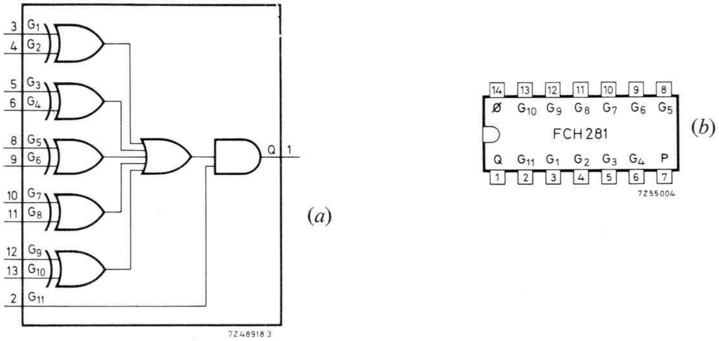


Fig. 2–52. The 5-bit Comparator, FCH281. (a) Logic; (b) pin layout and numbering.

2.3.2 THE 10-BIT PARITY CHECKER, FCH291

The diagrams are shown in Figs 2–53*a* and *b* respectively.

The function of the FCH291 is to determine whether the number of "1"s in a 10-bit word is odd or even. If odd, the output will be at HIGH level if the other input terminal to the AND gate is at HIGH. The circuit is useful in the detection of errors in transmitted data.

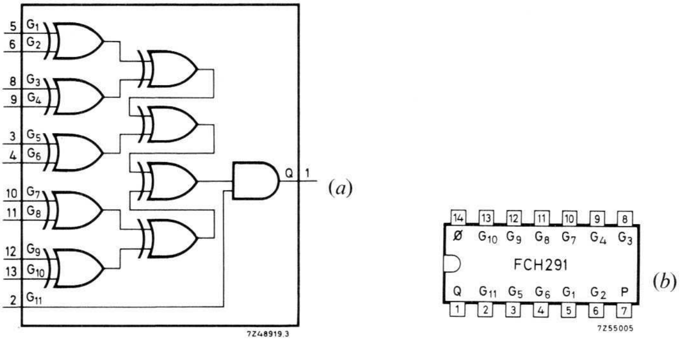


Fig. 2–53. The 10-bit Parity Checker, FCH291. (a) Logic; (b) pin layout and numbering.

2.3.3 THE DECODER CIRCUIT, FCH301

This is a fast binary-to-decimal decoder, formed from 18 gate and inverter functions. Its diagrams are shown in Figs 2–54*a* and *b*.

The decoder accepts 1-2-4-8 coded input combinations. Only the decoded output is at LOW, the other nine remaining at HIGH. If an input combination is not included in the 1-2-4-8 code, all outputs remain at HIGH.

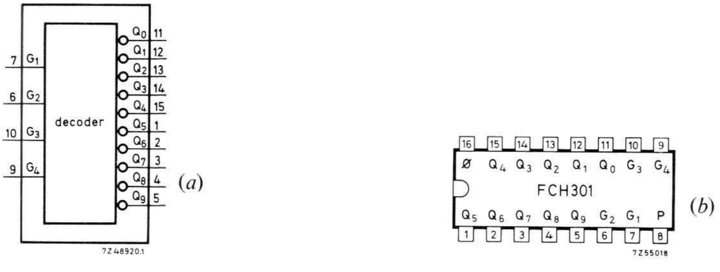


Fig. 2–54. The Decoder Circuit, FCH301. (a) Block diagram; (b) pin layout and numbering.

2.3.4 THE ASYNCHRONOUS DECADE, FCJ141

Figs 2–55*a* and *b* depict the logic diagram and terminal numbering respectively. The FCJ141 has a 1-2-4-8 coded output.

The counter is designed so that it will return to a legitimate state in a maximum of two clock pulses after having been forced by interference into an illegitimate combination*. The counter proceeds as follows. If forced to combination 10, it will go to 11 on the next clock pulse and to 4 on the next. If forced to 12, it will go to 13, then 4. If forced to 14, it will go to 15, then 0.

Maximum clock rate for the FCJ141 is 5 MHz.

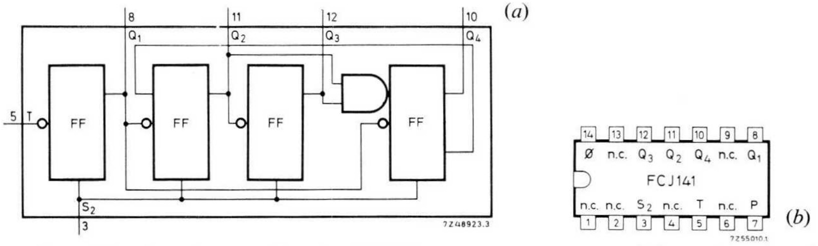


Fig. 2–55. The Asynchronous Decade, FCJ141. (a) Block diagram; (b) pin layout and numbering.

2.3.5 THE LEVEL DETECTOR, FCL101

The FCL101 circuit diagram and terminal numbering are shown in Fig. 2–56*a* and *b*. The FCL101 is a non-inverting Schmitt trigger circuit which changes state abruptly whenever the input signal exceeds specific d.c. triggering levels. These triggering levels can be adjusted externally either by choice of suitable resistor or, if very low hysteresis levels are required, by a voltage reference diode. Applications for the FCL101 are in the discrimination, restoration and shifting of signal levels, and the squaring of waveforms. A brief description of operation is given below.

The circuit of Fig. 2–56*a* shows the integrated circuit itself. Two external resistors are, however, necessary with the circuit. One resistor must be connected between pins 8 and 4 (output and common respectively); this adjusts the output level to that required for the following circuit elements. The other is to be connected between pins 10 and 4 (input and output

* See section 3.1.2.

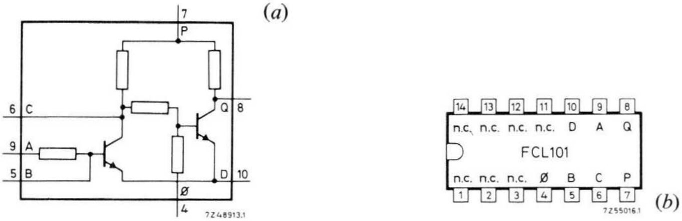


Fig. 2–56. The Level Detector, FCL101. (a) Circuit; (b) pin layout and numbering.

transistor emitters and common respectively); this governs the amount of positive feedback. Because of this positive feedback the circuit displays negative input resistance for certain parts of the input characteristic, as shown in Fig. 2–57. Hysteresis in the output is also strongly affected by the value of feedback resistance, as shown in Fig. 2–58. The hysteresis at D, ΔV_{Et}, can be expressed approximately as:

$$\Delta V_{Et} \approx AR_{fb} - BR^2{}_{fb}, \text{ (eq. 2.3.5)}$$

where R_{fb} is the feedback resistance, and A and B are constants dependent on the particular circuit.

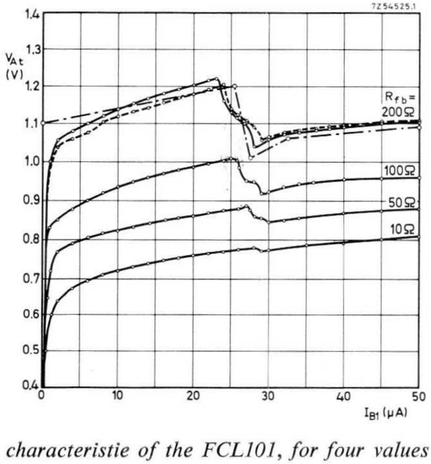


Fig. 2–57. The input characteristie of the FCL101, for four values of feedback resistor R_{fb} connected between pins 10 and 4. (– . – . –. calculated curve).

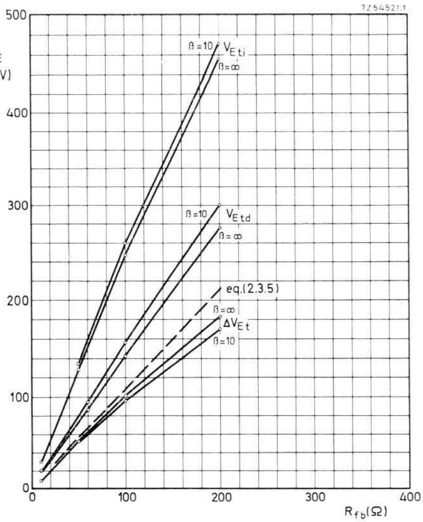


Fig. 2–58. Hysteresis ΔV_{Et} as a function of feedback resistor R_{fb}.

Fig. 2–59 shows the input/output characteristic. Further information about the FCL101 may be found in our publication "Hysteresis and Negative Feedback Resistance in the Schmitt Trigger" (ordering code 9399 304 84401), and "Considerations on Schmitt Trigger Level Detectors" (ordering code 9399 304 84201).

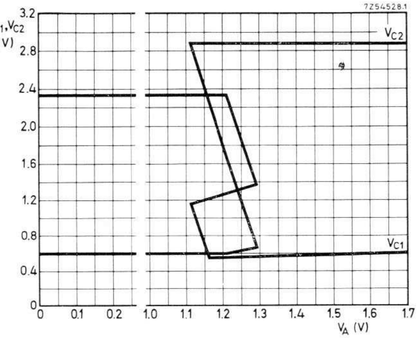


Fig. 2–59. Input-output characteristic of the FCL101 (calculated curves).

2.3.6 THE NUMERICAL INDICATOR TUBE DECODER/DRIVER, FCL111.

The FCL111 contains a decoder which accepts NBCD, Aiken or Excess-three coded signals, followed by ten output driver stages which can be used for either normal or dynamic drive of numerical indicator tubes such as the ZM1000, ZM1020 or ZM1080. The ZM1200 (the 14-digit "Pandicon" tube) can only be driven dynamically, i.e. the anodes of the tube are scanned, and the cathodes are paralled, being driven by one set of drivers. Fig. 2–60*a* and *b* show the circuit and pin numbering of the FCL111; for information on application of the FCL111 please refer to section 6.3.

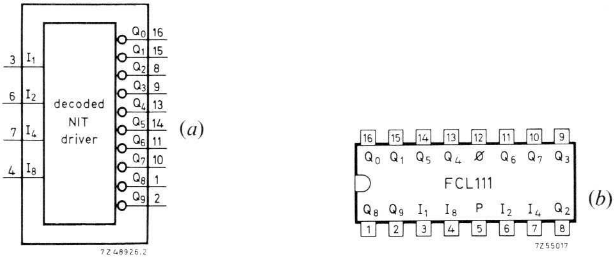


Fig. 2–60. The Numerical Indicator Tube Decoder/Driver FCL111. (a) Block diagram; (b) pin layout and numbering.

2.3.7 THE MONOSTABLE MULTIVIBRATORS, FCK101 AND FCK101Q.

The FCK101 and FCK101Q comprise a threshold-triggered monostable circuit and a gate. Each is capable of producing either HIGH or LOW level pulses, whose width is adjustable over a wide range by means of an external resistor and capacitor. The gate in the FCK101Q is simply an inverter; in the FCK101 however it is an AND gate, thus providing a conditional triggering facility for this type. The logic and terminal numbering diagrams for the FCK101Q are given in Figs 2–61*a* and *b*; those for the FCK101 are given in Figs 2–62*a* and *b*.

The Data Sheets state that for ringing-free operation, the duration t_2 of the input trigger pulse to the multivibrator must not exceed the duration t_1 of the output signal. This is however a general condition: ringing-free operation can also be achieved if t_1 is made less than 100 μs. Alternatively, the interface circuit of Fig. 2–63 could be used, if t_1 is greater than 100 μs and $t_1 < t_2$. The rate of fall of the input pulse is assumed to be less than 100 ns/V.

Not withstanding the above provisions, it is advisable to keep $t_1 > t_2$ if possible.

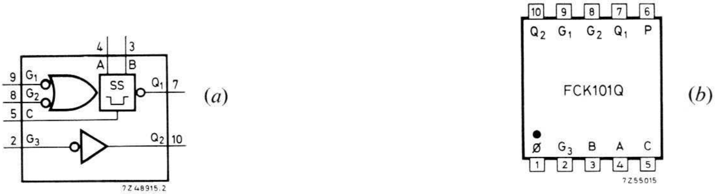


Fig. 2-61. The Monostable Multivibrator, FCK101Q (without extra gate facility). (a) Block diagram; (b) pin layout and numbering.

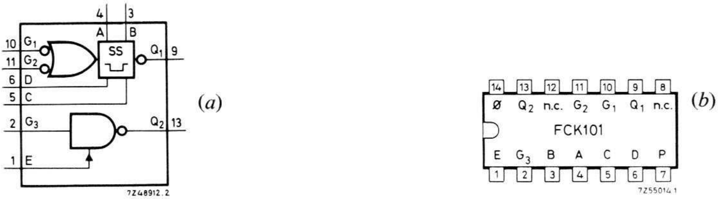


Fig. 2-62. The Monostable Multivibrator, FCK101 (with extra gate facility). (a) Block diagram; (b) pin layout and numbering.

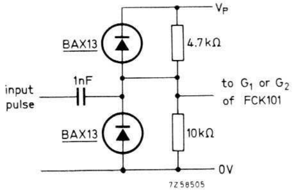


Fig. 2-63. Interface circuit between trigger pulse generator and terminals G_1 or G_2 of FCK101, to be used if the time that Q_1 is HIGH is less than the LOW duration of the G_1 or G_2 trigger signal, but greater than 100 μs.

3 Applications in Sequential Logic

Sequential logic circuits are defined as those circuits in which time plays a part in the logic process. Sequential logic circuits usually incorporate memory elements, so that the output is dependent not only on the input signals present at any moment, but also on the previous state of those signals. (The output of a combinational logic circuit, on the other hand, depends completely on the present state of the inputs.) Sections 3.2, 3.3 and 3.4 discuss the three most important types of sequential logic: scalers, counters, and shift registers, and section 3.1 below describes design methods for scalers and counters.

The sequential logic circuits described here use *J-K* flip-flops as memory elements; details about the flip-flops will be found in section 2.2.

3.1 Design of Scalers and Counters

3.1.1 INTRODUCTION

The aim of the design is to find an efficient and economical way of connecting the flip-flop outputs to the appropriate "preparation" (*J,K*) inputs, in such a way that at each trigger pulse the outputs change to the combination which uniquely identifies that pulse.

A *scaler* (alternatively named *divider* or *divide-by-n* circuit) simply divides incoming pulses by some integer (called the scaling factor, *n*). For every *n* input pulses, one output pulse is delivered, the time of occurrence of which relative to the train of input pulses being usually, but not always, a matter of indifference.

A *counter* discriminates between successive input pulses and delivers an output which uniquely identifies the sequential order of each pulse according to a prescribed numbering system. For each of every *n* input pulses, an output signal is produced indicative of the numerical order of occurrence of the pulse, as counted to the base *n*. The counter itself is usually supplemented by a decoder.

3.1.2 PROCEDURE

The design method described below will allow a scaler or counter to be designed having the least possible number of flip-flops. The method applies to both synchronous (*all* flip-flops triggered by an external clock pulse source) and asynchronous circuits (*not* all flip-flops so triggered). Karnaugh maps are used to determine the interconnections to be made between flip-flops. Some designs will require extra gates, dependent on the type of flip-flops used.*

Step 1. Establish the number of flip-flops.

For a scaler or counter of n, at least N flip-flops will be required. The relationship is:

$$2^{N-1} < n \leqslant 2^{N}.$$

For example, a decade counter will require four flip-flops, because

$$2^3 < 10 < 2^4.$$

Step 2. Establish the truth table.

The output of each flip-flop may be in either the HIGH or the LOW state, so that N flip-flops have 2^N output combinations. For a decade counter of course only ten of these combinations are valid (or "legitimate"); the remaining six are "illegitimate"** combinations. All combinations can be recorded in a truth table, and the decade counter truth table is included in Table 3–1.

The truth table gives the logic level at the non-inverted output of each of the four flip-flops A, B, C, D which, for the 1-2-4-8 code, identifies the correct decimal number (binary coded decimal or B.C.D equivalent) for any given number of input trigger pulses. Starting from combination 0, the first nine input pulses in this case must bring the counter to combination 9; the tenth pulse must give combination 0 and so on.

Step 3. Establish the "change function" table.

At any combination p, the state of a flip-flop in the counter upon arrival of another trigger pulse (i.e. at combination $p + 1$) must either:

— change, or
— not change.

If a change is to occur, a "1" should be placed in the combination p row for that flip-flop; if no change, a "0". In addition, an "X" can be

* Types FCJ111 and FCJ201, however, may be triggered on the J-K inputs as well as the T-input, which can mean a saving of extra gates.

** Sometimes called "redundant states".

Table 3–1. Truth Table and Change Function Table for a Synchronous 1-2-4-8 Decade Counter

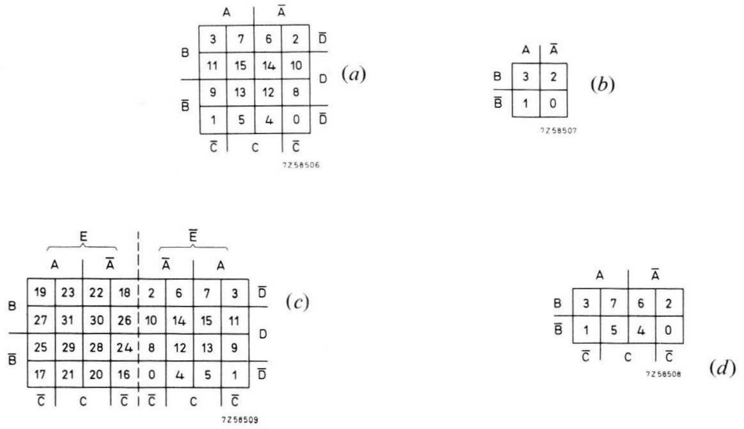
		truth table				change function table			
combination	B.C.D. equivalent	Q_A of flip-flop *A*	Q_B of flip-flop *B*	Q_C of flip-flop *C*	Q_D of flip-flop *D*	flip-flop *A*	flip-flop *B*	flip-flop *C*	flip-flop *D*
0	0	0	0	0	0	1	0	0	0
1	1	1	0	0	0	1	1	0	0
2	2	0	1	0	0	1	0	0	0
3	3	1	1	0	0	1	1	1	0
4	4	0	0	1	0	1	0	0	0
5	5	1	0	1	0	1	1	0	0
6	6	0	1	1	0	1	0	0	0
7	7	1	1	1	0	1	1	1	1
8	8	0	0	0	1	1	0	0	0
9	9	1	0	0	1	1	0	0	1
10	illegitimatie	0	1	0	1	*X*	*X*	*X*	*X*
11		1	1	0	1	*X*	*X*	*X*	*X*
12		0	0	1	1	*X*	*X*	*X*	*X*
13		1	0	1	1	*X*	*X*	*X*	*X*
14		0	1	1	1	*X*	*X*	*X*	*X*
15		1	1	1	1	*X*	*X*	*X*	*X*

placed in all illegitimate combination rows, if the illegitimate combinations are unspecified in the design (see section 3.1.3). An "*X*" can subsequently be treated either as "1" or "0", whichever is most convenient. The illegitimate rows (combinations 10-15) do not concern us, so an "*X*" is placed there. Similarly, the flip-flop *C* column receives a "1" in the rows corresponding to combinations 3 and 7. No change in the state of flip-flop *C* occurs in the other legitimate combinations, and to note this, zeroes are used. Flip-flop *B* changes state four times, and flip-flop *D* two times, as can be seen from the table.

Step 4. Organize the information.

Although we now have enough information to enable the flip-flop inputs and outputs to be interconnected to give the required result, organization of the data into meaningful patterns by means of Karnaugh maps can save time and perhaps result in less interconnections. A Karnaugh map is simply a chart, used here to express the change functions, legitimate and illegitimate, in an easily assimilated way. Although we are most interested in organizing the changes, let us first place the combination numbers in a Karnaugh map. This is done in Fig. 3–1*a* for a counter with four flip-flops.

The variables Q_A, Q_B etc. around the outside of the map are the flip-flop outputs which are at HIGH level. The numbers in the chart are the combination numbers; thus the number "2" is placed in the top right-hand corner because that point is defined as "$\bar{Q}_A$ and Q_B and $\bar{Q}_C$ and $\bar{Q}_D$" by the chart. (Refer to the truth table, Table 3–1.) Maps for counters having two, three and five flip-flops are also given in Fig. 3–1. (The outputs are simply designated by letters.)



(*a*)

	A		$\bar{A}$		
B	3	7	6	2	$\bar{D}$
	11	15	14	10	D
$\bar{B}$	9	13	12	8	
	1	5	4	0	$\bar{D}$
	$\bar{C}$	C		$\bar{C}$	

(*b*)

	A	$\bar{A}$
B	3	2
$\bar{B}$	1	0

(*c*)

	E				$\bar{E}$				
	A		$\bar{A}$		$\bar{A}$		A		
B	19	23	22	18	2	6	7	3	$\bar{D}$
	27	31	30	26	10	14	15	11	D
$\bar{B}$	25	29	28	24	8	12	13	9	
	17	21	20	16	0	4	5	1	$\bar{D}$
	$\bar{C}$	C		$\bar{C}$	$\bar{C}$	C		$\bar{C}$	

(*d*)

	A		$\bar{A}$	
B	3	7	6	2
$\bar{B}$	1	5	4	0
	$\bar{C}$	C		$\bar{C}$

Fig. 3–1. Karnaugh maps showing the combination numbers for counters with (a) four; (b) two; (c) three; and (d) five flip-flops.

Knowing the positions of each combination on the Karnaugh map, it is now an easy matter to replace a combination number with a sign indicating whether or not a change of state is to occur for a flip-flop at that combination. One Karnaugh map is required for each flip-flop. Taking flip-flop *B* of the decade counter as an example, the Karnaugh map would be built up as follows.

Where a change of state must occur, a "1" is entered. For instance, after reaching combination 5 ($A \cdot \bar{B} \cdot C \cdot \bar{D}$), flip-flop *B* must change state:

a "1" is entered in the appropriate place (see Fig. 3–2*a*). The next combination, 6, does not require that flip-flop *B* change state upon the subsequent trigger pulse: a zero is entered in the square represented by $\bar{A}\cdot B\cdot C\cdot \bar{D}$ (see Fig. 3–2*b*). Combinations 10 to 15 are illegitimate; subject to later considerations, whether or not a change of state is to occur is immaterial. We identify these combinations with an "*X*" (see Fig. 3–2*c*). The complete map for the change functions for flip-flop *B* is given in Fig. 3–2*d*.

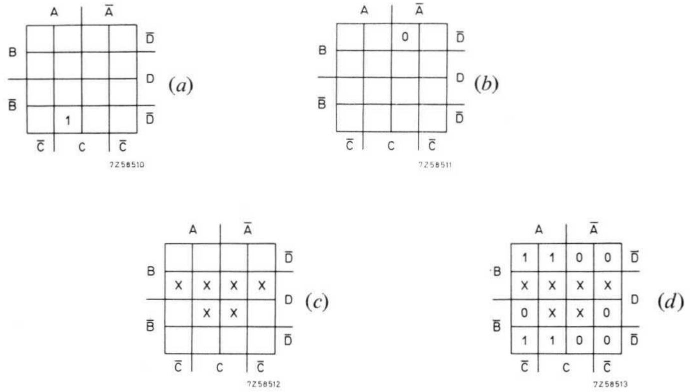


Fig. 3–2. Karnaugh "change function" maps for flip-flop B of a 4 flip-flop decade counter, showing combinations (a) 5; (b) 6; (e) 10, 11, 12, 13, 14, 15; (d) all combinations.

Step 5. Interpret the information.

The Karnaugh change function maps must now reveal the flip-flop interconnections. Before proceeding with the example, it might be wise to review the characteristics of the *J-K* type of flip-flop (see Fig. 3–3).

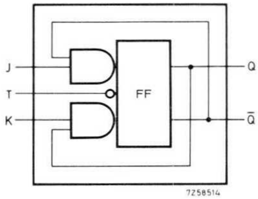


Fig. 3–3. Simplified representation of a J–K flip-flop. Note the internal connections $\overline{Q}$–J and Q–K to the input AND *gates.*

In this type the K input is combined in an AND function with Q (normal) output and the J input is internally AND-gated with $\bar{Q}$ (inverted) output. To change the state of the flip-flop, a trigger pulse must be applied to input T, with either the J or the K input being held at HIGH level. If a change of the state from Q = HIGH, $\bar{Q}$ = LOW to Q = LOW, $\bar{Q}$ = HIGH is required, input K must be held at HIGH level (J being at either H or L level); if a change from Q = LOW, $\bar{Q}$ = HIGH to Q = HIGH, $\bar{Q}$ = LOW is required, input J must be held at HIGH level (K level then being immaterial). Thus, from Fig. 3–2*d*, we see that the K input of flip-flop B *must* be at HIGH for combinations 3 ($A{\cdot}B{\cdot}\bar{C}{\cdot}\bar{D}$) and 7 ($A{\cdot}B{\cdot}C{\cdot}\bar{D}$). That is to say, input K_B of flip-flop B must be connected via an AND–OR gate combination to output A of flip-flop A, B of flip flop B, C and $\bar{C}$ of flip-flop C, and output $\bar{D}$ of flip-flop D. As mentioned above however, the B connection already exists inside the flip-flop so that we can ignore this term. In addition, both C and $\bar{C}$ are mentioned, and as the logical result of a variable and its inverse in an OR gate is "1", we can ignore C and $\bar{C}$ as well. In Boolean algebra,

$$A{\cdot}\bar{C}{\cdot}\bar{D} + A{\cdot}C{\cdot}\bar{D} = A{\cdot}\bar{D}\,(C + \bar{C}) = A{\cdot}\bar{D}.$$

Both combinations thus now reduce to $A{\cdot}\bar{D}$, to be realised by a single AND gate.

A further reduction of the connections to be made to K, to a single connection, can be done. Note that the adjacent combinations 11 and 15 contain X. This means that it is irrelevant (so far) if a change of state is to occur or not. Thus we can temporarily place a 1 in these two squares, i.e. $A{\cdot}B{\cdot}\bar{C}{\cdot}D$ and $A{\cdot}B{\cdot}C{\cdot}D$. Just as noted above, C can again be dropped, B can be neglected, leaving the Boolean funtion AD to be OR'ed with $A\bar{D}$. Finally the only connection to be made to K of flip-flop B is output A from flip-flop A. Note that this cannot be done for input J, because of the zero in the adjacent square (square $A{\cdot}\bar{B}{\cdot}\bar{C}{\cdot}D$).

The other flip-flops, each with its own Karnaugh map can be investigated in the same way. Their Karnaugh maps are presented in Fig. 3–4.

In particular, both inputs to flip-flop A are to be permanently connected to HIGH level. The reason for this is that all terms except A appear in both the positive and the inverse form, so that they can be ignored, and A and $\bar{A}$ are internally connected to K and J respectively. Thus J and K may remain constantly at "1" level. Flip-flop A as a result changes state upon every application of a trigger pulse to input T, as required by the truth table in Table 3–1.

The resulting interconnections between the flip-flops for the 1-2-4-8 code synchronous decade counter are presented in Fig. 3–5.

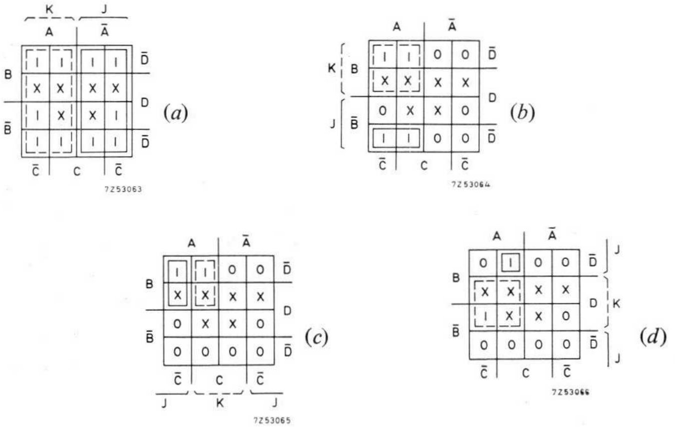


Fig. 3–4. Karnaugh "change function" maps for the decade counter having four flip-flops. (a) Flip-flop A (giving $J_A = 1$, $K_A = 1$); (b) flip-flop B ($J_B = A \cdot \overline{D}$, $K_B = A$); (c) flip-flop C ($J_C = A \cdot B$, $K_C = A \cdot B$); (d) flip-flop D ($J_D = A \cdot B \cdot C$, $K_D = A$).

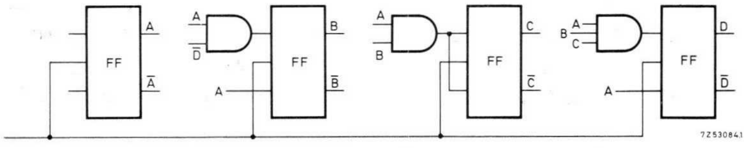
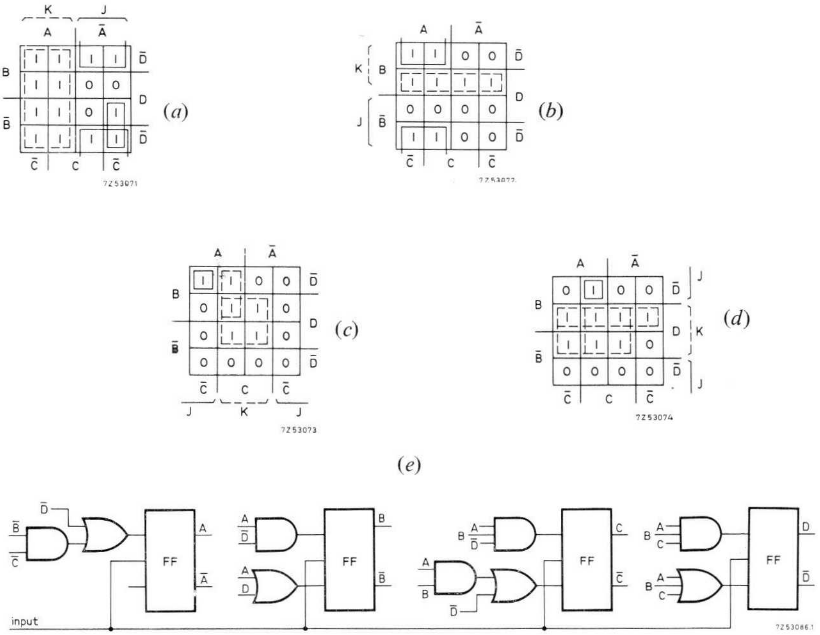


Fig. 3–5. The complete circuit of the synchronous 1248 code decade "up" counter described in the text. The illegitimate states are unspecified. See Fig. 3–4 for flip-flop input conditions.

3.1.3 FURTHER CONSIDERATIONS

Illegitimate Combinations

Although theoretically a counter will never enter one of the illegitimate states, this may in fact happen, for example as the result of external interference pulses. If the consequences of this are serious enough, a counter can be designed which will switch to zero, or to any legitimate state, immediately after entering an illegitimate state. This will ensure that the counter will not remain locked in the illegitimate state or states. Switching to any legitimate combination is easier to arrange and requires less additional circuitry than switching to zero. The two solutions are given in Figs 3–6 and 3–7. The legitimate combination that the counter of Fig. 3–7 must step to (from an illegitimate combination) should be chosen to give the fewest extra interconnections and gates. These counters do however require some trial and error before the best solution can be arrived at.



(a)

	A $\bar{C}$	A C	$\bar{A}$ C	$\bar{A}$ $\bar{C}$
B $\bar{D}$	1	1	1	1
B D	1	1	0	0
$\bar{B}$ D	1	1	0	1
$\bar{B}$ $\bar{D}$	1	1	1	1

(b)

	A $\bar{C}$	A C	$\bar{A}$ C	$\bar{A}$ $\bar{C}$
B $\bar{D}$	1	1	0	0
B D	1	1	1	1
$\bar{B}$ D	0	0	0	0
$\bar{B}$ $\bar{D}$	1	1	0	0

(c)

	A $\bar{C}$	A C	$\bar{A}$ C	$\bar{A}$ $\bar{C}$
B $\bar{D}$	1	1	0	0
B D	0	1	1	0
$\bar{B}$ D	0	1	1	0
$\bar{B}$ $\bar{D}$	0	0	0	0

(d)

	A $\bar{C}$	A C	$\bar{A}$ C	$\bar{A}$ $\bar{C}$
B $\bar{D}$	0	1	0	0
B D	1	1	1	1
$\bar{B}$ D	1	1	1	0
$\bar{B}$ $\bar{D}$	0	0	0	0

(e)

Fig. 3–6. The complete design of a synchronous 1248 code decade "up" counter, with the property that the counter will always step from any illegitimate state to zero. The resulting input conditions to the flip-flops are as follows.

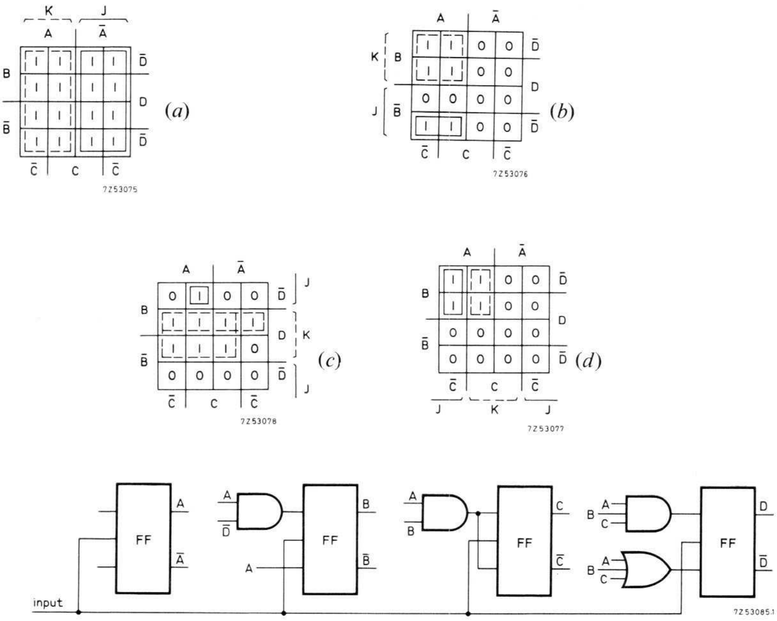
(a) Flip-flop A: $J_A = \overline{B}.\overline{C} + \overline{D}$, $K_A = 1$.

(b) Flip-flop B: $J_B = A{\cdot}\overline{D}$, $K_B = A + D$.

(c) Flip-flop C: $J_C = A{\cdot}B{\cdot}\overline{D}$, $K_C = A{\cdot}B + \overline{D}$.

(d) Flip-flop D: $J_D = A{\cdot}B{\cdot}C$, $K_D = \mathrm{A} + B + C$.

(e) The circuit.



(a)

	A	A	Ā	Ā	
B	1	1	1	1	D̄
B	1	1	1	1	D
B̄	1	1	1	1	D
B̄	1	1	1	1	D̄
	C̄	C	C	C̄	

(b)

	A	A	Ā	Ā	
B	1	1	0	0	D̄
B	1	1	0	0	D
B̄	0	0	0	0	D
B̄	1	1	0	0	D̄
	C̄	C	C	C̄	

(c)

	A	A	Ā	Ā	
B	0	1	0	0	D̄
B	1	1	1	1	D
B̄	1	1	1	0	D
B̄	0	0	0	0	D̄
	C̄	C	C	C̄	

(d)

	A	A	Ā	Ā	
B	1	1	0	0	D̄
B	1	1	0	0	D
B̄	0	0	0	0	D
B̄	0	0	0	0	D̄
	C̄	C	C	C̄	

Fig. 3–7. The complete design of a synchronous 1248 code decade "up" counter, with the .property that the counter will always step from any illegitimate state to a legitimate one The resulting input conditions to the flip-flops are as follows.

(a) Flip-flop A: $J_A = 1$, $K_A = 1$.

(b) Flip-flop B: $J_B = A{\cdot}\overline{D}$, $K_B = A$.

(c) Flip-flop C: $J_C = A{\cdot}B$, $K_C = A{\cdot}B$.

(d) Flip-flop D: $J_D = A{\cdot}B{\cdot}C$, $K_D = A + B + C{\cdot}$.

(e) The circuit.

Asynchronous Counters

Unlike synchronous counters, the flip-flops of asynchronous counters are not triggered from a common trigger clock pulse but from suitably chosen outputs of the other flip-flops in the counter. The first flip-flop must still be driven externally, of course. Designing an asynchronous counter involves determining for each flip-flop which of the others is to provide the trigger signal, in addition to determining the connections to be made to its *J-K* inputs.

Fig. 3–8 shows the design of an asynchronous 1-2-4-8 code decade "up" counter. After setting up the truth table, a decision must be made as to where to obtain the trigger pulse for each flip-flop.

All the FC family flip-flops change their output levels during a change of the trigger signal from H to L, so that during a L to H change, or during no change, the levels of the J and K inputs are immaterial for the present. Tentatively, an X may be entered in the change function table, denoting that "0" and "1" are equally suitable. Actually, this will depend finally on the particular type of flip-flop used (see section 2.2).

In the change function table, a "1" denotes a change of state of a flip-flop. However, because the counter always starts in the "zero" position (all flip-flop outputs LOW) we know that the first "1" in any column of the change function table represents an output change from L to H, the next "1" representing H to L, and so on. The trigger pulses available, i.e. the H to L changes thus occur at every alternate "1" in the table.

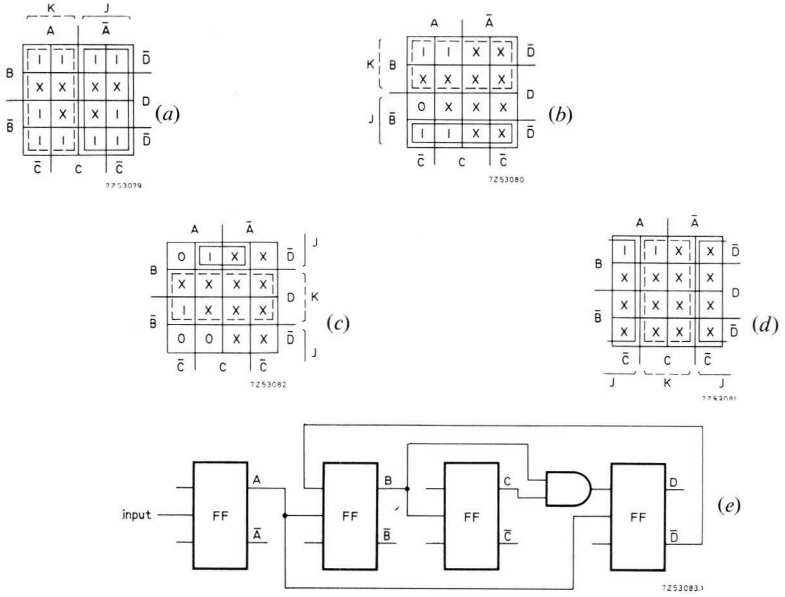


Fig. 3–8. The complete design of an asynchronous 1248 code decade "up" counter, with the illegitimate states unspecified. The resulting input conditions to the flip-flops are as follows.

(a) Flip-flop A: $J_A = 1$, $K_A = 1$.

(b) Flip-flop B: $J_B = \overline{D}$, $K_B = 1$.

(c) Flip-flop C: $J_C = 1$, $K_C = 1$.

(d) Flip-flop D: $J_D = B{\cdot}C$, $K_D = 1{\cdot}$

(e) The circuit.

The first flip-flop will be triggered by the clock pulses; flip-flop B will change its output state at half the frequency of A if it is triggered by output A. It must be kept in mind that the number of "1"s in any column must always be an even number, because the counter must finish with all outputs in the LOW state. If all alternate "1"s in column A were used to trigger B, column B would contain five "1"s, leaving flip-flop B at H. To prevent this, an "0" must be put in the last position in column B. The second and fourth "1" in this column can now be used to trigger flip-flop C, and there will thus be two "1"s in column c.

It is clear that flip-flop D cannot be triggered by C as this offers only one suitable change, which would leave D in the "1" state. On the other hand, triggering from B would mean that C and D behaved identically, creating a counter-of-8. Flip-flop A must thus trigger D. The first change in D should occur at the eighth clock pulse, so that a "1" must be inserted at position 7. It must not change in positions 1, 3 and 5, so that zeroes must be inserted in these positions. The flip-flop must return to the state of combination "0" at the ninth clock pulse, so a "1" must also be inserted at this position. If the illegitimate states are not specified for this counter, the resulting Karnaugh maps and the counter will appear as in Fig. 3–8.

3.2 Scalers

It is possible to build a scaler from flip-flops and gates having any scaling factor, using the method described in section 3.1. It is also possible however to build *gateless* scalers of any scaling factor, the disadvantage being that there is no single method by which such scalers may be designed. In fact there are many basic designs — the Johnson, Möbius (twisted ring) and ripple-through scalers being three of the most used — each design having its advantages and disadvantages.

Scalers with scaling factors n_1, n_2, n_3, . . . can be cascaded to give a scaler of scaling factor (n_1 n_2 n_3 . . . .) We shall therefore limit ourselves here to presentation in tabular form of the more important data for scalers with scaling factors of from 2 to 11, and 13. Scalers with scaling factors of 4, 8 and 16 (binary magnitudes) are straightforward designs; the scaling factors of 12, 14 and 15 are also very easily obtained by cascading two or more of the other scalers. Note that when cascading two or more scalers, the speed of the total will be the speed of the first (input side) scaler. For example, a scaler with scaling factor of 6 can be made from one 2-scaler and one 3-scaler, thus having the speed of a 2-scaler. Alternatively it could be made from one 3-scaler and one 2-scaler, when it would have the speed of a 3-scaler.

In Table 3–2, the 12 columns have the following significances.

Column 1: The scaling factor of the circuit.
Column 2: The figure number of the circuit.
Column 3: Type of flip-flop most suitable.
(FCJ101 and FCJ121* are preferred.)
Column 4: number of flip-flop packages required.
Column 5: the typical minimum clock pulse period for an asymmetric clock pulse.
Column 6: the typical maximum clock pulse frequency for an asymmetric clock pulse.
Column 7: the typical minimum clock pulse period for a symmetric clock pulse.
Column 8: the typical maximum clock pulse frequency for a symmetric clock pulse.
Column 9: the load presented by the scaler circuit to the clock pulse generator, expressed in equivalent d.c. gate loads.
Column 10: the output terminal which, because a negative-going signal is produced, will in most cases be chosen to deliver the output signal for drive of further logic operations (although the choice is optional).
Column 11: the fan-out (in equivalent d.c. gates) of the output terminal in Column 10.
Column 12: notes.

Note 1: type FCJ101 flip-flop only is to be used in this configuration.
Note 2: This configuration requires an extra gate if a negative-going output pulse flank is required (the circuit however needs one less flip-flop than the others).
Note 3: these configurations have illegitimate combinations into which they may lock if forced into those combinations by interference, for example. Configurations to which this note does not apply have illegitimate states from which the scaler will return to the legitimate state when triggered.

* The FCJ131, FCJ191 and FCJ211 are also suitable.

Table 3–2. Gateless Scaler Data Review

scaling factor	fig. no.	preferred flip-flop type FCJ...	number of packages	asymmetric clock pulse		symmetric clock pulse		load on clock pulse generator	output terminal	output fan-out	notes
				typical minimum clock pulse period (ns)	typical maximum clock pulse frequency (MHz)	typical minimum clock pulse period (ns)	typical maximum clock pulse frequency (MHz)				
1	2	3	4	5	6	7	8	9	10	11	12
3	3–10*a*	101	2	165	6.0	200	5.0	7	Q_B	7	
		121	1	195	5.0	240	4.2	6	Q_B	7	
	3–10*b*	101	2	165	6.0	200	5.0	7	Q_B	8	
		121	1	195	5.0	240	4.2	6	Q_B	8	
	3–10*c*	101	2	165	6.0	200	5.0	7	Q_B	8	
		121	1	195	5.0	240	4.2	6	Q_B	8	
	3–10*d*	101	2	165	6.0	200	5.0	7	Q_B	7	
		121	1	195	5.0	240	4.2	6	Q_B	7	
	3–10*e*	101	2	165	6.0	200	5.0	7	Q_B	8	
		121	1	195	5.0	240	4.2	6	Q_B	8	
	3–10*f*	101	2	165	6.0	200	5.0	7	Q_A	8	
		121	1	195	5.0	240	4.2	6	Q_A	8	
	3–10*g*	101	2	165	6.0	200	5.0	7	Q_B	8	
		121	1	195	5.0	240	4.2	6	Q_B	8	
	3–10*h*	101	2	165	6.0	200	5.0	7	Q_B	8	
		121	1	195	5.0	240	4.2	6	Q_B	8	

5	3–12*a*	101	3	265	3.9	265	3.9	7	Q_C	8	3
		121	1½	315	3.2	480	2.1	6	Q_C	8	
	3–12*b*	101	3	265	3.9	265	3.9	7	Q_C	8	
		121	1½	315	3.2	480	2.1	6	Q_C	8	
	3–12*c*	101	3	235	4.3	235	4.3	7	Q_B	7	
		121	1½	285	3.5	420	2.4	6	Q_B	7	
	3–12*d*	101	3	265	3.9	265	3.9	7	Q_C	8	3
		121	1½	315	3.2	480	2.1	6	Q_C	8	
	3–12*e*	101	3	235	4.3	235	4.3	7	Q_B	7	3
		121	1½	285	3.5	420	2.4	6	Q_B	7	
	3–12*f*	101	3	165	6.0	200	5.0	10.5	Q_C	8	
		121	1½	195	5.0	240	4.2	9	Q_C	8	
	3–12*g*	101	3	165	6.0	200	5.0	10.5	Q_C	7	
		121	1½	195	5.0	240	4.2	9	Q_C	7	
6	3–13*a*	101	3	265	3.9	265	3.9	3.5	Q_C	8	1
	3–13*b*	101	3	165	6.0	200	5.0	10.5	Q_C	7	3
		121	1½	195	5.0	240	4.2	9	Q_C	7	
7	3–14*a*	101	3	165	7.0	200	5.0	10.5			2,3
		121	1½	195	5.0	240	4.2	9			
	3–14*b*	101	4	265	3.9	265	3.9	14	Q_D	8	3
		121	2	315	3.2	480	2.1	12	Q_D	8	
	3–14*c*	101	4	165	6.0	200	5.0	14	Q_D	8	
		121	2	195	5.0	240	4.2	12	Q_D	8	
	3–14*d*	101	4	165	6.0	200	6.0	7	Q_D	8	
		121	2	195	5.0	240	4.2	6	Q_D	8	

Table 3-2 Gateless Scaler Data Review. (continued)

				asymmetric clock pulse		symmetric clock pulse					
scaling factor	fig. no.	preferred flip-flop type FCJ...	number of packages	typical minimum clock pulse period (ns)	typical maximum clock pulse frequency (MHz)	typical minimum clock pulse period (ns)	typical maximum clock pulse frequency (MHz)	load on clock pulse generator	output terminal	output fan-out	notes
1	2	3	4	5	6	7	8	9	10	11	12
9	3–16*a*	101	4	365	2.7	365	2.7	7	Q_C	7	3
		121	2	435	2.3	720	1.4	6	Q_C	7	
	3–16*b*	101	4	365	2.7	365	2.7	7	Q_C	7	3
		121	2	435	2.3	720	1.4	6	Q_C	7	
	3–17*a*	101	4	165	6.0	200	5.0	3.5	Q_D	8	3
		121	2	195	5.0	240	4.2	3	Q_D	8	
	3–17*b*	101	4	265	3.9	265	3.9	7	Q_D	8	3
10		121	2	315	3.2	480	2.1	6	Q_D	8	
	3–17*c*	101	4	335	3.0	335	3.0	7	Q_C	7	
10		121	2	405	2.5	660	1.5	6	Q_C	7	
	3–17*d*	101	5	165	6.0	200	5.0	18.5	Q_E	7	3
		121	$2\frac{1}{2}$	195	5.0	240	4.2	15	Q_E	7	
	3–17*e*	101	4	165	6.0	200	5.0	10.5	Q_D	8	
		121	2	195	5.0	240	4.2	9	Q_D	8	

11	3–18*a*	101	5	265	3.9	265	3.9	7	Q_D	7	3
		121	2½	315	3.2	480	2.1	6	Q_D	7	
	3–18*b*	101	5	335	3.0	335	3.0	7	Q_D	8	3
		121	2½	405	2.5	660	1.5	6	Q_D	8	
13	3–19*a*	101	5	365	2.7	365	2.7	7	Q_D	7	3
		121	2½	435	2.3	720	1.4	6	Q_D	7	
	3–19*b*	101	5	365	2.7	365	2.7	7	Q_D	7	3
		121	2½	435	2.3	720	1.4	6	Q_D	7	
	3–19*c*	101	5	365	2.7	365	2.7	7	Q_D	8	3
		121	2½	435	2.3	720	1.4	6	Q_D	8	
	3–19*d*	101	5	335	3.0	335	3.0	7	Q_D	8	1,3
2	3–9	101	1	165	6	200	5		Q_A	8	
		121	½	195	5	240	4.2		Q_A	8	
4	3–11*a*	101	2	165	6	200	5		Q_B	8	
		121	1	195	5	240	4.2		Q_B	8	
	3–11*b*	101	2	165	6	200	5		Q_B	8	
		121	1	195	5	240	4.2		Q_B	8	
8	3–15*a*	101	3	165	6	200	5		Q_C	8	
		121	1½	195	5	240	4.2		Q_C	8	
	3–15*b*	101	3	165	6	200	5		Q_C	8	
		121	1½	195	5	240	4.2		Q_C	8	

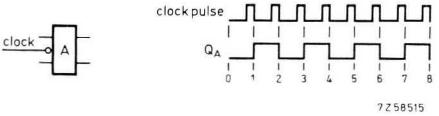


Fig. 3–9. Scaler-of-2.

Fig. 3–10. Scalers-of-3. Output time relationships are shown to the right of each circuit. (See Table 3–2 for characteristics). →

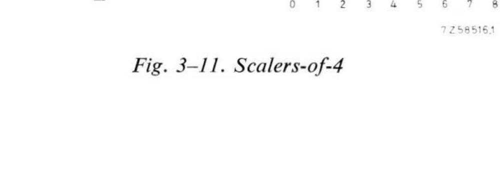


Fig. 3–11. Scalers-of-4

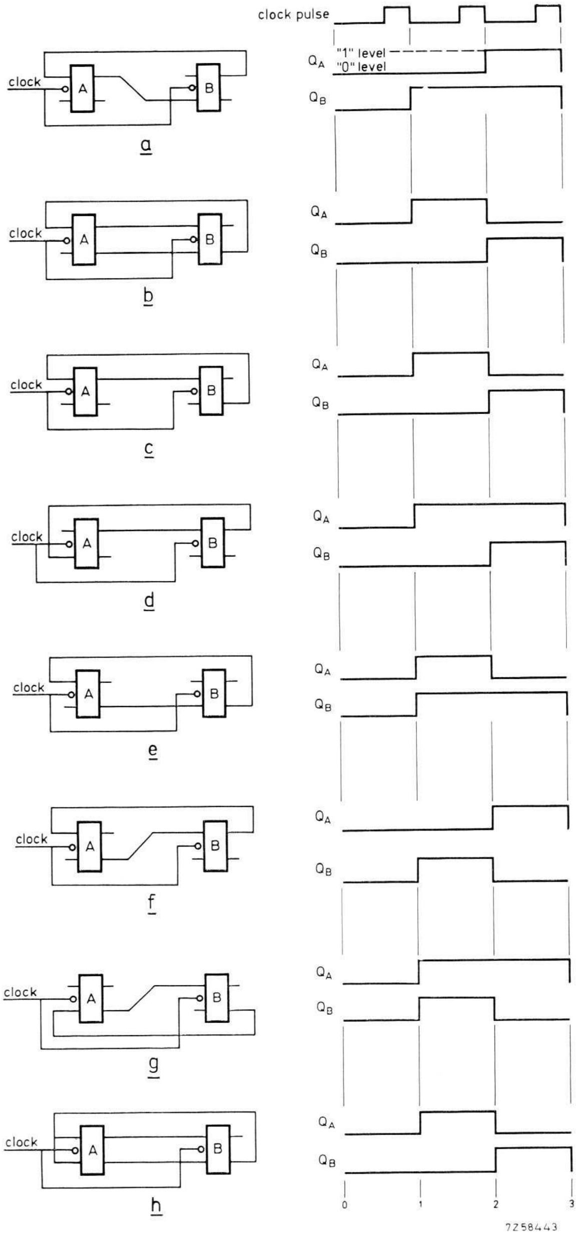

clock pulse
"1" level
"0" level
Q_A
Q_B
clock
A
B
a
b
c
d
e
f
g
h
0
1
2
3
7Z58443

Fig. 3-10

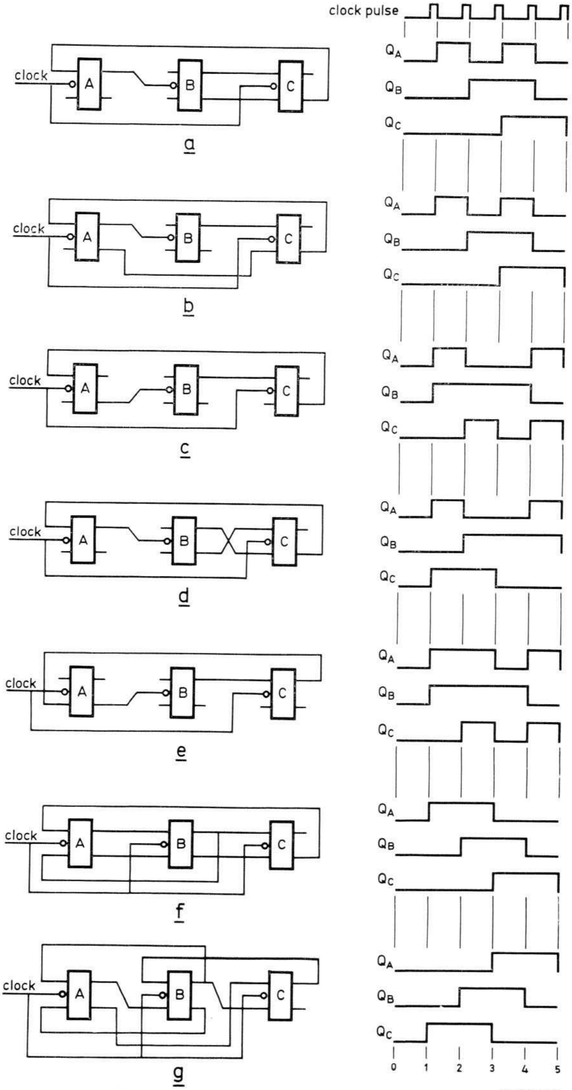


Fig. 3–12. Scalers-of-5.

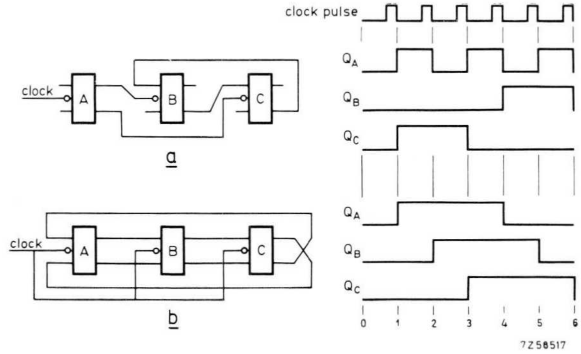


Fig. 3–13. Scalers-of-6.

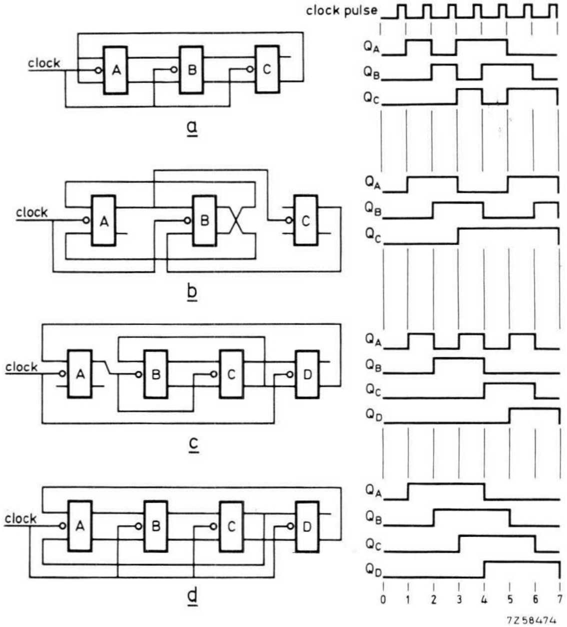


Fig. 3–14. Scalers-of-7.

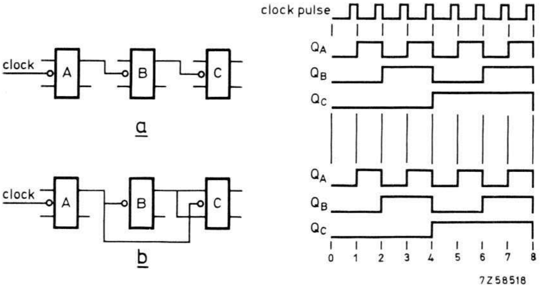


Fig. 3–15. Scalers-of-8.

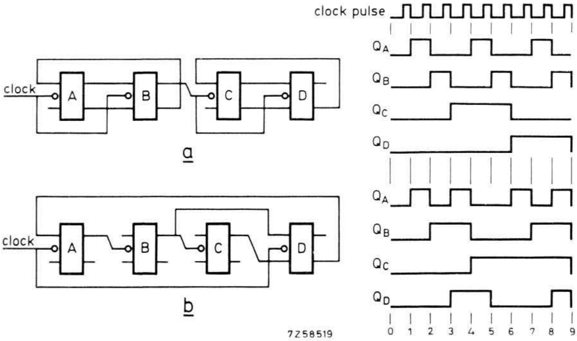


Fig. 3–16. Scalers-of-9.

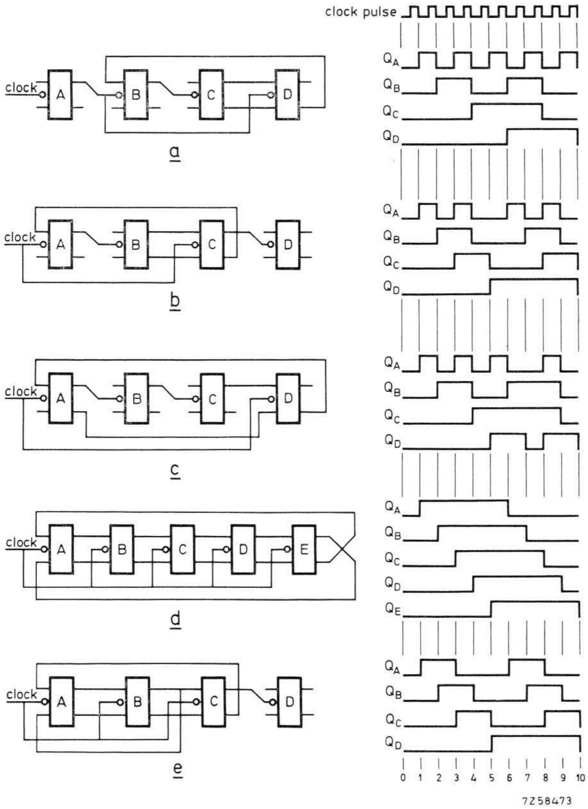


Fig. 3–17. Scalers-of-10.

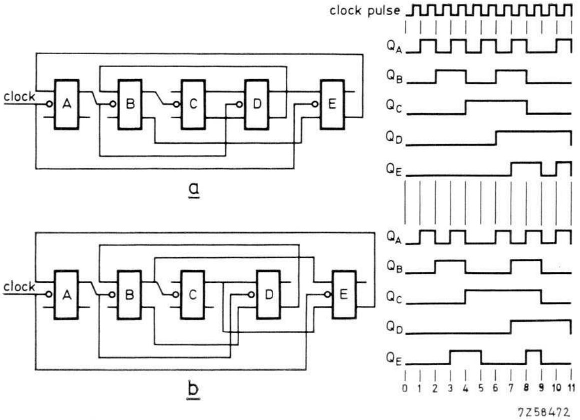


Fig. 3–18. Scalers-of-11.

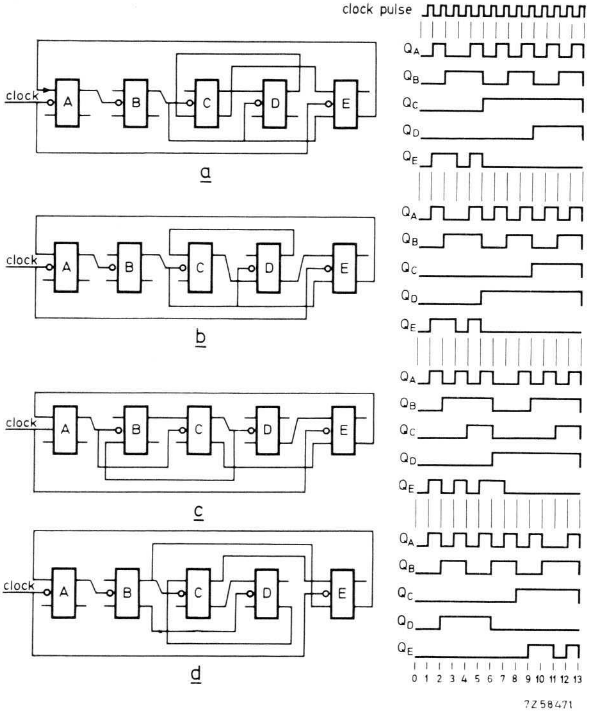


Fig. 3–19. Scalers-of-13.

3.3 Counters

From the descriptions of counters and scalers given in section 3.1.1, it may be inferred that the two circuits are basically similar. In fact a scaler can be operated as a counter if a decoding circuit is added. The so-called "weighted code" method is commonly employed for this decoding purpose. In this "weighted code" type of counter the "weighting" of each flip-flop allows a direct numerical indication of the counters position upon adding the "weights" of all the flip-flops in the HIGH state at any given moment.

In the following sections some examples are given of counters-of-16 and counters-of-10 (decade counters).

3.3.1 COUNTERS-OF-16

There are no less than 24 ways of counting to 16, ranging from the completely synchronous counter of Fig. 3–20 to the ripple-through counter of Fig. 3–24, some intermediate combinations being shown in Figs 3–21, 22, 23. We have studied the problem with an eye to selecting the best configuration with respect to one or more of the following properties: delay; load on driving stage; excess fan-out from each stage; maximum counting rate; number of packages.

Of the twenty-four we have rejected ten because the timing of the *J-K* input signals could, under total worst-case conditions, be wrong for one or more flip-flops. For each of the remaining fourteen configurations three counters were designed:

- Using FCJ101: for speed (particularly in certain configurations) and the ability to preset on any number.

- Using FCJ191 of FCJ211: for economy, but retaining the preset facility.

- Using FCJ121: for plain economy (no preset facility).

The resulting 42 counters were then compared in respect of the properties mentioned earlier, 11 being found to excel in at least one property. These counters and their characteristics (for full temperature range: T_{amb} = 0–75 °C) are presented in Table 3–3.

Table 3–3. Characteristics of Counters-of-16.

number of packages	recom-mended flip-flop	circuit figure	reset to zero	preset on any number	load upon the driving stage[1]	available fan-out[3] *A*	*B*	*C*	*D*	max.[2] delay (ns)	max. freq. (MHz)
4	FCJ101	3–20	Yes	Yes	14	2	4	6	8	100	3.8
4	FCJ101	3–21*a*	Yes	Yes	7	6	1	6	8	200	3.8
2	FCJ211	3–21*b*	Yes	Yes	5	6	3	6	8	240	4.0
2	FCJ121	3–21*c*	Yes	No	5	6	3	6	8	240	4.0
4	FCJ101	3–22*a*	Yes	Yes	7	6	4	4	8	300	3.8
2	FCJ191	3–22*b*	Yes	Yes	5	6	5	5	8	360	4.0
2	FCJ121	3–22*c*	Yes	No	5	6	5	5	8	360	4.0
4	FCJ101	3–23	Yes	Yes	4	4	1	6	8	300	6.0
4	FCJ101	3–24*a*	Yes	Yes	4	4	4	4	8	400	6.0
2	FCJ191	3–24*b*	Yes	Yes	3	5	5	5	8	480	5.0
2	FCJ121	3–24*c*	Yes	No	3	5	5	5	8	480	5.0

1) Load on the driving stage. This is expressed as the number of equivalent gate loads and is valid only as a d.c. load. This means that for the FCJ101 which is edge triggered (a.c.) the published d.c. values for the trigger input have been raised by extra gate loads to ensure that the driving stage is capable of discharging the capacitive trigger input within the allotted time.

2) Maximum delay. The time quoted is worst case, both with respect to device tolerance and to number-to-number transition; it is the elapsed time between the rear flank of the clock (trigger) pulse and the moment at which the new code is available. Maximum delay does not limit the maximum counting frequency in simple scaler applications, as can be seen in the adjacent column, but where specific action is required between clock pulses, the value is important. In this case, to the figure quoted must be added the delay of such external circuits as comparators, inhibit gates, etc., to fix the total loop delay, and thus the maximum allowable clock rate for the application.

3) Fan-out. The unloaded complementary outputs ($\bar{A}$, $\bar{B}$, $\bar{C}$, $\bar{D}$) are not included in the figure quoted, full fan-out of eight being available in every case.

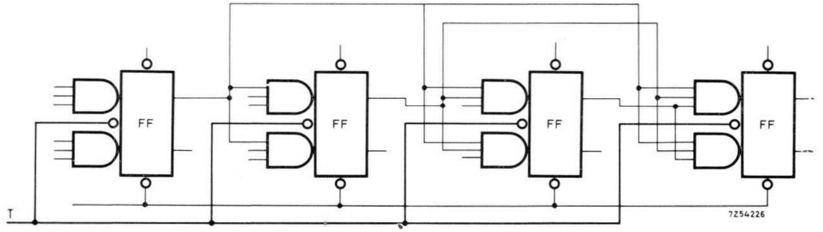
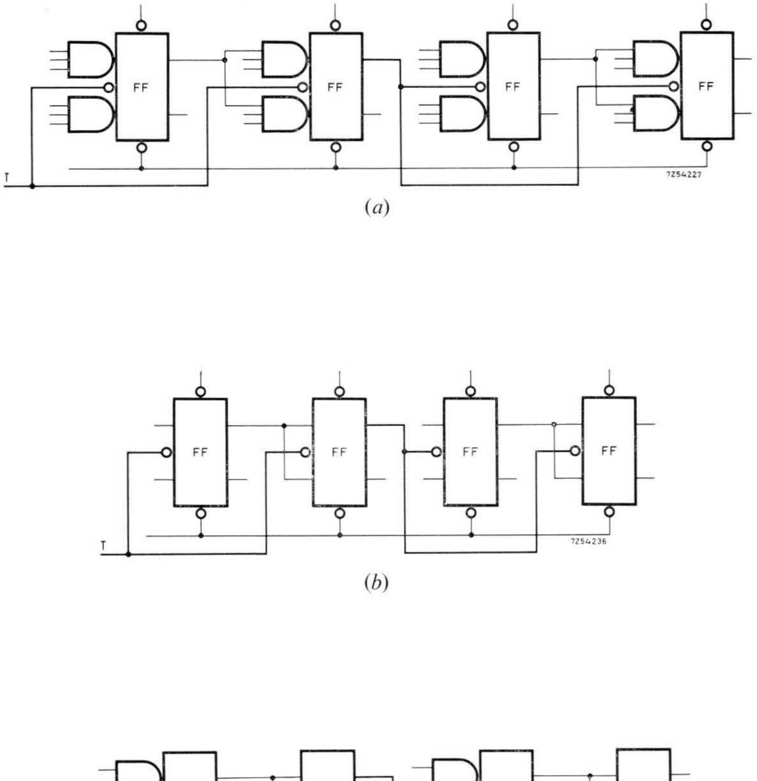


Fig. 3–20. Counter-of-16, wholly synchronous, 1248 code. For characteristics see Table 3–3.

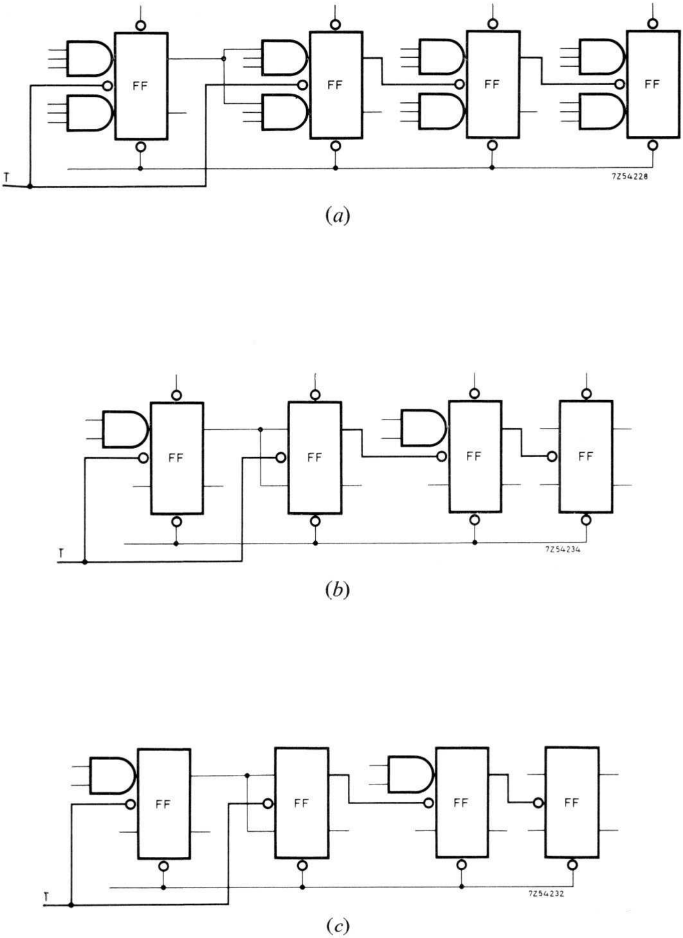


(*a*)

(*b*)

(*c*)

Fig. 3–21. Counters-of-16 (two counters-of-4 in series), 1248 code.



(*a*)

(*b*)

(*c*)

Fig. 3–22. Counters-of-16 (counters-of-4,-2,-2 in series), 1248 code.

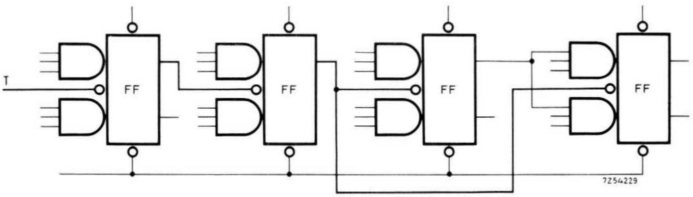
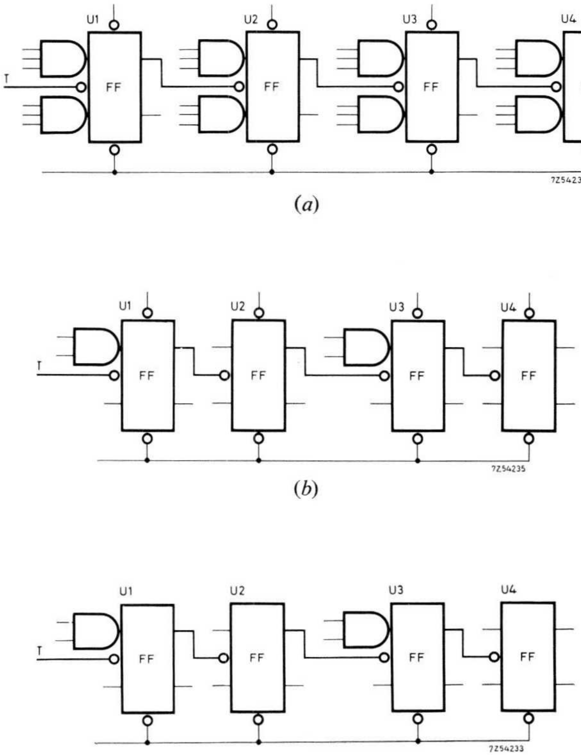


Fig. 3–23. Counters-of-16 (counters-of-2,-2,-4 in series), 1248 code.



(*a*)

(*b*)

(*c*)

Fig. 3–24. Counters-of-16, wholly asynchronous (ripple-through type), 1248 code.

3.3.2 DECADES (COUNTERS-OF-10)

Decade counters differ with respect to counters-of-16 or other counters of powers of 2, in that the redundancy of some flip-flop combinations makes discrimination between legitimate and illegitimate states necessary. There are a number of ways of making this discrimination, and counters of both weighted and unweighted type are possible depending on choice of legitimate states.

Below are given some configurations for decade counters using various codes and flip-flop types. (The reader is also referred to section 3.1 in which a 1-2-4-8 code decade counter is taken as an example.) Table 3–4 lists the characteristics and features of the circuits which are depicted in Figs 3–25 to 3–31. The circuits were chosen because they possess one or more of the following advantages:

- economy, denoting few flip-flop and gate packages;
- small input requirements;
- availability of fan-out;
- high counting speed;
- small delay between an input pulse and the switching over all of relevant flip-flops.

Note that the maximum counting speed is quoted for input waveforms as specified in the Data Sheets of the flip-flops concerned. It has also been assumed that the maximum propagation delay occurs, i.e. worst-case conditions exist. If a symmetrical shaped input waveform is used ($t_{TH} = t_{TL}$) having t_{TL} equal to that specified in the Data Sheets, the counting speed must be reduced.

It should also be noted that counters made with type FCJ191 flip-flops are similar in characteristics to those using the type FCJ121, except that the FCJ191 has a "setting" ability. The same logic diagram can be used for the FCJ121 as for the FCJ191; the "set" terminal for the latter is drawn dotted in the diagrams. The FCJ191 has a 16 pin D.I.L. package.

Notes to table 3-4

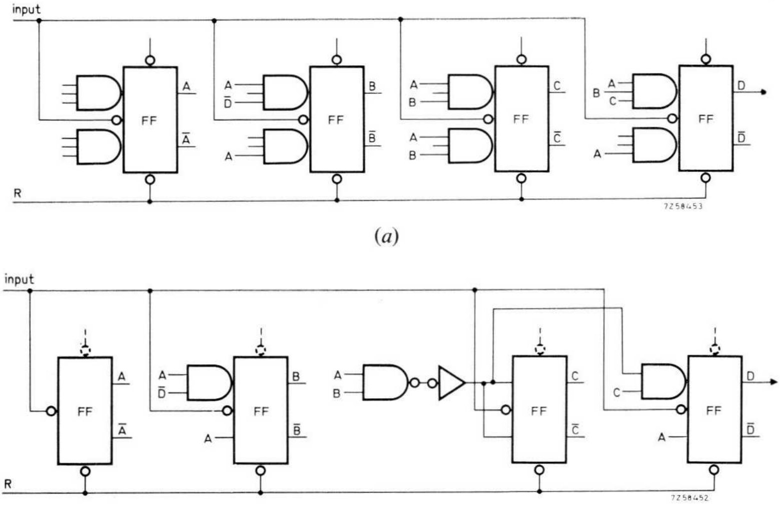
* Synchronous or asynchronous.

** Equivalent gate loads.

† T is the reciprocal of the maximum counting speed, and composed mainly of $t_{TL} + t_{TH}$, to which relevant preparatory times and flank rise and fall times must be added.

Table 3–4. Decimal Counters

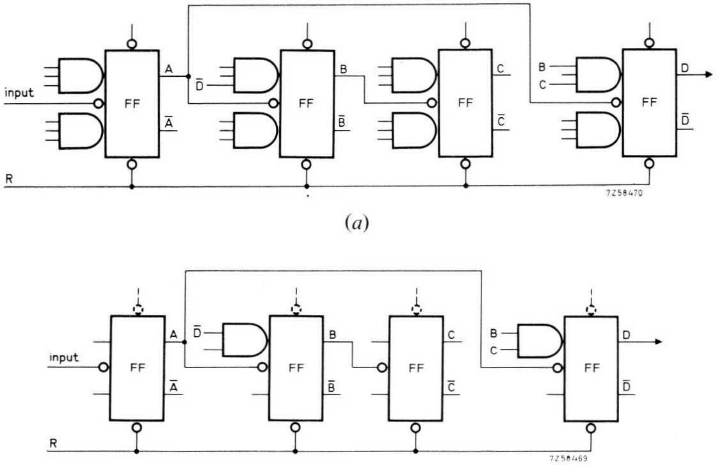
code	S or A*	flip-flop type	number of		requir-ed input drive **	excess fan-out								max. count-ing speed	t_{TL}	t_{TL} (ns)	T*	max. delay (ns)	ability to set	fig. no.
			flip-flop packages	gate packages		A	$\overline{A}$	B	$\overline{B}$	C	$\overline{C}$	D	$\overline{D}$							
B.C.D. up	S	FCJ101	4	0	14	2	8	5	8	7	8	8	7	5.0	100	50	200	100	yes	3–25*a*
	A	FCJ101	4	0	3.5	1	8	3	8	7	8	8	7	6.0	100	50	166	300	yes	3–26*a*
	S	FCJ121	2	2	9.2	4	8	7	8	7	8	8	7	2.8	280	60	350	120	no	3–25*b*
	A	FCJ121	2	0	2.3	5	8	6	8	7	8	8	7	1.8	480	60	550	480	no	3–26*b*
	S	FCJ201	4	0	8	4	9	6	9	8	9	9	8	3.0	200	100	330	200	yes	3–25*a*
	A	FCJ201	4	0	2	5	9	6	9	8	9	9	8	3.0	200	100	330	400	yes	3–26*a*
	S	FCJ191	2	2	9.2	4	8	7	8	7	8	8	7	2.8	280	60	350	120	yes	3–25*b*
	A	FCJ191	2	0	2.3	5	8	6	8	7	8	8	7	1.8	480	60	550	480	yes	3–26*b*
B.C.D. down	S	FCJ101	4	1	14	8	2	8	6	8	6	7	7	4.2	100	50	240	100	yes	3–27*a*
	A	FCJ101	4	1	3.5	8	1	8	3	8	6	8	7	3.0	100	50	335	300	yes	3–28*a*
	S	FCJ121	2	2	9.2	7	4	8	7	8	6	7	7	3.9	195	60	255	120	no	3–27*b*
	S	FCJ201	4	1	8	9	3	9	7	9	7	8	8	3.0	235	100	335	200	yes	3–27*a*
	A	FCJ201	4	1	2	9	5	9	6	9	7	9	8	3.0	200	100	335	400	yes	3–28*a*
	S	FCJ191	2	2	9.2	7	4	8	7	8	6	7	7	3.9	195	60	255	120	yes	3–27*b*
	A	FCJ191	2	1	2.3	8	3	8	5	8	6	8	7	2.8	120	60	360	360	yes	3–28*b*
B.C.D. revers-ible	S	FCJ101	4	2	14	7	7	7	7	7	6	7	7	4.7	100	50	210	100	yes	3–29*a*
	S	FCJ201	4	2	8	8	8	8	8	8	7	8	8	2.9	240	100	340	200	yes	3–29*a*
	S	FCJ191	2	4	9.2	4	5	7	7	7	6	7	6	2.9	280	60	340	120	yes	3–29*b*
excess-three up	S	FCJ121	2	2	9.2	6	7	7	8	6	8	7	8	3.8	200	60	260	120	no	3–30*b*
	A	FCJ121	2	1	6.9	7	7	7	8	5	8	7	8	2.6	280	100	380	240	no	3–31
	S	FCJ201	4	1	8	6	8	7	8	6	9	8	9	2.6	280	100	380	200	yes	3–30*a*
	S	FCJ191	2	2	9.2	6	7	7	8	6	8	7	8	3.8	200	60	260	120	yes	3–30*b*
	A	FCJ191	2	1	6.9	7	7	7	8	5	8	7	8	2.6	280	100	380	240	yes	3–31



(*a*)

(*b*)

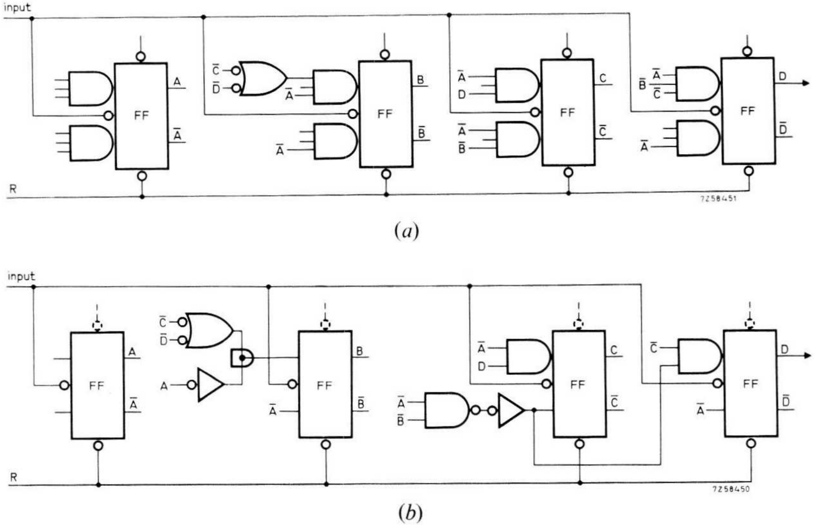
Fig. 3–25. Synchronous 1248 code decade "up" counter.



(*a*)

(*b*)

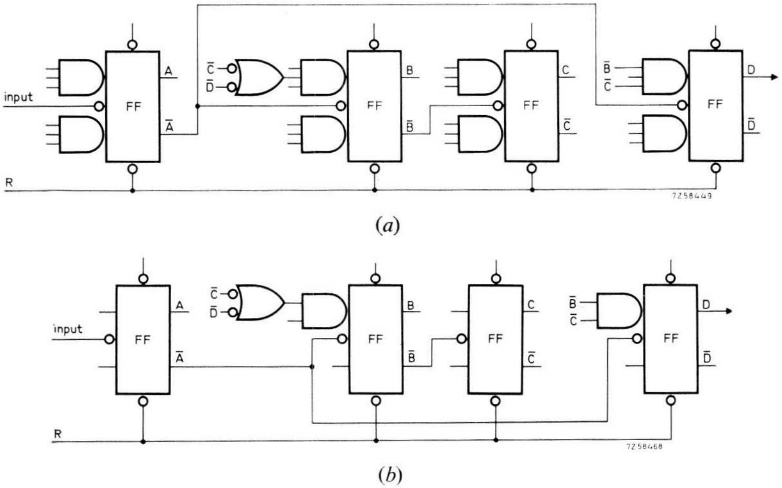
Fig. 3–26. Asynchronous 1248 code decade "up" counter.



(*a*)

(*b*)

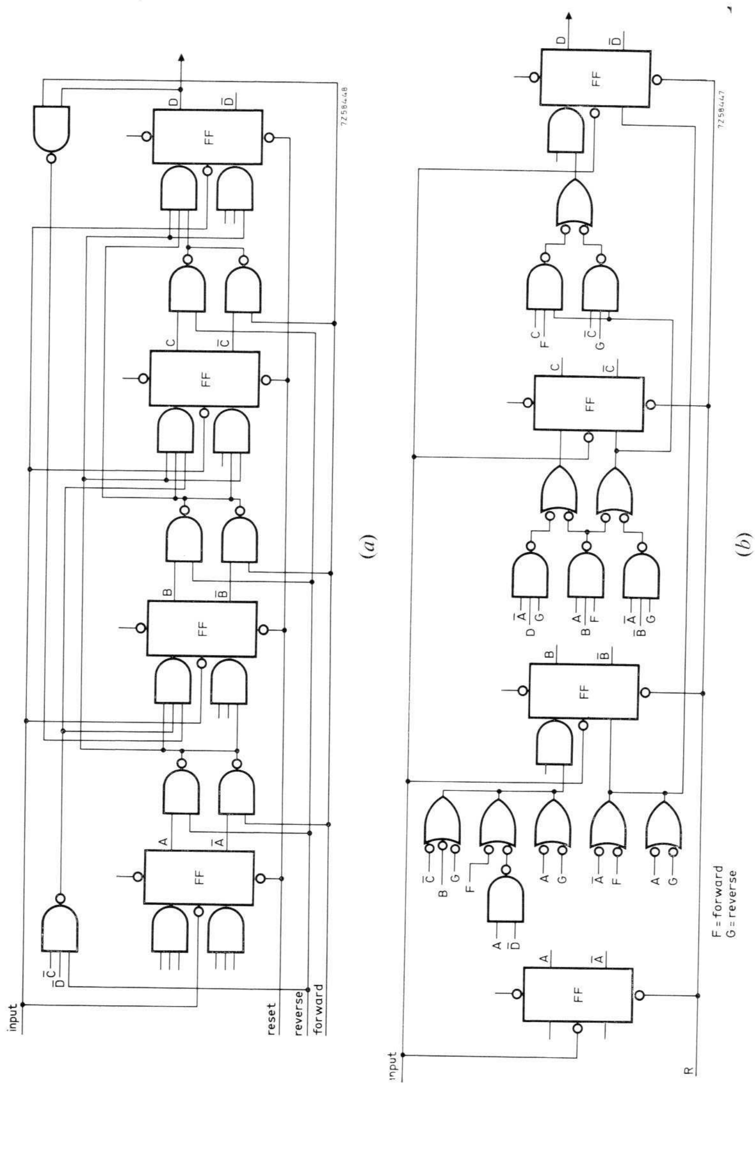
Fig. 3–27. Synchronous 1248 code decade "down" counter.



(*a*)

(*b*)

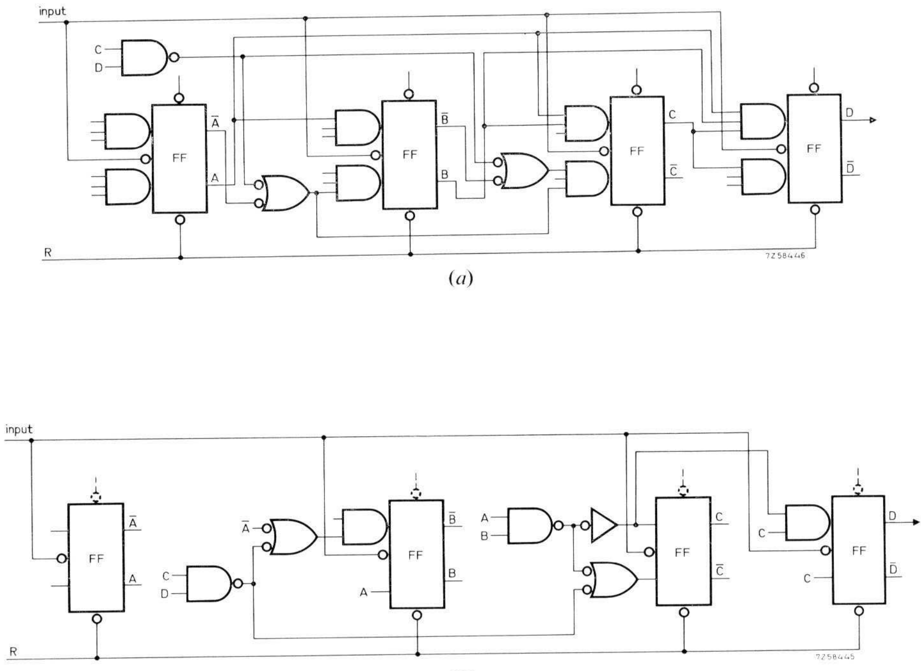
Fig. 3–28. Asynchronous 1248 code decade "down" counter.



(a)

(b)

Fig. 3–29. Reversible synchronous 1248 code decade counter.



(a)

(b)

Fig. 3–30. Synchronous Excess-Three code decade "up" counter.

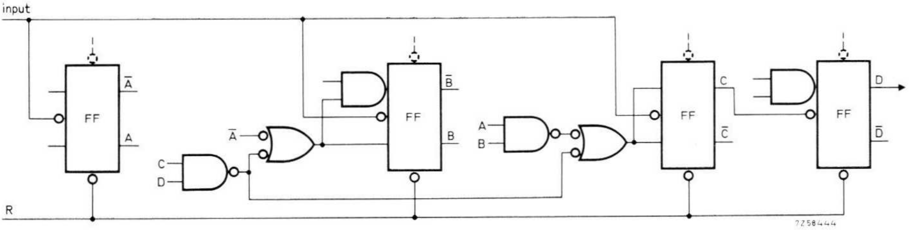


Fig. 3–31. Asynchronous Excess-three code decade "up" counter.

For completeness sake two reversible type 1248 decade counters that can be cascaded have been included (Fig. 3–32*a* based on FCJ 111, Fig. 3–32*b* on FCJ 201.)

The following truth-table is valid for inputs L_F and L_R.

counting	L_F	L_R
forward	L	H
reverse	H	L
blocked	H	H

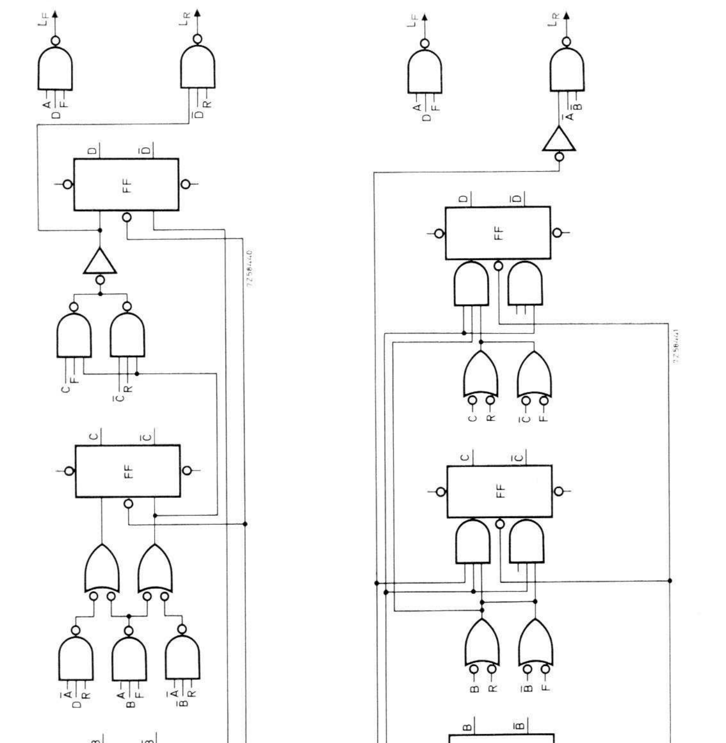


Fig. 3–32. Reversible synchronous 1248 code decade counter which may be cascaded, based on (a) FCJ111, (b) FCJ201.

3.4 Shift Registers

A shift register is a circuit able to store information and transfer it upon command; it is the simplest of the three main types of sequential logic circuit. It may be termed a "passive" device, requiring information to be written in before passing it to the output, opposed to "active" devices such as counters which require only trigger pulses to pass information to the output.

The presentation and the extraction of information each can be in series or parallel. There are thus four possibilities: series in, series out; parallel in, parallel out; series in, parallel out; parallel in, series out.

The simplest circuitry is achieved by using the series in, series out mode. This is depicted in Fig. 3–33*a*. For this mode of operation the flip-flops need not have a separate set terminal; their reset terminals may be connected internally where dual or multiple flip-flops in one package are used, as may also the trigger input terminals. Any flip-flop in the FC family may thus be used; Fig. 3–33 was drawn for the FCJ121 type.

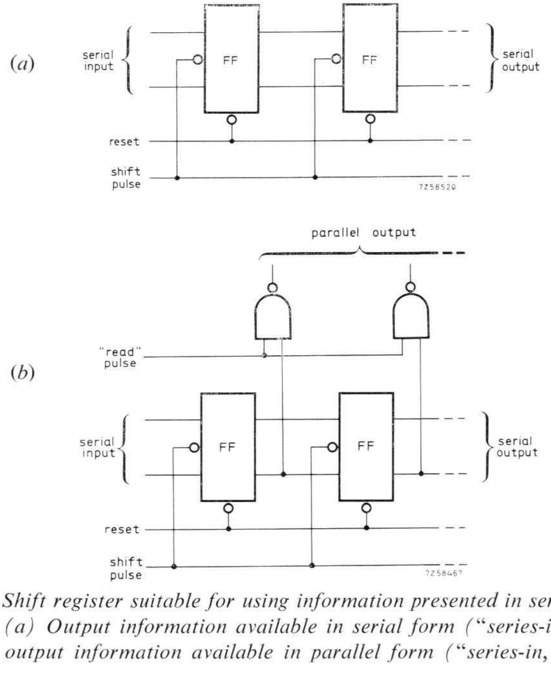


Fig. 3–33. Shift register suitable for using information presented in serial form ("series-in" type). (a) Output information available in serial form ("series-in, series-out" circuit); (b) output information available in parallel form ("series-in, parallel-out" circuit).

If parallel output is required, each flip-flop must then be provided with a gate at its complementary output ($\bar{Q}$), as shown in Fig. 3–33*b*. The gate output level then is the same as the flip-flops other (normal) output when the "Read" signal, a "H" level pulse, is applied to the gates other input. The gate output will normally be HIGH. It will go LOW when both the normal (non-inverted) flip-flop output goes LOW (complementary output high) and a "Read" pulse is applied. Thus the inactive state of the gate outputs is the HIGH level.

The general characteristics of a serial input shift register are the following.

- The input information must be available in both the inverted and the non-inverted forms (unless *D* flip-flops are used).
- Shifting takes place at the negative-going flank of each shift pulse. If the type FCJ 121 is used as flip-flop, the number of such flip-flops that can be driven synchronously from various shift pulse sources is listed in Table 3–5, for the temperature range 0 °C to 75 °C. Note that if the shift pulse is to be derived from several gate or line drivers in parallel, care must be taken to ensure that excessive skewness of the shift pulse does not occur.
- The shift register can be reset by applying a low level signal to the reset terminals. Table 3–5 gives the maximum number of type FCJ121 flip-flops which can be reset by various reset signal sources.
- The maximum shift pulse frequency is 5 MHz.

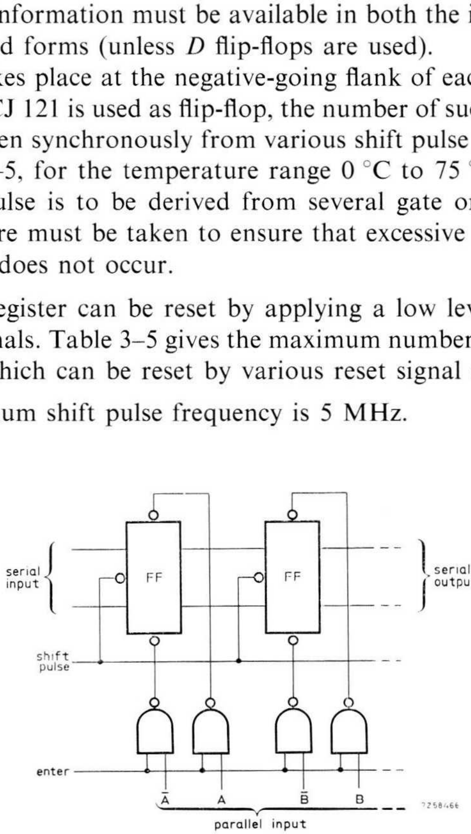


Fig. 3–34. Shift register suitable for using parallel information ("parallel-in, series-out" circuit).

Table 3–5. Source capability when driving type FCJ121 flip-flops.

source	maximum number of type FCJ121 flip-flops	
	able to be triggered	able to be reset
FCH211 (gate)	3	4
FCH101 (gate)	3	4
FCH221 (line driver)	6	8
FCH231 (line driver)	8	12
Heavy duty line driver (see section 2.1.6)	48	66

The circuit of a shift register using parallel input/serial output is given in Fig. 3–34. (Parallel output can be achieved in the same way as shown in the circuit of Fig. 3–33*b*.) Parallel input requires a flip-flop type having a separate set terminal, and all the FC family flip-flops have this except types FCJ121 and FCJ131.

The parallel input shift register has the following characteristics.

— The parallel input information must be available both in the inverted and non-inverted form.

— The shifting action occurs at the negative-going flank of the shift pulses. All inputs (where applicable) of flip-flops in the FC family have the same requirements so that the maximum numbers of type FCJ121 given in Table 3–5 for triggering are also valid for the other types.

— The "Enter" input signal must be at HIGH level to enter new information.

— A separate "Reset" signal is not necessary for parallel input shift registers, because if no information is presented to the gate inputs, the next "Enter" signal will cause all flip-flops to register zero.

— The maximum shift pulse frequency is 5 MHz.

4 Applications in Combinational Logic

In the previous chapter were discussed "Sequential Logic" circuits whose functioning depended not only on the state of the inputs at any one moment but also on conditions which had existed previous to that moment. Circuits of the "Combinational Logic" type however give an output which directly reflects the input states, the previous conditions having no influence on the result. Combinational Logic circuits have no memorizing capacity, because the *sequence* of changes of state at the inputs has no bearing on the output state.

Combinational Logic circuits are built up from logic gates such as the FCH NAND gate series. By combining these basic units into various configurations, an unlimited number of higher order functions may be made, examples being Adders, Complementers, Code Correctors, Comparators, Parity Checkers, Encoders and Decoders.

Some of these circuits are discussed in the following three sections: 4.1, 4.2 and 4.3.

4.1 Adders

The basic function of an adding circuit, addition of two binary digits (say A and B), can be expressed in Boolean algebra by:

$$S_H = \bar{A}\cdot B + A\cdot\bar{B}\,,$$

and

$$C_H = A\cdot B\,,$$

where S_H is the exclusive-OR combination of A and B, and C_H is the remainder or "carry" quantity to be added to the next significant digits. The circuit which performs the above operation is called a "half-adder" and is shown in Fig. 4-1.

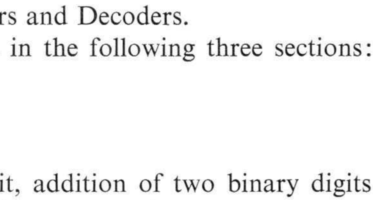


Fig. 4–1. The half-adder logic circuit.

This circuit can be made by proper interconnection of the four 2-input NAND gates in the FCH191 package; note that $\bar{C}_H$ is produced instead of C_H, the latter requiring an extra inversion stage. Fig. 4-2 shows the pin numbers and interconnections.

In electronic calculators a more common operation is the summing of three binary digits, because the carry quantity must be added to the sum of the next significant two binary digits if the sum of two multi-digit binary numbers is to be produced. This can be done by the "full adder". It can be built up from two half-adders, in which case the Boolean algebra expression for the output S is:

$$S = \bar{C}_i .\ S_H + C_i . \bar{S}_H \, ,$$

where S_H is as before, and C_i is the "carry-in" from the adjacent less-significant digits sum. The "carry-out", C_0, which is to be added to the adjacent more significant digits, is given by:

$$C_0 = C_H + C_i \cdot S_H \, ,$$

where C_H is as before. The adder circuit is shown in Fig. 4-3. It can be built from two FCH191 packages (each contains four NAND gates) plus one NAND gate as shown in Fig. 4-4.

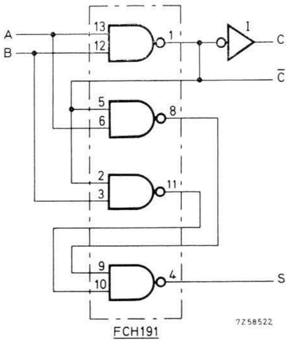


Fig. 4–2. The half-adder circuit showing the most convenient interconnections when using the four-gate FCH191 package plus one gate.

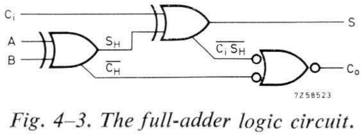


Fig. 4–3. The full-adder logic circuit.

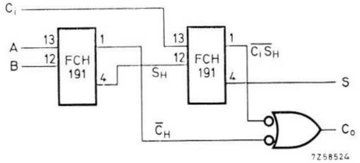


Fig. 4–4. The full-adder circuit interconnections when using two FCH191 packages plus one gate.

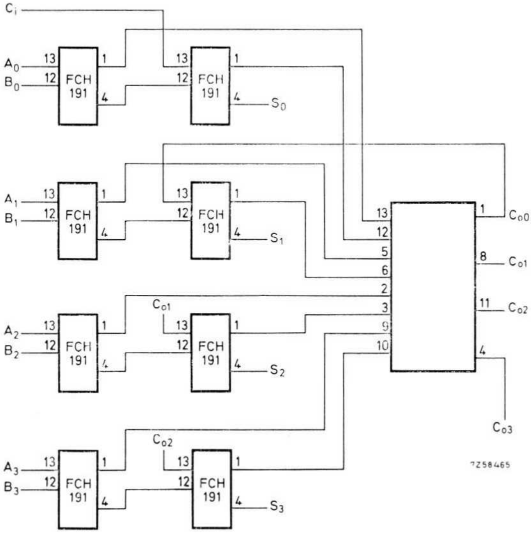


Fig. 4–5. Logic diagram of an adder circuit capable of adding two 4-digit numbers.

Fig. 4-5 shows the interconnections necessary to construct a circuit capable of adding two 4-digit numbers A and B. Let us assume the numbers are of the form $A_3A_2A_1A_0$ and $B_3B_2B_1B_0$, that is, A_3 and B_3 are the most significant and A_0 and B_0 are the least significant digits of numbers A and B respectively. C_i is the carry-in to the first adder (adder of digits A_0 and B_0) and C_{o0} is its carry-out, C_{o0} being at the same time the carry-in to the second adder (of A_1 and B_1). C_{o1}, C_{o2} and C_{o3} are the carry-outs of the second, third and fourth adders respectively, C_{o3} being at the same time the carry-out for the adder as a whole. The outputs are S_0, S_1, S_2 and S_3. Nine FCH191 packages are necessary to make the circuit.

There are other configurations which could be used to achieve the same result. The full adder of Fig. 4-4 which uses nine 2-input NAND gates is re-drawn in Fig. 4-6 using one 2-input and seven 3-input NAND gates. By using Wired-Output as shown in Fig. 4-7, parallel addition of two 12-digit binary numbers can be achieved with only 29 packages—24 of type FCH171, 3 of type FCH191 and 2 of type FCH211.

If we set as criterion a least number of packages however, we find that it is advantageous to use the first method (the 2-input gates of the FCH191), resulting in 27 packages for 12-digit addition. Applying Wired-Output here results in a further saving; the circuit is given in Fig. 4-8 for a single adder. For 12-digit addition, 21 type FCH191 plus 4 type FCH211 packages are necessary.

It is obvious that no clear rule can be given for designing an adder with the least number of packages. In fact, it will often be found that "left-over" gates in a package are usefully employed in other duties in the system. Table 4-1 gives the package usage of the four circuits used as examples above.

Table 4–1. Package Usage of Adder Circuit Examples.

circuit	$\overline{C}_0$ available	$\overline{S}$ available	package usage FCH211	FCH191	FCH171	Total
fig. 4–4	no	no	—	$2^1/_4$	—	$2^1/_4$
fig. 4–6	no	no	—	$^1/_4$	$2^1/_3$	$2^7/_{12}$
fig. 4–7	no	yes	$^1/_6$	$^1/_4$	2	$2^5/_{12}$
fig. 4–8	no	yes	$^1/_3$	$1^3/_4$	—	$2^1/_{12}$

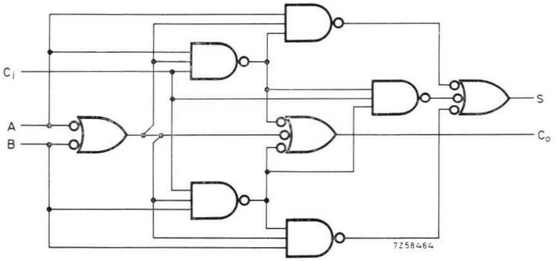


Fig. 4–6. Full-adder logic circuit requiring one 2-input and seven 3-input NAND gates.

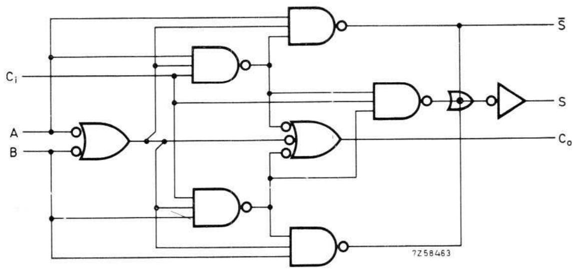


Fig. 4–7. Full-adder logic circuit of Fig. 4–6 using Wired-Output to give the $\overline{S}$ signal without an extra gate.

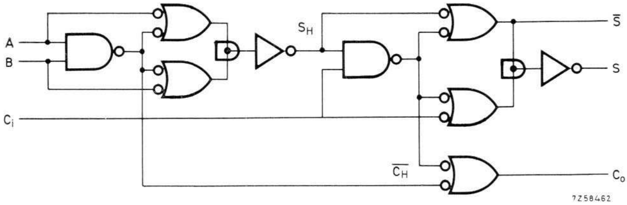


Fig. 4–8. Full-adder logic circuit of Fig. 4–3. using Wired-Output functions to reduce the number of packages necessary.

4.2 Complementers (Inverters)

Besides addition, a calculator is often required to perform substraction. This is achieved with the basic adder circuits of section 4.1 simply by obtaining the complement of the number to be substracted (the subtrahend) and adding this inverted number to the other number. Thus an adder preceded by a complementing circuit will serve either as adder or subtractor. The complementer circuit of Fig. 4-9*a* accepts 4-digit parallel input, and operates as follows.

If signal I is LOW then all inputs A_0 to A_3 that are LOW will cause their respective outputs to go LOW, and all (or any) inputs at HIGH will cause their outputs to go HIGH, that is to say, the input signals appear at the respective outputs. If however I goes HIGH, then all inputs at LOW will cause their respective outputs to go HIGH, and inputs at HIGH cause LOW outputs. In other words, I HIGH causes inversion.

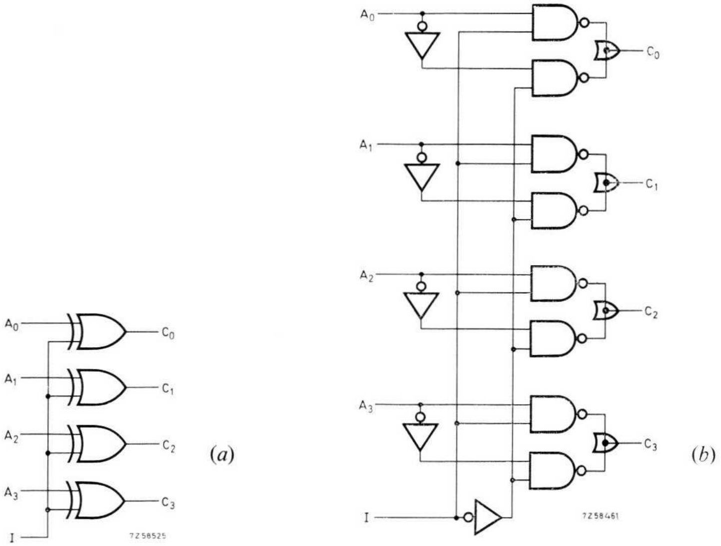


Fig. 4-9. Complementer circuit for binary numbers. (a) Logic; (b) configuration using Wired-Output with two FCH191 and 5/6 FCH211 packages.

This circuit can be formed from four FCH191 packages. Wired-Output can be used, giving the configuration shown in Fig. 4-9*b* (requiring two FCH191 and 5/6 FCH211 packages).

In most desk-top calculators, a binary coded decimal (B.C.D.) arithmetic is used, as opposed to the purely binary numbers we have been discussing so far. Many codes are in use, among which are the NBCD, Excess-Three, and Aiken codes. Complementing numbers written in the latter two codes is an easy matter, as it is only necessary to invert each digit of the subtrahend. Thus the circuit of Fig. 4-9*a* or *b* can be used directly. Complementing NBCD (Natural Binary Coded Decimal) numbers is unfortunately not so easy, as can be seen from Table 4-2. In the table, A is the number, and C its complement.

From Table 4-2 we can see that the least significant digit (A_0) for each decimal number simply has to be inverted to get the least significant digit of the numbers complement, presenting no difficulties. The next significant digit A_1 is found to be the same for the number and its complement, again presenting no problem. Fig. 4-10 shows how the complements C_0 and C_1 are obtained from inputs A_0 and A_1 respectively.

Table 4–2. NBCD Code Numbers and their Complements.

	NBCD code digits																			
significance	A	C	A	C	A	C	A	C	A	C	A	C	A	C	A	C	A	C	A	C
	0	9	1	8	2	7	3	6	4	5	5	4	6	3	7	2	8	1	9	0
$2^0(A_0,C_0)$	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0
$2^1(A_1,C_1)$	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1	0	0	0	0
$2^2(A_2,C_2)$	0	0	0	0	0	1	0	1	1	1	1	1	1	0	1	0	0	0	0	0
$2^3(A_3,C_3)$	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0

Referring to the table again we can formulate a rule relating A and C_2 for each number. C_2 can only be HIGH if A_1 and A_2 are unequal. If $A_1 = A_2$, then C_2 must be at 0-level. In Boolean algebra:

$$I(\bar{A}_1 \cdot A_2 + A_1 \cdot \bar{A}_2) + \bar{I} \cdot A_2 = C_2 \,,$$

where I is the inverter signal. The circuit shown in Fig. 4-10 will perform this function.

Again looking at Table 4-2, the rule relating C_3 to A is the following: C_3 must be HIGH only if A_1, A_2 and A_3 are all LOW. This can be expressed as:

$$\overline{I(A_1 + A_2 + A_3) + \bar{I} \cdot \bar{A}_3} = C_3 \,,$$

which function is performed by the lower part of the circuit in Fig. 4-10.

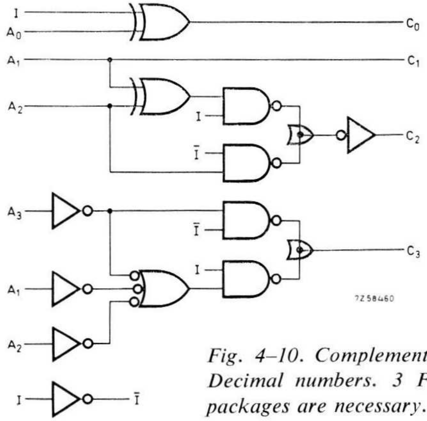


Fig. 4–10. Complementer logic circuit for Natural Binary Coded Decimal numbers. 3 FCH191, 5/6 FCH211 and 1/3 FCH171 packages are necessary.

In total, this circuit, which forms the nine-complement in NBCD code of a decimal digit, requires three FCH191, plus 5/6 of FCH211 plus 1/3 of FCH171. To make full use of all packages therefore a complementer for a multiple of six digits would be necessary. Fig. 4-11 gives another solution; here, two FCH191 plus one FCH121 plus 1/3 of FCH171 are required. Five packages are saved per complementer of six digits, 20 packages being necessary. The 3-input NAND gate used in the A_3-C_3 inversion in Fig. 4-11 could be eliminated by using the circuit of Fig. 4-12; this means however that $16\frac{1}{2}$ of FCH191 and five FCH211 packages, a total of $21\frac{1}{2}$ packages, would be required. Table 4-3 shows package usage of NBCD and Excess-Three code alternative circuits.

Table 4–3. Complementer Circuit Package Usage.

circuit		number of packages			
		FCH171	FCH191	FCH211	Total
NBCD,	fig. 4–10	$^1/_3$	3	$^5/_6$	$4^1/_6$
NBCD,	fig. 4–11	$^1/_3$	2	1	$3^1/_3$
NBCD,	fig. 4–11 + 4–12 combined	—	$2^3/_4$	$^5/_6$	$3^7/_{12}$
Excess-3	fig. 4–9*a*	—	2	$^5/_6$	$2^5/_6$

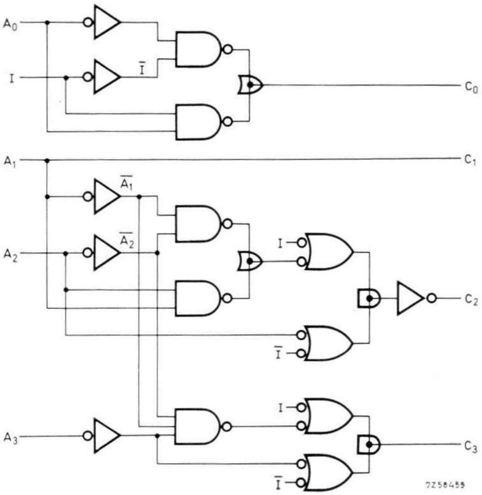


Fig. 4–11. Another configuration of the NBCD code complementer of Fig. 4–10. 2 FCH191, 1 FCH121 and 1/3 FCH171 packages are necessary.

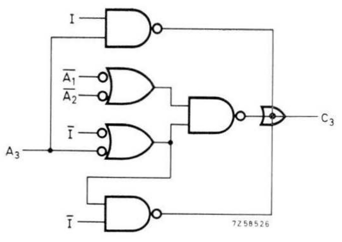


Fig. 4–12. Alternative circuit for the A_3–C_3 complementing part of Fig. 4–11.

4.3 Encoding, Decoding, and Code Correcting Circuits

Usually, the input and output signals associated with the logic system are decimally coded. In addition, the signals within the system itself are sometimes given special codes in order to reduce errors or increase efficiency in some way. Wherever a change of code occurs, an encoding (input to system), decoding (system to output) or code correction (within the system) circuit is necessary. These are discussed separately below.

4.3.1 ENCODERS

Encoders are mostly of the form "decimal-in, binary-out". Table 4-4 lists six common binary codes used for the expression of the ten decimal digits 0 to 9.

Table 4–4. Common Decimal to Binary Codes.

decimal digit	NBCD (1-2-4-8) code	aiken code	1-2-4-2 code I	1-2-4-2 code II	excess-three code	one-change code
0	0 0 0 0	0 0 0 0	0 0 0 0	0 0 0 0	1 1 0 0	0 0 0 0 0
1	1 0 0 0	1 0 0 0	1 0 0 0	1 0 0 0	0 0 1 0	1 0 0 0 0
2	0 1 0 0	0 1 0 0	0 1 0 0	0 1 0 0	1 0 1 0	1 1 0 0 0
3	1 1 0 0	1 1 0 0	1 1 0 0	1 1 0 0	0 1 1 0	1 1 1 0 0
4	0 0 1 0	0 0 1 0	0 1 0 1	0 0 1 0	1 1 1 0	1 1 1 1 0
5	1 0 1 0	1 1 0 1	1 1 0 1	1 0 1 0	0 0 0 1	1 1 1 1 1
6	0 1 1 0	0 0 1 1	0 0 1 1	0 1 1 0	1 0 0 1	0 1 1 1 1
7	1 1 1 0	1 0 1 1	1 0 1 1	1 1 1 0	0 1 0 1	0 0 1 1 1
8	0 0 0 1	0 1 1 1	0 1 1 1	0 1 1 1	1 1 0 1	0 0 0 1 1
9	1 0 0 1	1 1 1 1	1 1 1 1	1 1 1 1	0 0 1 1	0 0 0 0 1

Encoding circuits producing the six codes listed in the table are shown in Figs. 4–13 to 4–18.

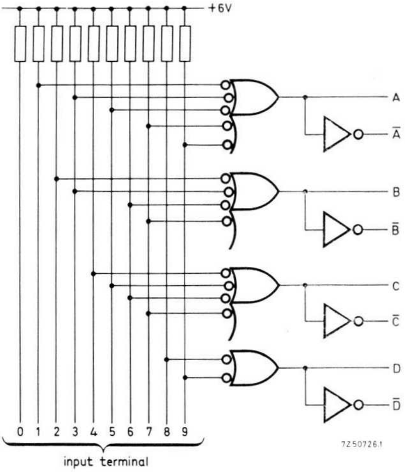


Fig. 4–13. Encoding circuit for decimal to 1248 code.

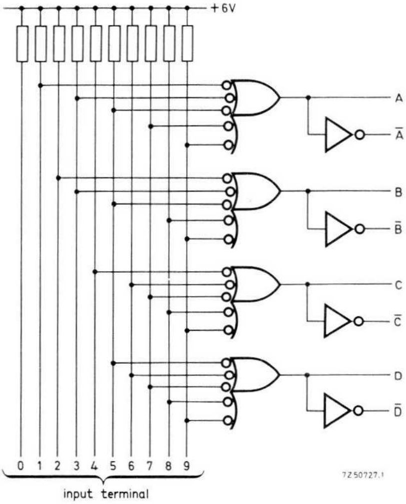


Fig. 4–14. Encoding circuit for decimal to Aiken 1242 code.

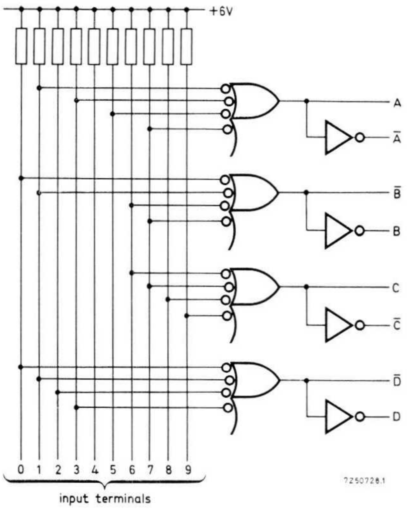


Fig. 4–15. Encoding circuit for decimal to 1242 code (version I).

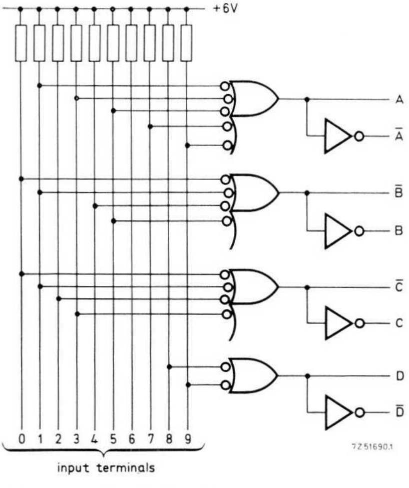


Fig. 4–16. Encoding circuit for decimal to 1242 code (version II).

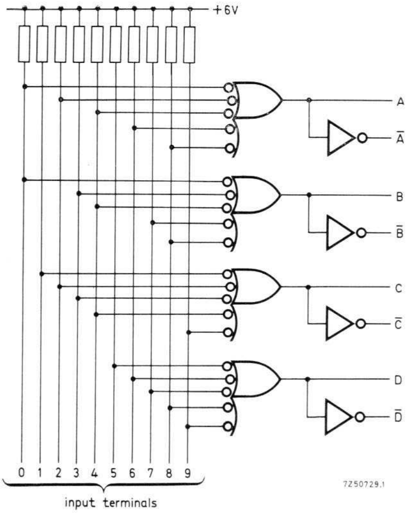


Fig. 4–17. Encoding circuit for decimal to Excess-Three code.

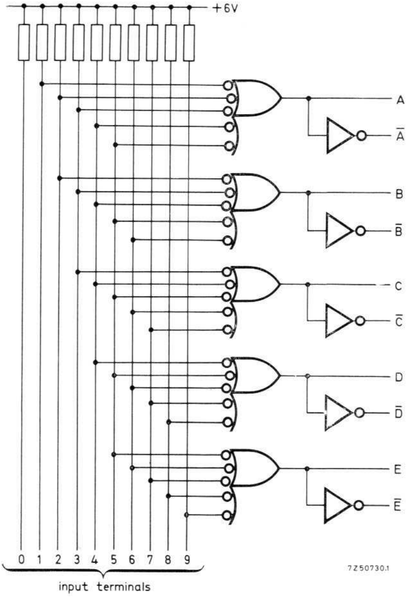


Fig. 4–18. Encoding **circuit** *for decimal to One-Change code.*

4.3.2 DECODERS

Decoders are commonly used in converting one or other binary code used within the system into decimally-coded digits for display or printing. The six diagrams following (Figs. 4-19 to 4-24) show gate configurations giving decimal output for binary input using the six codes given in Table 4-4.

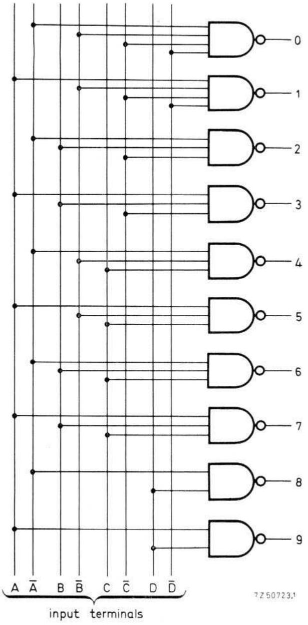


Fig. 4–19. Code converter for 1248 to decimal code.

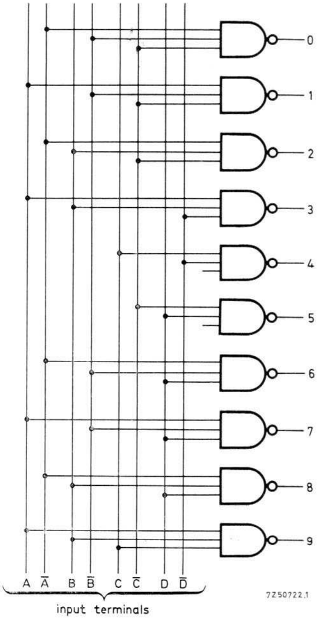


Fig. 4–20.
Code converter for Aiken 1242 to decimal code.

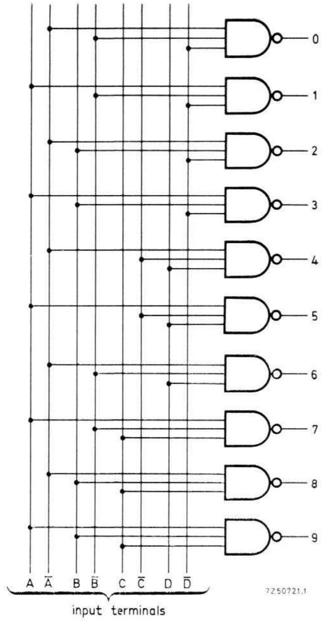


Fig. 4–21.
Code converter for 1242 (version I) to decimal code.

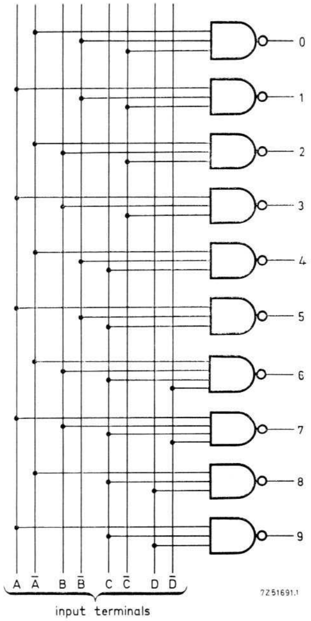


Fig. 4–22.
Code converter for 1242 (version II) to decimal code.

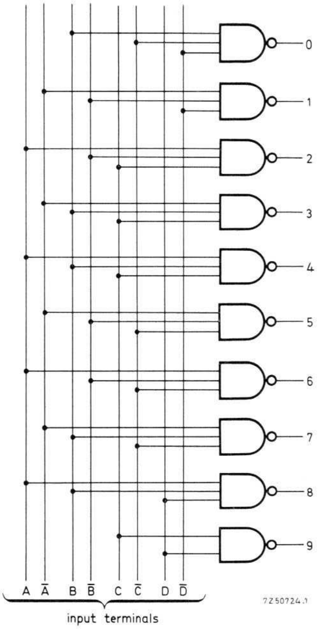


Fig. 4–23.
Code converter for Excess-three to decimal code.

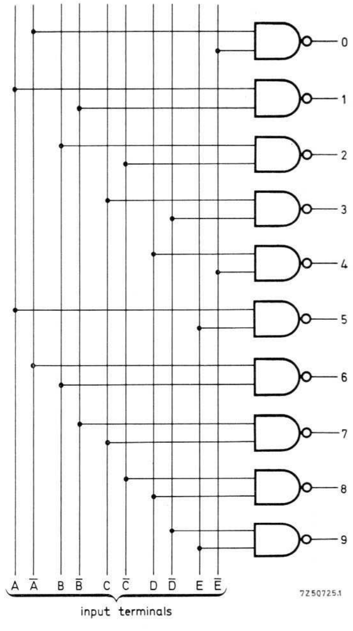


Fig. 4–24.
Code converter for One-change to decimal code.

4.3.3 CODE CORRECTORS

In the adder circuits discussed in section 4.1 it is obviously possible to add two decimal numbers A and B such that their sum S is greater than 9, at which the carry-out signal should be given, together with outputs $S_0S_1S_2S_3$ indicating (in NBCD code) the number $S=10$. For instance, if $A = 6$, $B = 7$ then the carry-out signal should appear ($S > 9$), and S_0, S_1, S_2, S_3 should indicate 1, 1, 0, 0 respectively (NBCD equivalent of 13—10 = 3). However the adder has no way of sensing whether S is greater than 9 so that the carry-out signal is produced only when S is greater than 15. What actually occurs is presented under the "Binary Sum" column in Table 4-5; what *should* happen is given under the "NBCD Sum" column of the table.

The circuit shown in Fig. 4-25 performs this conversion, the outputs being now S_{N0}, S_{N1}, S_{N2}, S_{N3} (in order of significance), and the carry-out terminal, at high level when the sum S_N is greater than 9, being C_{10}. It is assumed that the adder circuit preceding it has the inverted S outputs $\bar{S}_1$, $\bar{S}_2$ and $\bar{S}_3$ available. If not, the code corrector circuit must be provided with the inverters shown dotted in Fig. 4-25. Without them, the

Table 4–5. Conversion of Binary Sum to NBCD Sum

decimal	binary sum (see Fig. 4.5)					NBCD sum				
sum	S_0	S_1	S_2	S_3	C_{03}	S_{N0}	S_{N1}	S_{N2}	S_{N3}	C_{10}
0	0	0	0	0	0	0	0	0	0	0
1	1	0	0	0	0	1	0	0	0	0
2	0	1	0	0	0	0	1	0	0	0
3	1	1	0	0	0	1	1	0	0	0
4	0	0	1	0	0	0	0	1	0	0
5	1	0	1	0	0	1	0	1	0	0
6	0	1	1	0	0	0	1	1	0	0
7	1	1	1	0	0	1	1	1	0	0
8	0	0	0	1	0	0	0	0	1	0
9	1	0	0	1	0	1	0	0	1	0
10	0	1	0	1	0	0	0	0	0	1
11	1	1	0	1	0	1	0	0	0	1
12	0	0	1	1	0	0	1	0	0	1
13	1	0	1	1	0	1	1	0	0	1
14	0	1	1	1	0	0	0	1	0	1
15	1	1	1	1	0	1	0	1	0	1
16	0	0	0	0	1	0	1	1	0	1
17	1	0	0	0	1	1	1	1	0	1
18	0	1	0	0	1	0	0	0	1	1

circuit can be built from two FCH191 plus 2/3 of FCH171 plus 1/3 of FCH211. The three inverters would require half the FCH211 package.

The above assumes that the output sum is required in NBCD code, but circuits can be built to give an output in other codes. In the following example this is the Excess-Three code. Table 4-6 gives the conversion required.

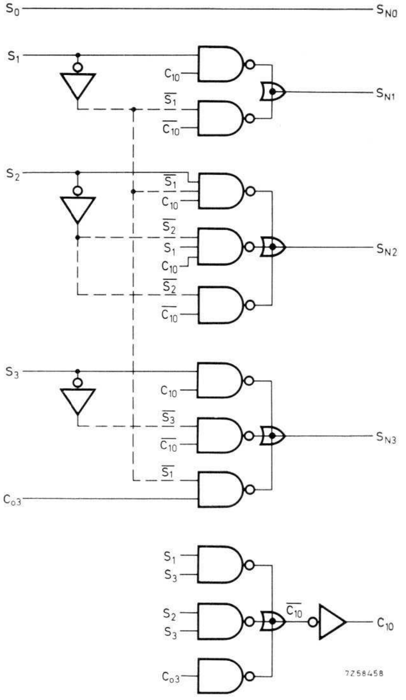


Fig. 4–25. Code corrector circuit for binary to NBCD code.

The conversion is most easily done in this case by placing a second adder circuit in series with the "real" adder, as shown in the block diagram of Fig. 4-26. The second adder corrects the first by adding 3 if a decimal carry is to occur, and subtracting 3 if it is not; its output carry C_{ea} ("carry end around") is fed back to its input. If the output sum S_E is negative, it will appear in complementary form in Excess-Three code, indicated by the next significant sum digit when used as a sign digit.

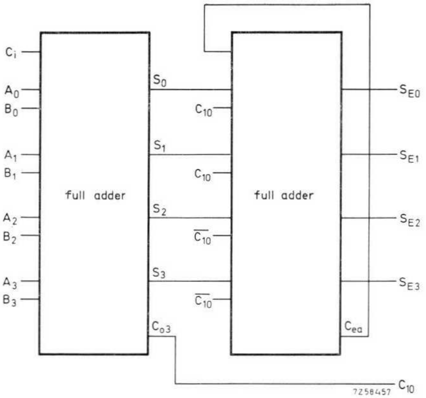


Fig. 4–26. Code corrector circuit for binary to Excess-Three code.

Table 4–6. Conversion of Binary Sum to Excess-Three Sum

decimal	binary sum					excess-three sum				
sum	S_0	S_1	S_2	S_3	C_{03}	S_{EO}	S_{E1}	S_{E2}	S_{E3}	C_{10}
0	0	1	1	0	0	1	1	0	0	0
1	1	1	1	0	0	0	0	1	0	0
2	0	0	0	1	0	1	0	1	0	0
3	1	0	0	1	0	0	1	1	0	0
4	0	1	0	1	0	1	1	1	0	0
5	1	1	0	1	0	0	0	0	1	0
6	0	0	1	1	0	1	0	0	1	0
7	1	0	1	1	0	0	1	0	1	0
8	0	1	1	1	0	1	1	0	1	0
9	1	1	1	1	0	0	0	1	1	0
10	0	0	0	0	1	1	1	0	0	1
11	1	0	0	0	1	0	0	1	0	1
12	0	1	0	0	1	1	0	1	0	1
13	1	1	0	0	1	0	1	1	0	1
14	0	0	1	0	1	1	1	1	0	1
15	1	0	1	0	1	0	0	0	1	1
16	0	1	1	0	1	1	0	0	1	1
17	1	1	1	0	1	0	1	0	1	1
18	0	0	0	1	1	1	1	0	1	1

The output carry C_{10} of the most significant digit (the sign digit) must be fed back as the carry-in of the least significant digit. Thus the code corrector here is simply a complete 4-digit adder, composed of from 8 to 10 packages depending on the configuration used.

Other code correctors are possible requiring less packages. The example given in Fig. 4-27 (again for Excess-Three code output) requires $5^{3}/_{4}$ packages of FCH191 plus $^{5}/_{6}$ of FCH211, making $6^{7}/_{12}$ packages in total (the four exclusive-OR gates each taking one FCH191 package). Use of the simplified exclusive-OR circuit given in Fig. 4-9*b* would result in a requirement of $3^{3}/_{4}$ of FCH191 plus $1^{1}/_{2}$ of FCH211, making $5^{1}/_{4}$ packages total.

Code correction networks for other codes are more complicated than those given above, and as they are not so commonly used we refer the reader to specialized literature on the subject. Table 4-7 shows the package usage of NBCD and Excess-Three code circuits. The numbers between brackets in this table refer to the case where the inverted sum of the binary adders is available, for instance in Figs. 4-7 and 4-8.

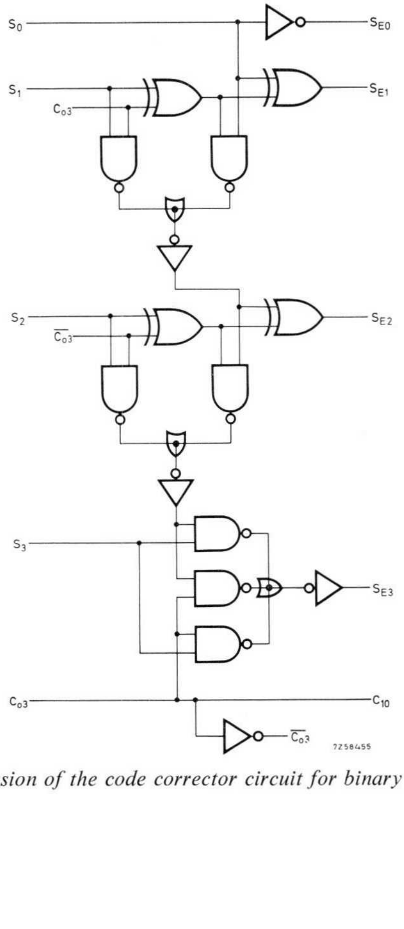


Fig. 4–27. Another version of the code corrector circuit for binary to Excess-Three code.

Table 4–7. Code Correction Circuit Package Usage.

circuit		number of packages			
		FCH171	FCH191	FCH211	Total
NBCD,	fig. 4–25	$^{2}/_{3}$	2	$^{5}/_{6}$ (1/3)	$3^{1}/_{2}$(3)
excess-3,	fig. 4–27	—	$5^{3}/_{4}$	$^{5}/_{6}$	$6^{7}/_{12}$
excess-3,	fig. 4–27 (simplified)	—	$3^{3}/_{4}$	$1^{1}/_{2}$	$5^{1}/_{4}$

4.4 Complete Adder System Using Combination Logic

From sections 4.1, 4.2 and 4.3 it is now possible to design a complete adder/subtractor system meeting most individual needs. As an example, let us find the combination of adding, complementing, and code-correcting circuits which will give the least number of integrated circuit packages. A block diagram of the complete system is shown in Fig. 4-28.

The circuits of Figs. 4-8, 4-11 and 4-25 are the most economical in this respect if NBCD code is chosen (see Tables 4.1, 4.3 and 4.7), giving a total of $11^{5}/_{12}$ packages. For Excess-Three code, the circuits of Figs 4-8, 4-9 and 4-27 (simplified) are best, giving 13 packages.

A negative sum at the output of the system is given in the complementary form; if this is not acceptable, an extra complementer will have to be added to invert the output to the true value whenever the subtrahend is greater than the minuend. A device could be added to indicate a minus sign at this occurrence.

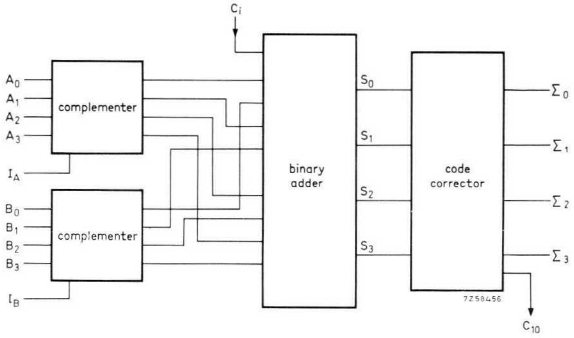


Fig. 4–28. Block diagram of a complete adder system.

5 Interface Circuits - Input to FC Logic

The conversion of input signals from outside the logic system to binary levels suitable for use within the system can be done in a number of ways. In general, the approach is as follows: first amplify the input signals at the source, transmit the amplified signals over a shielded cable to the logic system, and finally terminate the cable at a level detector such as the FCL101 before actually entering the logic. The level detector will ensure that no indeterminate signal (i.e. one that floats somewhere between HIGH and LOW) can cause oscillation or other upset in the logic. Often, some kind of filter is inserted in the line to prevent noise affecting the logic circuits.

Four common input devices are mechanical switches, photocells, VSOs and EPDs.

5.1 Mechanical Switches

Although the shielded cable in the above arrangement will exclude most external noise, unwanted signals can still enter the system at the input device and thus affect the logic operation. Such unwanted signals frequently occur at the closing or opening of a mechanical switch contact. Three simple interface circuits giving different degrees of protection are shown in Figs 5–1, 5–2 and 5–3.

The circuit of Fig. 5–1 uses a capacitor C to extend the time of input level change at the gate. The value of C must be chosen so that after operating the switch the contacts will have stopped bouncing before the gate input signal has changed sufficiently to give a changed output. If the delay is made long, system speed can be excessively affected and unstable operation can result.

The circuit of Fig. 5–2 requires a two-pole switch. The cross-coupled NAND gates form a memory or bistable circuit. With S in position 1, output Q_1 must be at HIGH. Upon operating S, Q_1 will remain at HIGH until the switch blade actually touches contact 2. Q_2 will then go to HIGH, making Q_1 LOW. Contact bounce will not affect this state, because contact 1 must actually be touched by the switch blade again (brought to 0 V) before Q_1 can be brought back to HIGH. A disadvantage of this

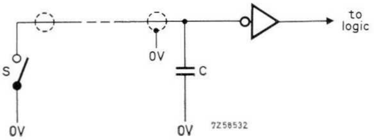


Fig. 5–1. Interface between switch S and the logic, using a by-pass capacitor.

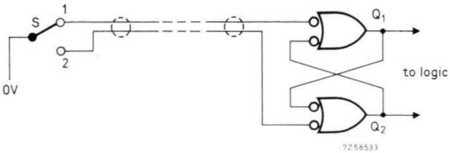


Fig. 5–2. Interface between switch S and the logic, using a flip-flop "memory".

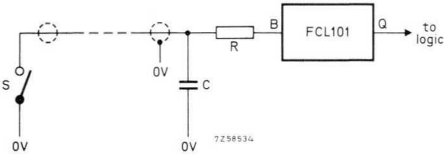


Fig. 5–3. Interface between switch S and the logic, using an R–C combination plus a Level Detector.

arrangement is that the input impedance of the NAND gates is high thus making the transmission cable prone to interference but, system speed is not diminished.

In Fig. 5–3, an *R–C* delay network cuts down the high-frequency noise present in the signal while the Level Detector FCL101 ensures that the resultant slowly-changing signal level gives predictable operation of the logic. This arrangement enables speed to be increased over that given in Fig. 5–1, because the FCL101 can be set to give a well-defined output at quite a low input voltage which will, nevertheless, be high enough to distinguish reliably between true signals and noise.

5.2 Photocells

Many types of photocells are available for providing input signals. Two self-contained units are available: the Light Interruption Probe (LIP1) and the Photoelectric Detector/Lamp Unit (CSPD/1MLU) having maximum detection frequencies of 10 kHz and 6 Hz respectively. These units are shown in Figs. 5–4 and 5–5.

For situations where these units do not meet every requirement, a circuit built up from discrete components and using for example a light-dependent resistor (LDR) photodiode or phototransistor (e.g. the BPX25) can usually be designed.

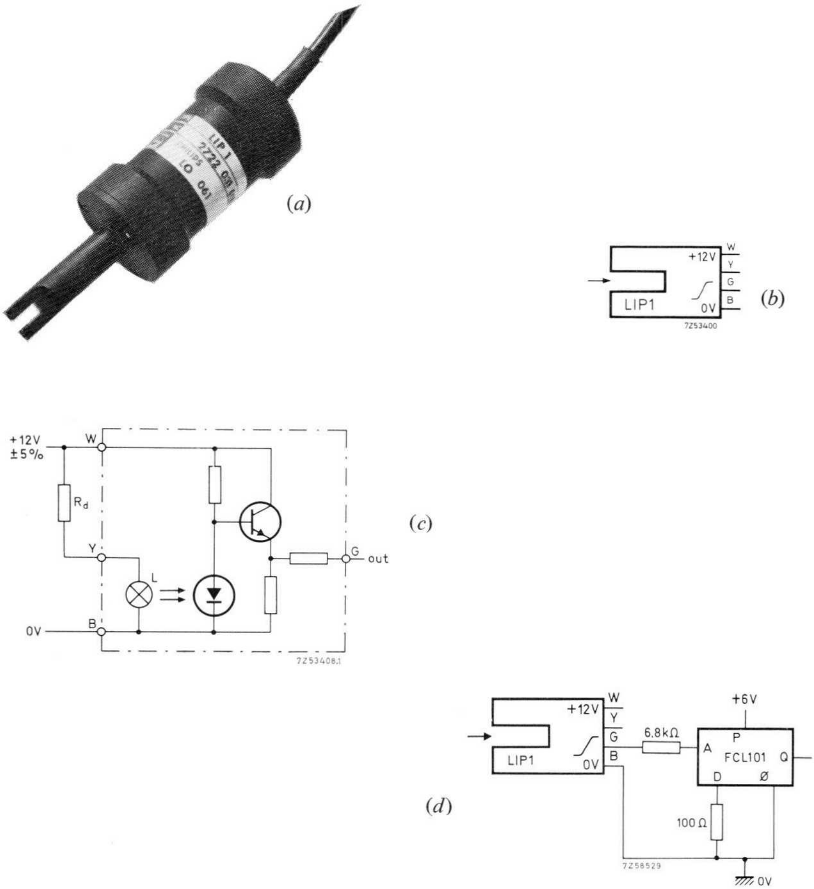


Fig. 5–4. The Light Interruption Probe, LIP1. (a) Photo of the unit; (b) symbol; (c) circuit diagram; (d) interface circuit.

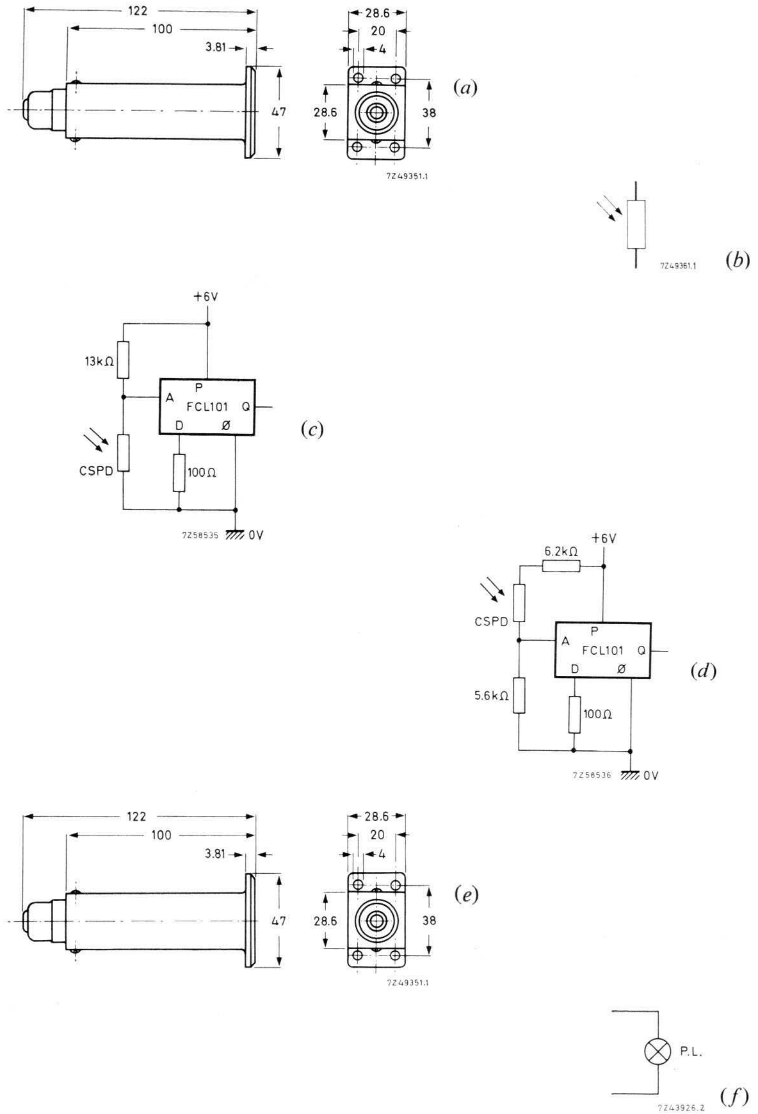


Fig. 5–5. The Photoelectric Detector, CSPD and Lamp Unit (1MLU); (a) Drawing of CSPD; (b) CSPD symbol; (c, d) interface circuits (e) drawing of1MLU; (f) symbol of 1MLU.

5.3 The Vane Switched Oscillator (VSO)

The VSO is illustrated in Fig. 5–6*a* and its symbol and dimensions (in mm) are given in Figs 5–6*b* and *c*. Its catalogue number is 2722 031 00001.

The left-hand part of the circuit (see Fig. 5–6*d*) is an oscillator. Coils L_1 and L_2 are physically separated by the gap in the VSO. A piece of metal entering this gap will reduce the feedback from L_2 to L_1, and consequently lower the voltage across L_3. This voltage is then rectified and filtered and taken to terminal 3, whose voltage is thus reduced with respect to terminal 4. The supply voltage is +12 V ±10%; current drain, 12 mA. The output has an impedance of 4.1 kΩ and delivers 5.75 V ± 15% (non-activated).

The unit is capable of detecting objects passing at a frequency of 1 kHz. Any metal can be used as the activating vane. The VSO has a wide field of application: revolution counting, angular position detection, counting of small objects, and so on.

5.4 The Electronic Proximity Detector (EPD)

The EPD is illustrated in Fig. 5–7*a*. Its symbol and dimensions (in mm) are shown in Figs 5–7*b* and *c* . Its catalogue number is 2722 031 00021. In the EPD circuit (Fig. 5–7*d*), TR_1 and L_1 and L_2 form an oscillator which is coupled to the detector stage TR_2. Normally TR_2 rectifies the oscillator output, thereby making the base voltage of TR_3 negative with respect to its emitter (TR_3 conducting). This brings the output voltage to about +12 V. Reducing the oscillation by inserting a metal object (its shape is irrelevant) into the field of the sensing coil reduces the output voltage.

The supply voltage required is 12 V ± 5%; current drain 16 mA. The non-activated unit has an output impedance of 680 Ω and will deliver a voltage equal to supply voltage minus 0.5 V (approx.). The output impedance is 3.3 kΩ upon activation.

The unit is capable of detecting objects passing at a frequency of 1 kHz. Any good electrical conductor (such as steel, aluminium) can be detected.

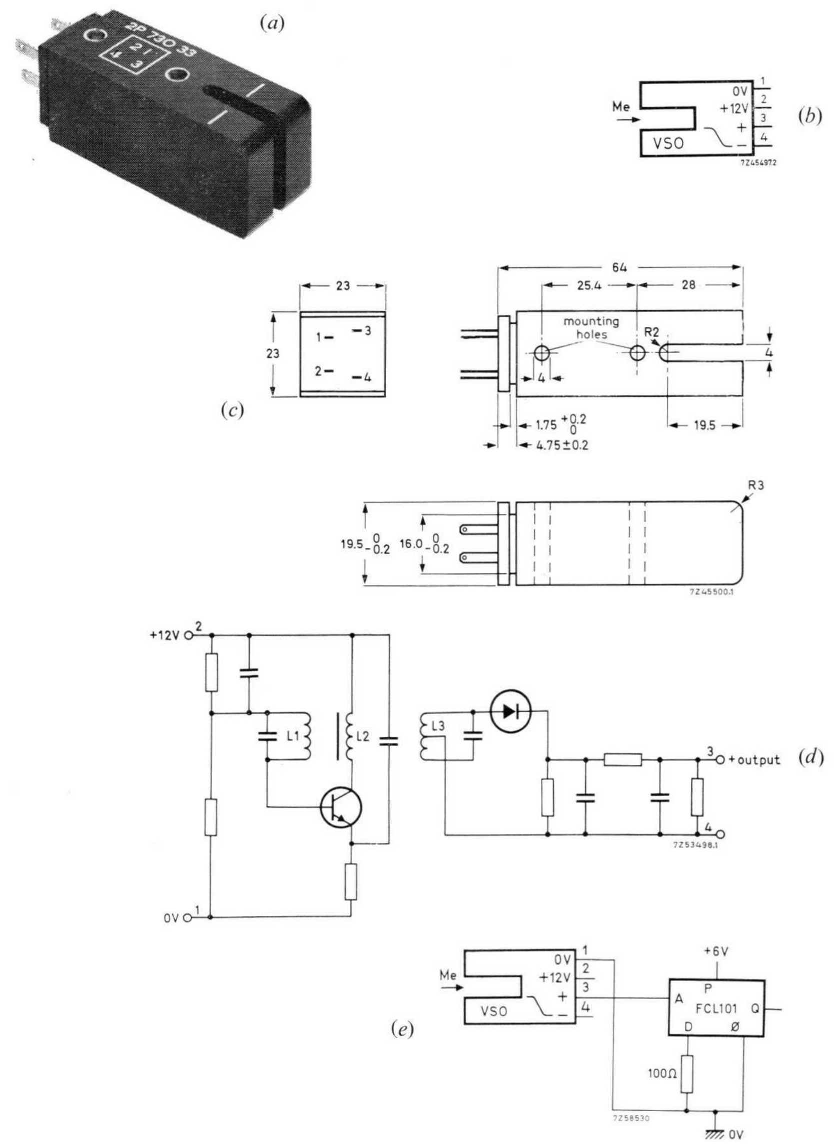


Fig. 5–6. The Vane Switched Oscillator, VSO. (a) Photo; (b) symbol; (c) dimensions; (d) circuit diagram; (e) interface.

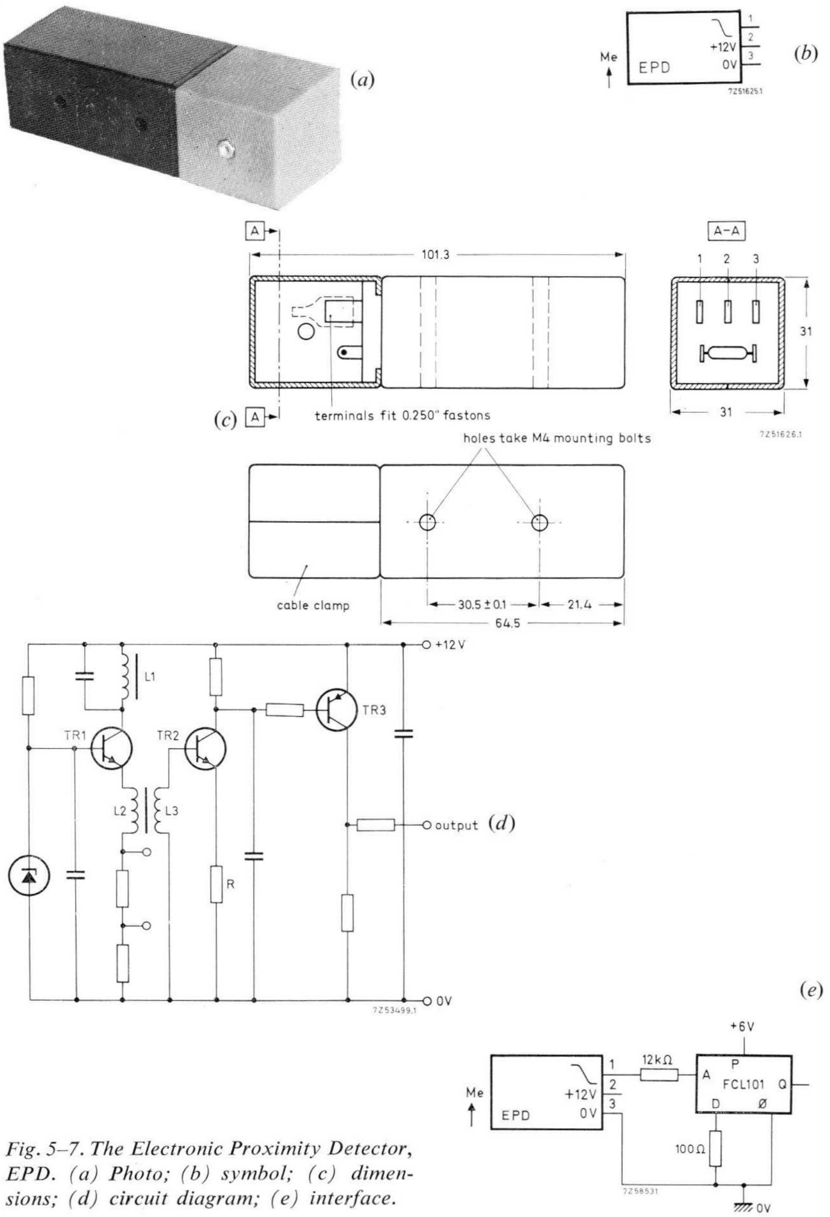


Fig. 5–7. The Electronic Proximity Detector, EPD. (a) Photo; (b) symbol; (c) dimensions; (d) circuit diagram; (e) interface.

6 Interface Circuits — FC Logic to Output

The results of the logic operations performed in the system by the integrated circuits must be presented to the user — whether another circuit, a machine, or humans — in a usable and useful form. Three broad classes of interface output circuit are described in this chapter: section 6.1, Thyristors; section 6.2, Transistors; and section 6.3, Numerical Indicator Tubes.

6.1 Interface With Thyristors

The term "thyristor" covers semiconductor devices operating on the p-n-p-n regenerative principle; included in the family are the Silicon Controlled Rectifier (SCR), Silicon Controlled Switch (SCS), Triac, Diac, and many more. The various devices have different capabilities (some conducting only in one direction while others conduct in both directions) and different triggering requirements. As an example the circuit of Fig. 6–1 is given to illustrate drive of SCRs by integrated

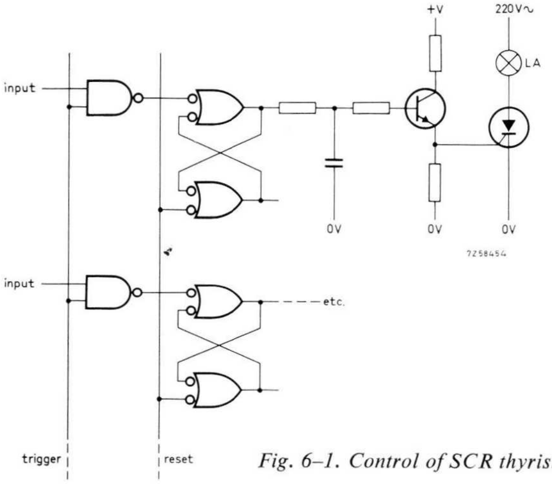


Fig. 6–1. Control of SCR thyristors with integrated circuits.

circuits. Triggering requirements for SCRs are in the range 2 V—5 V at 10 mA—70 mA (pulse width down to 10 μs).

In Fig. 6–1, the lamp *LA* is to be lit. A "light" HIGH signal is applied to one input of the first NAND gate and a HIGH trigger signal on the other input will cause its output to go LOW. The output of the memory formed by two crosscoupled gates is set HIGH and the transistor conducts. The SCR is triggered by the voltage appearing across the emitter resistance and current flows through the lamp.

This circuit was designed to drive a bank of display lamps. By replicating the circuit as many times as there were lamps, letters or words could be formed on the display panel by proper selection of inputs prior to triggering. Erasure is obtained by resetting the memory units by a common LOW signal. The capacitor hinders the effects of noise entry.

6.2 Interface with Transistors

Drive of transistors for increased output of the logic system is described below. The circuit of Fig. 6–2 has a maximum LOW output current of 100 mA, while that of Fig. 6–3 has 1000 mA capability (Maximum HIGH

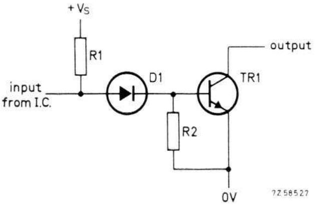


Fig. 6–2. R_1 = 560 Ω ± 5%
R_2 = 6.8 kΩ ± 5%
S_1 = BAX13
S_2 = BFY50.

Fig. 6–2. Driving a transistor amplifier for increased output (100 mA at LOW state).

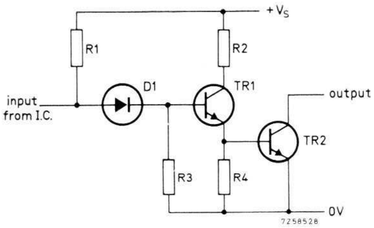


Fig. 6–3. R_1 = 560 Ω ± 5%
R_2 = 30 Ω ± 5%
R_3 = 6.8 kΩ ± 5%
R_4 = 470 Ω ± 5%
S_1 = BAX13
S_2 = BFY51
S_3 = BFY50; heatsink
thermal resistance
<0.07 degC/W.

Fig. 6–3. Driving a transistor amplifier for increased output (1000 mA at LOW state).

output voltage in both cases: 35 V). Supply voltage is + 6 V ± 5 % for both circuits. The following data is valid for the circuit of Fig. 6–2:

at LOW input, circuit output voltage is 35 V max.
circuit output current is 5 μA max.
at HIGH input, circuit output voltage is 0.2 V max.
circuit output current is 100 mA max.

The following data is valid for the circuit of Fig. 6–3:

at LOW input, circuit output voltage is 35 V max.
circuit output current is 5 μA max.
at HIGH input, circuit output voltage is 1.0 V max.
circuit output current is 1000 mA max.

6.3 Interface with Numerical Indicator Tubes

Presentation of data by means of numerical indicator tubes has become increasingly popular since the introduction of inexpensive versions of the tubes. In this section we give ways of controlling these tubes using certain members of the FC family. When compared to conventional circuits using gates, flip-flops and discrete components, the circuits given are cheaper and simpler. Static and dynamic display (parallel and serial display respectively) are both possible. Some advantages of static display are that no extra discrete components are necessary to drive the tubes, and it is never difficult to extend an existing static display to include more decades. The main advantages of dynamic display are that fewer integrated circuits are necessary for displays of more than three decades, and that the multi-decade numerical indicator tube ZM 1200 ("Pandicon") can be used.

The FCL111 is used in all the circuits described. It is a decoder for the NBCD, Excess-3, and Aiken codes, followed by ten output driver stages and specially developed for the ZM series of indicator tubes. The outputs of the FCL111 are capable of driving one indicator tube directly. (For dynamic display, where one FCL111 can be used to drive a number of tubes, clamping diodes are necessary.)

Figs 6–4, 6–5*a* and 6–6*a* show the interconnections necessary between tube and ICs for NBCD, Excess-3, and Aiken coded inputs respectively. The function of counter is fulfilled by the FCJ141, a four-stage asynchronous decade counter producing NBCD-coded outputs, if NBCD code is chosen for presentation to the FCL111 (Fig. 6–4). If, however,

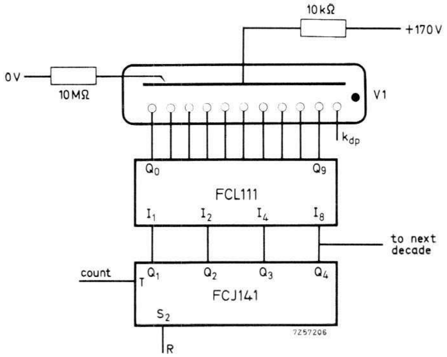


Fig. 6–4. Drive of numerical indicator tube V_1 (type ZM1000, ZM1020 or ZM1080) with FCJ141 and FCL111 (NBCD code). R = reset.

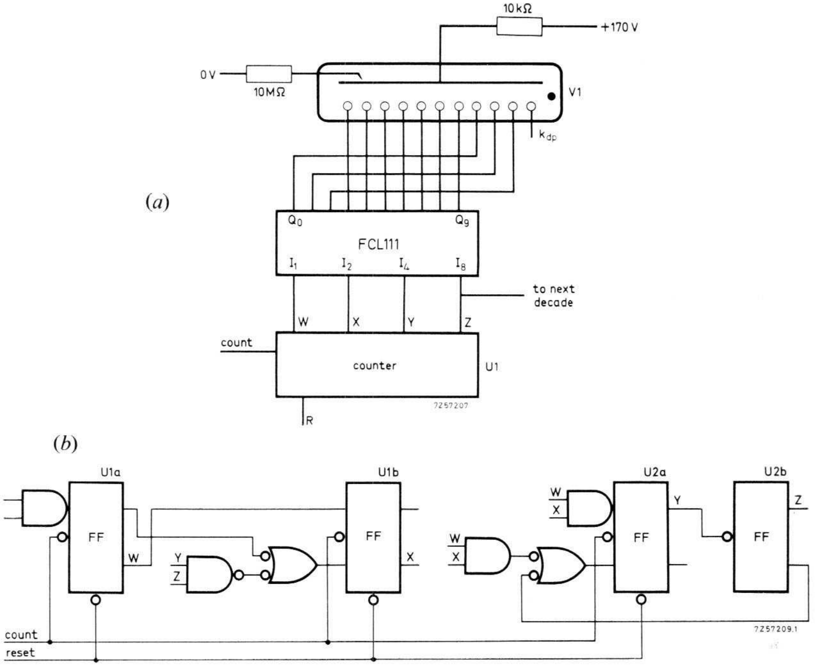


Fig. 6–5. (a) Drive of V_1 with FCL111 and counter U_1 in Excess-Three code; (b) counter U_1 of Fig. 6–5a. (Flip-flops U_{1a}, U_{1b}, U_{2b}, U_{2a}: two FCJ121; gates: one FCH191.) R = reset.

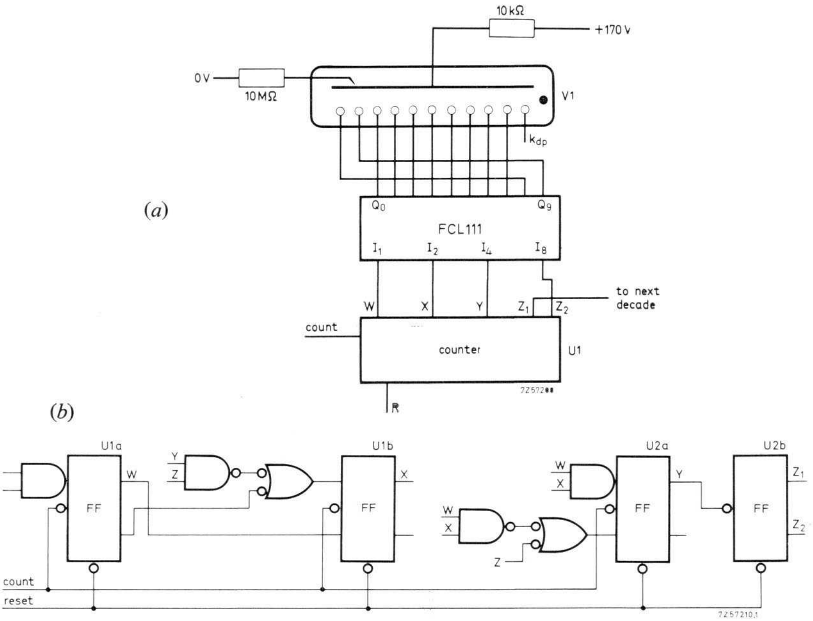


Fig. 6–6. (a) Drive of V_1 with FCL111 and counter U_1 in Aiken code; (b) counter U_1 of Fig. 6–6a. (Flip-flops U_{1a}, U_{1b}, U_{2a}, U_{2b}: two FCL121; gates: one FCH191.) R = Reset.

the Excess-3 or Aiken codes are used in the intermediate stage, counters producing these codes can be made from the FCH191 and FCJ121 types (or only FCJ191, if presetting is required), as shown in Figs 6–5*b* and 6–6*b*.

Fig. 6–7 shows a method for driving a number of indicator tubes for static display. The circuit makes use of three types of IC (one of each per tube): FCL111, FCJ221 and FCJ141. The FCJ221 is a quadruple latch flip-flop, ideally suited for use as a buffer stage between processor and tube so that there is no interruption to the counting process. A stationary image of the digits to be displayed is obtained by applying a pulse T_2 to the FCJ221 without stopping the count input pulses T_1. The comparative timing of pulses T_1 and T_2 is given in Fig. 6–8. In this diagram the symbol t_{pdfQ3} represents the maximum fall propagation delay of the FCJ141 (fall propagation delay at output Q_3). The fall propagation delay at output Q_4 after pulse T_1 is t_{pdfQ4} (delay t_{pdfQ4} approximately equal to two-thirds delay t_{pdfQ3}). Time t_{tot} must be equal to or greater than $^1/_3\,(2\,n + 1)\,t_{pdfQ3}$.

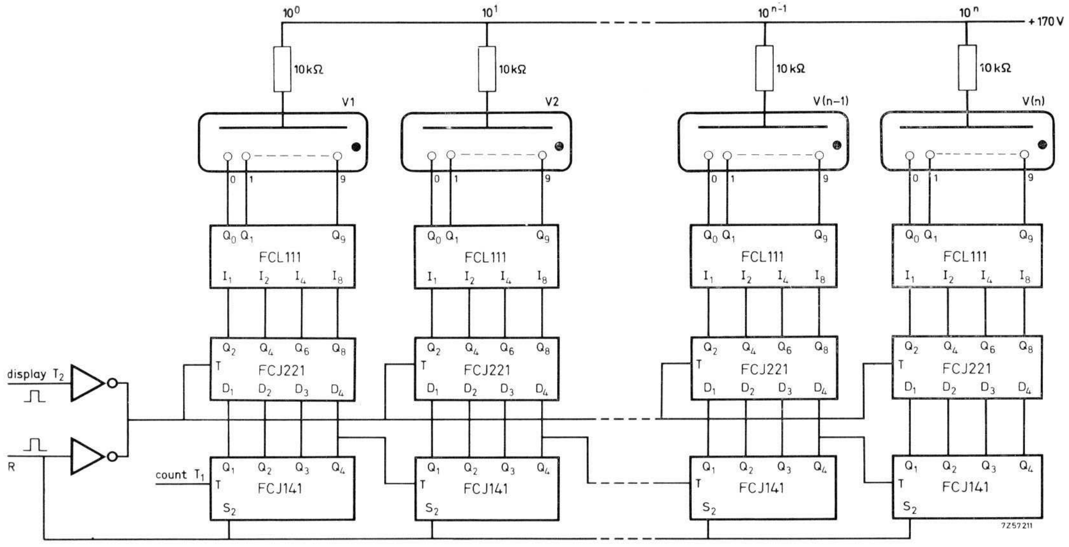


Fig. 6–7. Static drive of numerical indicator tubes V_1–V_n. R = reset. Although only four tubes are shown, any number can be driven.

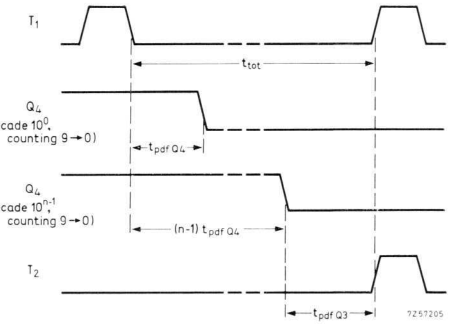

T1
t_tot
Q4
(decade 10^0, counting 9 → 0)
t_pdf Q4
Q4
(decade 10^n-1, counting 9 → 0)
(n-1) t_pdf Q4
T2
t_pdf Q3
7Z57205

Fig. 6–8. Timing diagram of pulses T_1 and T_2.

Fig. 6–9 shows a way in which dynamic drive can be applied to a number of indicator tubes. There are two types of circuits used for dynamic drive: cathode scanning and anode scanning. The circuit given is for anode scanning; this method is preferable for display of the contents of passive registers, such as magnetic core memories and shift registers. The circuit uses a total of six integrated circuits to drive up to ten tubes, and is thus more sparing of ICs than static drive, for more than two tubes. The ICs are: FCH301, FCJ141, FCL111, FCJ221 and two FCJ211. After a shift pulse T_1 from the control is applied to the 4-bit shift register (FCJ211), the parallel input information is transferred to the output of this register (if serial output is to be used, four shift pulses are necessary). A second pulse T_2 applied to the FCJ221 transfers its contents to the input of the decoder/driver (FCL111), and the cathode line corresponding to the decimal information in the shift register will be activated. Pulse T_2 is also applied to the decade counter FCJ141, increasing the count by 1 and driving the next anode switching transistor (516BSY) into conduction. If the shift register contains the number *m*, and the decade counter is at the count of *n*, then the *n*th indicator tube will display the number *m*.

If more or less than ten indicator tubes are required, the FCJ141 must be replaced by a counter which will count up to the required number.

Many designs of counter are possible, some of the most suitable being given in chapter 3. Detailed information about dynamic display systems with numerical tubes is given in the following publications: "Cold Cathode Numerical Indicator Tubes" (ordering code 9399 244 32701)

"Numerical Indicator Tube Drive with Transistor Type BSX21" (ordering code 22-188); "Principle and Circuitry of a Dynamic Information Display System using Numerical Indicator Tubes" (ordering code 22-075).

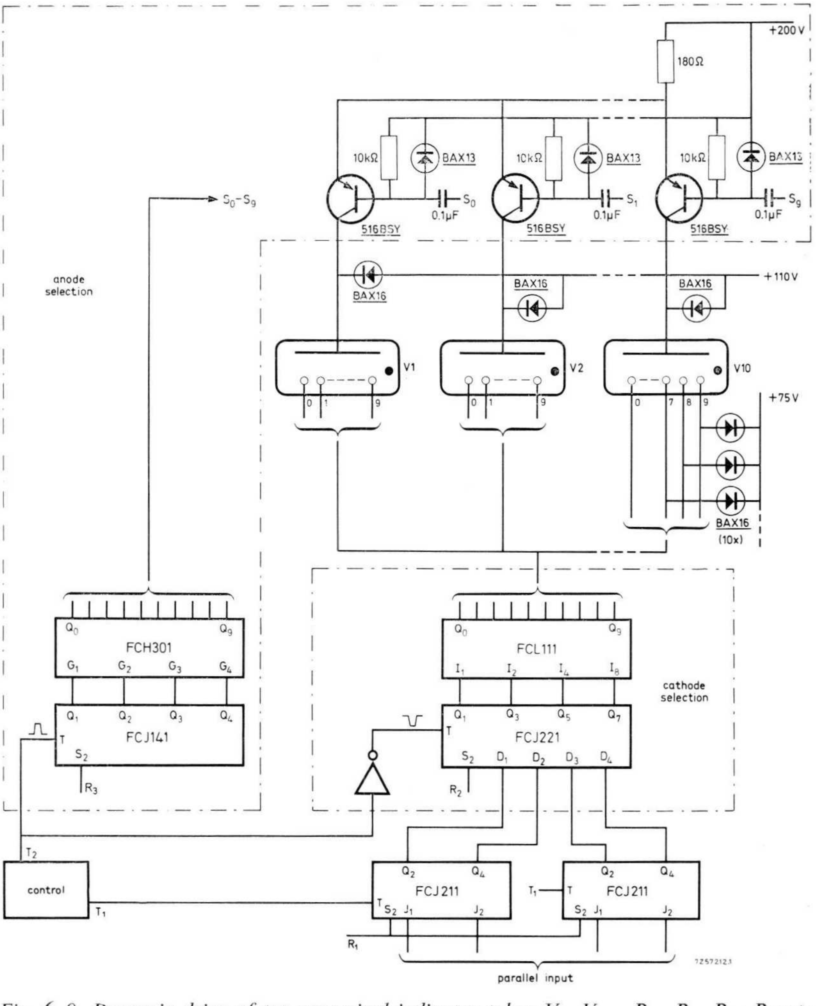


Fig. 6–9. Dynamic drive of ten numerical indicator tubes V_1–V_{10}. R_1, R_2, R_3: Resets. For simplicity, only three tubes are shown.

7 Mounting Methods and Hardware

This chapter deals with the hardware and accessories necessary for the physical application of the FC family of integrated circuits. Section 7.1 describes methods of mounting the integrated circuits on printed wiring boards and chassis. Section 7.2 deals with printed wiring boards and other hardware, and section 7.3 deals with power supply.

7.1 Mounting of Integrated Circuit Packages

The FC family consists of integrated circuits contained in packages of two forms: the Dual-In-Line (DIL) package type XG14 or XG16 for standard temperature range ICs, and the Flat-Pack type XE14 for extended temperature range types.

7.1.1 DIL PACKAGE MOUNTING

DIL packages may be mounted either directly on the printed wiring board or indirectly via a special socket which is in turn fixed on to the board. The XG14 is a 14-pin DIL having a pin spacing of 2.54 mm (0.1 inch). Its dimensions are given in mm in Fig. 7–1. The XG14 is designed to be used with printed wiring boards having the standard *e* grid hole pattern ($e = 0.1$ inch) and a row spacing of $3e$. The XG16 has 16 pins but is otherwise similar to the XG14.

The second method of fixing mentioned above, i.e. using an intermediate socket, is very useful in cases where it is anticipated that circuit changes are to be made, for instance, when developing a layout. The socket (cat. no. 4322 026 69000) is shown in Fig. 7–2. It has gold-plated terminals.

A special miniature PWB is available for use with the socket to facilitate testing of DIL Integrated Circuits. This board has the catalogue number 4322 026 39710 and is shown in Fig. 7–3.

7.1.2 FLAT-PACK MOUNTING

Flat-Pack ICs (see Fig. 7–4) may either be mounted directly on the PWB

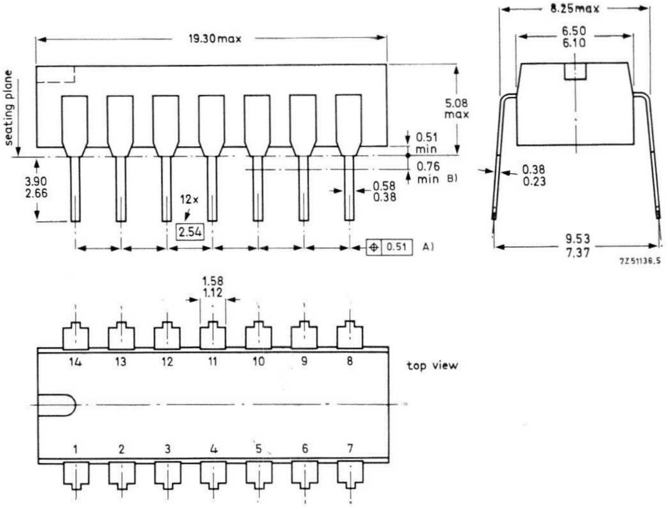


Fig. 7–1. Dimensions of the standard 14-pin DIL package type XG14 (in mm).

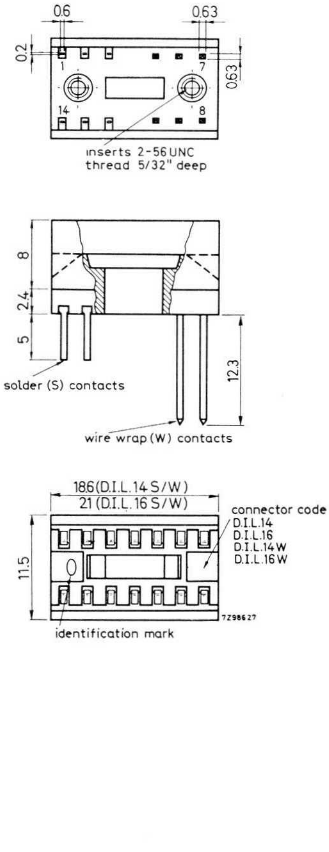


Fig. 7–2. Socket for the XG14 package (dimensions in mm).

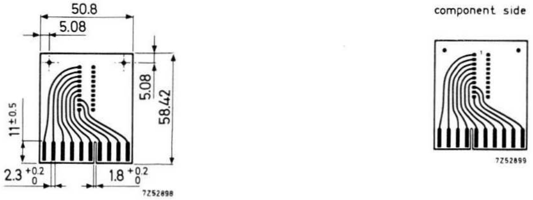


Fig. 7–3. Special printed wiring board for use with the XG14 socket.

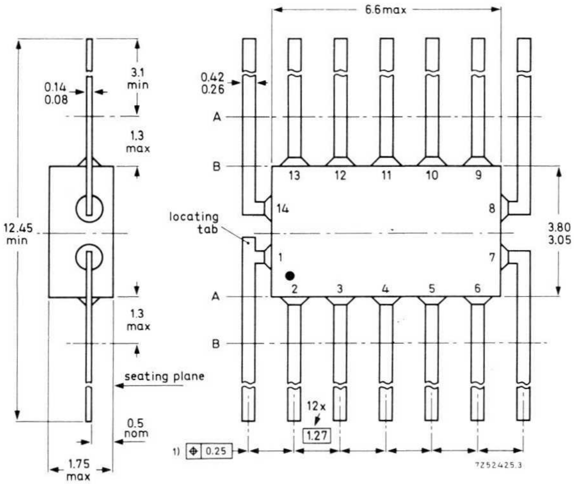


Fig. 7–4. Dimensions of the standard 14-pin flat-pack package type E2 (in mm).

by spot welding each lead to the proper board termination, or placed in a specially formed case which can then be fixed on to the PWB in various ways depending on the construction. The first method is the preferred one, requiring however a special welding tool for the job.

7.2 Printed Wiring Boards, Connectors and Chassis

The PWB complete with integrated circuits and any other components must now be electrically connected to the rest of the system via a connector and physically held in place by a steel chassis. The three parts are described on next pages.

7.2.1 PRINTED WIRING BOARDS

Printed wiring boards are made for special and general purposes and in many different sizes. If the boards are to fit a standard chassis however, they must have well-defined dimensions. The two general purpose ("Experimenter") PWBs having dimensions suitable for their use in our standard Miniature Mounting Chassis have the catalogue number 4322 026 39880 (for mounting fifteen XG14 packages) and 4322 026 39890* (for packages with various numbers of pins). The connections between packages and "to-the-board" connectors are to be made with insulated wires. The packages are mounted perpendicular to the connector so that the wires to the connector can run parallel to the rows of pins of the packages instead of between the pins. The boards are shown in Figs 7–5*a* and *b*.

Note that the boards can also be delivered with connector and extractor in which case the catalogue numbers are ... 38760 (for ... 39880) and ... 38770 (for ... 39890).

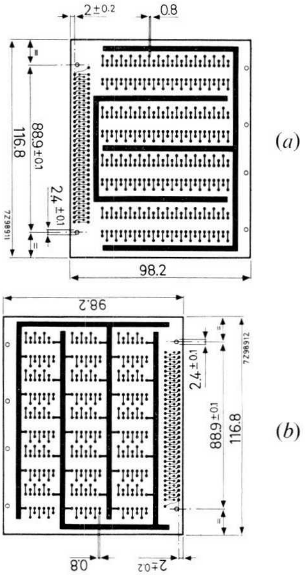


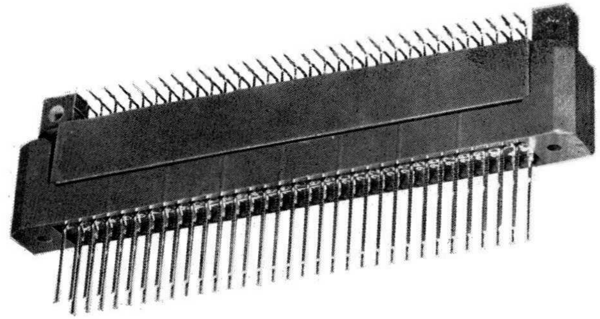
Fig. 7–5. Miniature printed wiring board dimensions, in mm; (a) Type 4322 026 39880; (b) type 4322 026 39890.

[*] Specially designed for the FC family, having printed supply terminals.

7.2.2 CONNECTORS

If special purpose boards are used, the connectors will have to be ordered separately. The two versions of the Miniature Mounting Chassis (see section 7.2.3) will accept two types of male connectors. The two complete (male plus female) connectors have the catalogue numbers 2422 026 89082 (normal mini wire-wrap terminals) and 2422 026 89083 (back panel mini wire-wrap). See Fig. 7–6.

These connectors have two rows of contacts. Each row has 32 contacts. Both connectors may be dip-soldered. They are made of green glass-fibre filled diallyl phthalate and the contacts are of phosphor-bronze plated with 4 μ rolled gold.



Fig, 7–6. Typical connector for use in miniature mounting chassis. Illustrated is the type F054.

7.2.3 CHASSIS (Fig. 7–7)

The standard chassis is the Miniature Mounting Chassis which is designed to fit the 19″ rack system. There are two versions: cat. no. 4322 026 38250 (normal) and 4322 026 38280 (back panel wiring).

Each chassis has place for 41 PWBs. As each board can accomodate up to 36 ICs, each chassis can contain 1476 Integrated Circuits. The PWB must have the dimensions shown in Fig. 7–8.

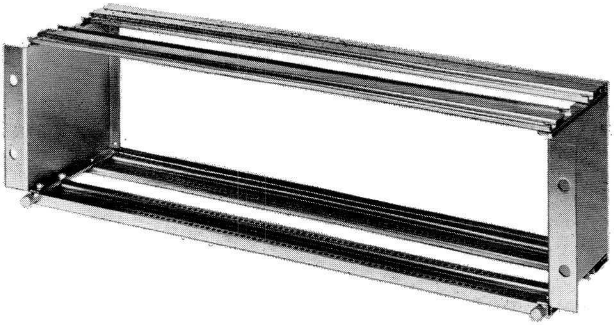


Fig. 7–7. The miniature mounting chassis, cat.no. 4322 026 38280.

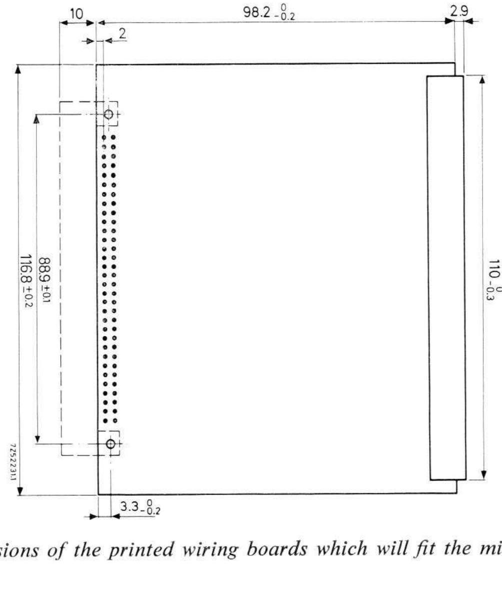


Fig. 7–8. Dimensions of the printed wiring boards which will fit the miniature mounting chassis (in mm).

7.3 Power Supply

Power supply requirements for the FC family of integrated circuits are not difficult to fulfil. The requirements can be stated simply:

Voltage, 6 V ± 5%

Current, 4 A per mounting chassis (approx.).

Permissible ambient temperature, between 0 °C and +75 °C.

No special power supply unit has been designed for the FC family as it has been found that a user's existing power supply is often quite sufficient or that he prefers to make his own unit. General purpose power supply units are available, however, if this is not the case. The four regulated power supply units listed below give an adjustable output voltage of 4.5 V to 15 V, have an operating temperature range of from —10 °C to +65 °C, and have overload protection. The output current available and catalogue number for each type are listed below.

max. output current	cat.no.
1 A	9415 012 00001
3 A	9415 012 01001
5 A	9415 012 02001
10 A	9415 012 03001

Explanation of drawing symbols and notation

Various symbols and notation styles are in use to describe logic functions. In this book we have followed the American Forces standard MIL-STD-806B as far as gate symbols are concerned.

Before proceeding with the gate symbols, however, the notation used should be clearly understood. The following passage is quoted freely from Appendix B of MIL-STD-806B:

"A single arrangement of hardware may perform either the AND function or the OR function. This functional duality is employed in numerous single device and multi-device systems.

"We shall consider the AND function as an element whose output is "active" when all its inputs are "active". Any "non-active" AND input will produce a "non-active" output. The OR function is considered as an element whose output is "active" when one or more inputs are "active". All OR inputs "non-active" will produce a "non-active" output. Note that these definitions do not yet refer to logical "1" or "0", or any electrical reference states such as "HIGH", "LOW", or voltages.

"Graphic representations as well as English notations are employed to illustrate the relationship of specific functions, operating either alone or as parts of a system.

The state condition of active input(s) and the resultant active output are identified by the presence or absence of active state signal indicators (small circles) at the input(s) or output of the logic functions (AND/OR). A small circle at the input(s) indicates that the relatively low (L) input signal activates the function. Conversely, the absence of a small circle indicates that the relatively HIGH (H) input signal activates the function. A small circle at the output indicates that the output of the activated function is relatively LOW (L). Absence of a circle at the output indicates that the output of the activated function is relatively HIGH (H)".

Elsewhere in MIL-STD-806B the terms HIGH and LOW (symbols H and L) are defined as being the *more* positive and *less* positive voltages respectively. The last paragraph in the MIL-STD-806B excerpt thus defines the way in which the activating voltage level at any point in a system can be immediately identified — it is the less positive voltage level if a small circle is present, the more positive level, if no circle is drawn. In the FC family of integrated circuits, H is + 5 V and L is 0 V (approximately).

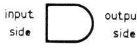
Confusion sometimes arises when the symbols "1" and "0" are used in logic expressions. Although we rarely use these symbols in this publication (being more a practical than a theoretical work) the symbols can be briefly explained now. In essence, "0" is the non-active, and "1" the active state. As there are no further conditions than these, this means that a single Truth Table (made up of "1s" and "0s") can be written for a function, no matter what practical form that function has (i.e. no matter whether circles are present on inputs and outputs or not).

I GATE SYMBOLS

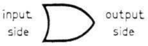
The symbols have been taken from MIL-STD-806B published 26-2-1962 and these, together with the explanation given in the MIL-Spec. are printed in normal face type. The numbers in brackets refers to the section of the MIL-Spec. from which the extract has been made.
Additional information which we consider will help the reader to understand the symbols is given in smaller print.

1 Logic symbols (5 partial)

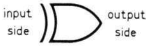
AND The symbol shown below represents the AND function (5.1).



OR The symbol shown below represents the OR function (5.2).



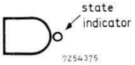
EXCLUSIVE-OR The symbol shown below represents the Exclusive-OR function (5.6).



STATE INDICATOR* (Active) (5.3). The presence of the small circle symbol at the input(s) or output(s) of a function indicates:

(a) Input Condition. The electrical condition at the input terminal(s) which control the active state of the respective function.

(b) Output Condition. The electrical condition existing at the output terminal(s) of an activated function.



*) See p. 163 for drawing dimensions.

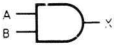
(5.3.1) A small circle(s) at the input(s) to any element (logical or non-logical) indicates that the relatively LOW (L) input signal activates the function. Conversely, the absence of a small circle indicates that the relatively HIGH (H) input signal activates the function.

(5.3.2) A small circle at the symbol output indicates that the output terminal of the activated function is relatively LOW (L), the absence indicates that the output terminal is relatively HIGH (H).

This small circle shall never be drawn by itself on a diagram.

EXAMPLES

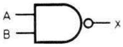
AND (5.1.1)



A	B	X
L	L	L
H	L	L
L	H	L
H	H	H

The output is HIGH if and only if all inputs are HIGH.

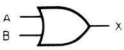
(5.4)



A	B	X
L	L	H
H	L	H
L	H	H
H	H	L

The output is LOW if and only if all inputs are HIGH.

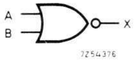
OR (5.2.1)



A	B	X
L	L	L
H	L	H
L	H	H
H	H	H

The output is HIGH if and only if any one or more of the inputs are HIGH.

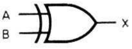
(5.5)



A	B	X
L	L	H
H	L	L
L	H	L
H	H	L

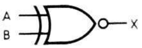
The output is LOW if and only if any one or more of the inputs are HIGH.

EXCLUSIVE-OR (5.6.1)



A	B	X
L	L	L
H	L	H
L	H	H
H	H	L

The output is HIGH if and only if any one input is HIGH and all other inputs are LOW.



A	B	X
L	L	H
H	L	L
L	H	L
H	H	H

The output is LOW if and only if any one input is HIGH and all other inputs are LOW.

Table I, an extract from MIL-STD-806 B, shows two-input AND and OR gate symbols with all the possible combinations of terminals with or without state indicator. It will be noted that the AND-gate symbol in the 1st column has the same function table as the OR-gate symbol in the 2nd column.

TABLE I (continued overleaf)

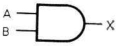
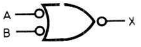
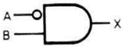
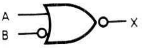
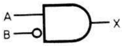
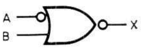
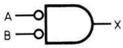
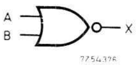
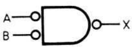
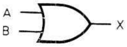
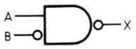
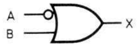
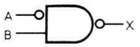
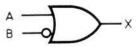
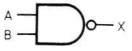
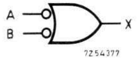
AND symbols	OR symbols	function table		
		A	B	X
		H H L L	H L H L	H L L L
		H H L L	H L H L	L L H L
		H H L L	H L H L	L H L L
	 Note 2	H H L L	H L H L	L L L H
		H H L L	H L H L	H H H L

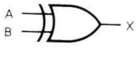
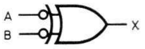
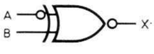
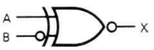
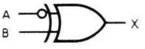
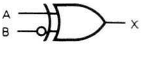
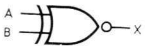
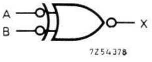
TABLE I (continued)

AND symbols	OR symbols	function table		
		A	B	C
		H	H	H
		H	L	L
		L	H	H
		L	L	H
		H	H	H
		H	L	H
		L	H	L
		L	L	H
		H	H	L
		H	L	H
		L	H	H
		L	L	H

Note 1. In literature often described as NAND
Note 2. In literature often described as NOR
Although the MIL-STD-806B does not use the expressions NAND and NOR they are referred to because these terms are commonly used.

Although MIL-STD-806B shows the EXCLUSIVE-OR symbol only without state indicator, for the sake of completeness Table II shows it with all the possible combinations of terminals with or without state indicator. It will be noted from the table that one function table (3rd column) is applicable to four symbols.

TABLE II

symbol	symbol	function table		
 	 	A	B	X
		H	H	L
		H	L	H
		L	H	H
		L	L	L
 	 Note 3 	A	B	X
		H	H	H
		H	L	L
		L	H	L
		L	L	H

Note 3. In literature described as BINARY-COMPARATOR

HIGH and LOW

The terms relatively HIGH and LOW are explained with reference to the following three examples.

+5.0 volts = HIGH		
+0.5 volts = LOW	+0.5 volts = HIGH	
	—5.0 volts = LOW	—5.0 volts = HIGH
		— 10 volts = LOW

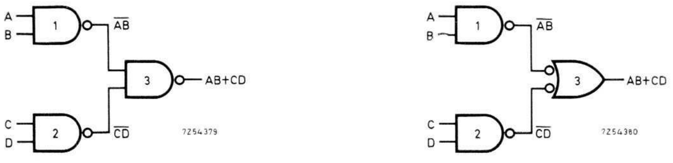
From these it can be seen that the more positive voltage is termed relatively HIGH and the less positive is termed relatively LOW. These terms are abbreviated to HIGH (H) and LOW (L) respectively and are used throughout MIL-STD-806B.

2 Drawing practice

Any of the symbols from Tables I and II may be used in diagrams, bearing in mind the following general rule.
Every signal line shall *preferably* have at each end

either a state indicator circle,
or no state indicator circle.

An example showing how compliance with this rule can be achieved is given in the following sketch.



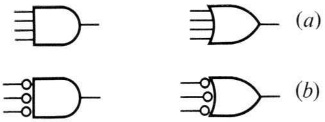
(*a*) (*b*)

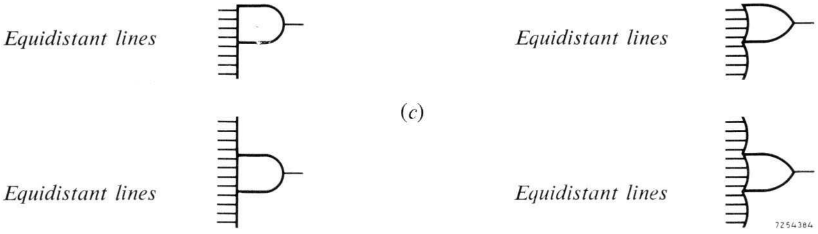
A further advantage of drawing symbol 3 as in (*b*) is that it is more apparent that 3 behaves as an OR function than when 3 is drawn as in (*a*) (cf. table I).

When state indicators are *not* used there should not be drawn more than *four* input lines to the input side of the fundamental symbol (see drawings (*a*) below).

When state indicators *are* used there should not be drawn more than *three* input lines to the input side of the fundamental symbol (see drawings (*b*) below).

These rules should be followed to avoid drawing difficulties caused by lines tending to cross over each other. When more input lines are needed the input side of the symbol can be extended in any of the ways indicated in Figs (*c*) below.

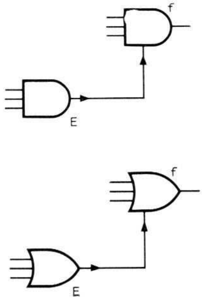




EXPANDED INPUTS

If the number of input lines to a fundamental symbol must be expanded with the aid of an expander circuit, the expander line(s) enter(s) the fundamental symbol as drawn below.

The expander symbol shall be drawn to the same dimensions as the fundamental symbol; however, two filled arrows shall be drawn on each connection line, one arrow close to the expander symbol and one arrow close to the fundamental symbol.



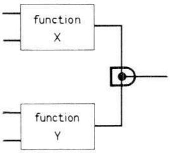
f = "fundamental" symbol

E = Expander symbol

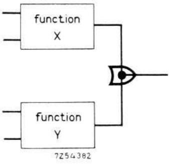
A signal line provided with arrows does not generally imply the usual logic levels. Therefore such a line should not be connected to "normal" inputs of gates but only to one expander input of one gate.

3 Output combinations (6.3)

Where functions have the capability of being combined according to the AND (or OR) function, simply by having the outputs connected, that capability shall be shown by enveloping the branched connection with a smaller sized AND or OR symbol*.



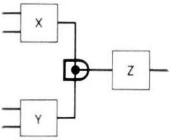
Dot" AND".



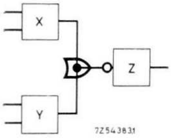
Dot" OR".

[*] These connections of outputs are often described in the literature as "WIRED–OR" or "WIRED-OUTPUT".

EXAMPLES



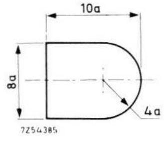
The function Z is activated on its input by a HIGH level (because a state indicator is *not* applied there) if and only if both outputs of functions *X* and *Y* are HIGH.
The branched connection shall therefore be enveloped by a small-sized AND symbol.



The function Z is activated on its input by a LOW level (because a state indicator is applied there) if and only if one or both outputs of the functions X and Y are LOW.
The branched connection shall therefore be enveloped by a small-sized OR symbol.
It should be noted that it would seem necessary to use state indicators on all terminals of the Dot "OR" symbol for correct interpretation of the circuit. However it is not usual to use state indicators on Dot symbols.

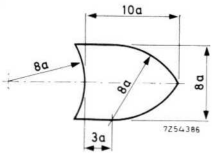
4 Drawing dimensions

Ratio of dimensions of symbols may be derived from the drawings below.

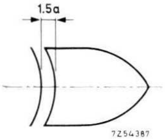


AND

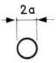
Symbols, enveloping a branched connection shall have half the dimensions of the fundamental symbols.



OR



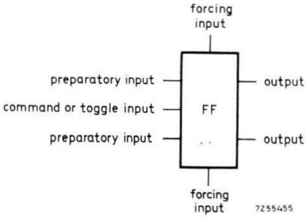
EXCLUSIVE-OR



STATE INDICATOR

II FLIP-FLOP SYMBOLS

1 General symbol



2 Definitions

Active or "1" state of an input signal

That state (either a level or a transition from one level to the other) which causes, directly or indirectly, a change of the output state. Conversely, the inactive or "0" state of an input signal is that state which does not cause an output change.

Output state

There may be one output terminal (Q) or two (Q_1 and Q_2). If there are two, the "output state" refers to the states of the signals at Q_1 and Q_2; since these are normally complementary, the state at Q_1 is usually considered to represent the output state.

Preparatory input terminal (e.g. J, K, D)

An input terminal to which application of an active signal does not directly cause a change of the output state but prepares the circuit for such a change.

Command input terminal (T)

An input terminal to which application of an active signal causes the output to assume the state corresponding to the preparatory inputs. It is also known as the "clock input terminal".

Toggle input terminal (T)

An input terminal at which an active transition from one level to the other directly causes a change of the output state.

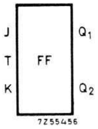
Forcing input terminal (S_1 = *"direct set"*, S_2 = *"direct reset"*)

An input terminal at which application of an active signal directly causes the output to assume a specific state, irrespective of the states of other input terminals.

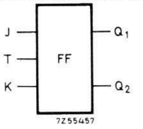
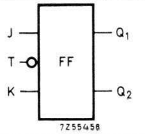
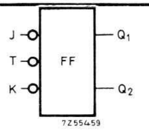
3 Location of terminals and use of polarity state indicator

Legend:
H = HIGH level
L = LOW level
L→H = transition from LOW level to HIGH level
H→L = transition from HIGH level to LOW level
X = state (level or transition) has no influence
? = indeterminate, unless exact timing of relevant input signals (e.g. S_1 and S_2) is known.

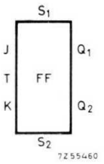
3.1 *J-K flip-flop without forcing inputs*



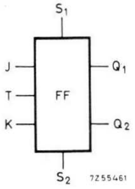
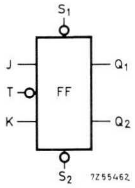
An active ("1") signal at J, together with an inactive (0") signal at K and an active signal transition at T, causes the "1" state at Q_1 and the "0" state at Q_2.

symbol	function table				
	J	K	T	Q_1	Q_2
	H	L	L→H	H	L
	L	H	L→H	L	H
	H	H	L→H	reversed	
	L	L	L→H	no change	
	X	X	H→L	no change	
	H	L	H→L	H	L
	L	H	H→L	L	H
	H	H	H→L	reversed	
	L	L	H→L	no change	
	X	X	L→H	no change	
	L	H	H→L	H	L
	H	L	H→L	L	H
	L	L	H→L	reversed	
	H	H	H→L	no change	
	X	X	L→H	no change	

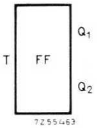
3.2 *J-K flip-flop with forcing inputs*



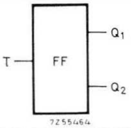
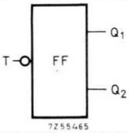
Irrespective of the states at *J*, *K* and *T*: an active ("1") signal at S_1, together with an inactive ("0") signal at S_2, causes Q_1 to assume the "1" and Q_2 the "0" state.

symbol	function table						
	J	K	T	S_1	S_2	Q_1	Q_2
	X	X	X	H	L	H	L
	X	X	X	L	H	L	H
	X	X	X	H	H	?	?
	H	L	L→H	L	L	H	L
	L	H	L→H	L	L	L	H
	H	H	L→H	L	L	reversed	
	L	L	L→H	L	L	no change	
	X	X	H→L	L	L	no change	
	X	X	X	L	H	H	L
	X	X	X	H	L	L	H
	X	X	X	L	L	?	?
	H	L	H→L	H	H	H	L
	L	H	H→L	H	H	L	H
	H	H	H→L	H	H	reversed	
	L	L	H→L	H	H	no change	
	X	X	L→H	H	H	no change	

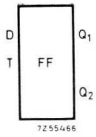
3.3 "*Toggle*" *flip-flop*



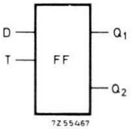
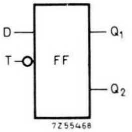
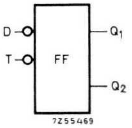
An active ("1") signal transition at *T* causes the complementary states at Q_1 and Q_2 to reverse.

symbol	function table		
	T	Q_1	Q_2
	L→H H→L	reversed no change	
	H→L L→H	reversed no change	

3.4 *Edge-triggered D flip-flop*

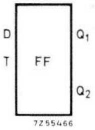


An active ("1") signal transition at T causes Q_1 to assume the same state as D. i.e., if D is in the "1" state during the active transition at T, Q_1 also assumes the "1" state: if D is "0", Q_1 also becomes "0". The output state will remain unchanged until the next active transition at T occurs.

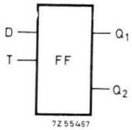
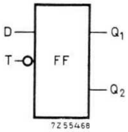
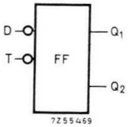
symbol	function table				
	D	T		Q_1	Q_2
		level	transition		
	H		L→H	H	L
	L		L→H	L	H
	X	X		no change	
	X		H→L	no change	
	H		H→L	H	L
	L		H→L	L	H
	X	X		no change	
	X		L→H	no change	
	L		H→L	H	L
	H		H→L	L	H
	X	X		no change	
	X		L→H	no change	

3.5 *Level-operated ("gated") D flip-flop, or "Bistable latch"*

(Graphical symbol equal to 3.4.)

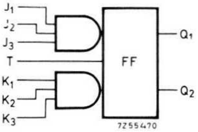


As long as the signal at T is at its active ("1") level, the signal at Q_1 follows the signal at D. When the signal at T changes to its inactive ("0") level, the signal at Q_1 latches (subsequent changes in D cause no change in Q_1). Q_1 unlatches when the signal at T returns to its active level.

symbol	function table				
	D	T		Q_1	Q_2
		level	subsequent transition		
	H	H		H	L
	H	H	H→L	H	L
	L	H		L	H
	L	H	H→L	L	H
	X	L		no change	
	H	L		H	L
	H	L	L→H	H	L
	L	L		L	H
	L	L	L→H	L	H
	X	H		no change	
	L	L		H	L
	L	L	L→H	H	L
	H	L		L	H
	H	L	L→H	L	H
	X	H		no change	

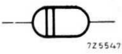
4 Multiple inputs

Where inputs are functionally combined by an input gate, the connecting line between the gate symbol and the flip-flop symbol may conveniently be omitted, as shown in the drawing.

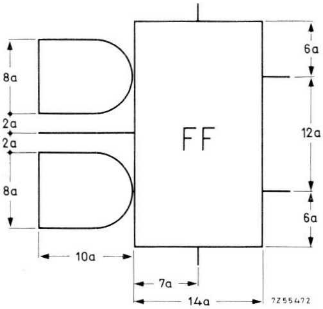


5 Time delay circuit

The following time delay symbol (MIL-STD-806 B, 5.15) is used in some logic flip-flop block diagrams:



6 **Drawing dimensions** (to MIL-STD-806 B)



The ratio of dimensions is given in the drawing above.

Technology relating to the products described in this publication is shared by the following firms.

Argentina
FAPESA I.y.C.
Melincué 2594
Tel. 50-9941/8155
BUENOS AIRES

Australia
Philips Electrical Pty. Ltd.
Miniwatt Electronics Division
20, Herbert St.
Tel. 43-2171
ARTARMON, N.S.W.

Austria
WIVEG
Prinz Eugenstrasse 32
Tel. 65 16 21
1041 WIEN

Belgium
PHILIPS S. A.
Place de Brouckère 2
Tél. 19.18.00 - 19.30.00
1000 BRUXELLES

Brazil
IBRAPE S.A.
Rua Manoel Ramos Paiva 506
Tel. 93-5141
SAO PAULO

Canada
Philips Electron Devices
116 Vanderhoof Ave.
Tel. 425-5161
TORONTO 17, Ontario

Chile
Philips Chilena S.A.
Av. Santa Maria 0760
Tel. 39 40 01
SANTIAGO

Columbia
SADAPE S.A.
Calle 19, No. 5-51
Tel. 422-175
BOGOTA D.E. 1

Denmark
Miniwatt A/S
Emdrupvej 115
Tel. 69 16 22
KØBENHAVN NV

Finland
Oy Philips A.B.
Elcoma Division
Kaivokatu 8
Tel. 10 915
HELSINKI 10

France
R.T.C.
La Radiotechnique-Compelec
Avenue Ledru Rollin 130
Tel. 797-99-30
PARIS 11

Germany
VALVO G.m.b.H.
Valvo Haus
Burchardstrasse 19
Tel. (0411) 33 91 31
2 HAMBURG 1

Greece
Philips S.A. Hellénique
Service Division
54, Av. Syngrou
ATHENES

Hongkong
I.D.C.C. ELCOMA
c/o Philips Hongkong Ltd.
Tel. 24 10 18
HONGKONG

India
INBELEC Div. of
Philips India Ltd.
Band Box Building
254-D, Dr. Annie Besant Road
Tel. 45 33 86, 45 64 20, 45 29 86
Worli, BOMBAY 18 (WB)

Indonesia
P.T. Philips-Ralin Electronics
ELCOMA Division
Djalan Gadjah 18
Tel. 44 163
DJAKARTA

Ireland
Philips Electrical (Ireland) Pty.
Newstead, Clonskeagh
Tel. 69 33 55
DUBLIN 6

Italy
Philips S.p.A.
Sezione Elcoma
Piazza IV Novembre 3
Tel. 69.94
MILANO

Japan
I.D.C.C. Ltd.
Kokusai Building, 7th floor
Marunouchi
Tel. (213) 6751.7
TOKYO

Mexico
Electrónica, S.A. de C.V.
Varsovia No. 36
Tel. 5-33-11-80
MEXICO 6, D.F.

Netherlands
Philips Nederland N.V.
Afd. Elonco
Boschdijk, VB
Tel. (040) 43 33 33
EINDHOVEN

New Zealand
EDAC Ltd.
70-72 Kingsford Smith Street
Tel. 873 159
WELLINGTON

Norway
Electronica A/S
Middelthunsgate 27
Tel. 46 39 70
OSLO 3

Peru
CADESA
Av. Abancay 1176
Offs. 606-607
Tel. 7 73 17
LIMA

Portugal
Philips Portuguesa S.A.R.L.
Rua Joaquim Antonio de Aguiar 66
Tel. 68 31 21/9
LISBOA

South Africa
EDAC (Pty) Ltd.
South Park Lane
New Doornfontein
Tel. 24/6701-2
JOHANNESBURG

Spain
COPRESA S.A.
Balmes 22
Tel. 2 32 03 00
BARCELONA 7

Sweden
ELCOMA A.B.
Lidingövägen 50
Tel. 08/67 97 80
10250 STOCKHOLM 27

Switzerland
Philips A.G.
Edenstrasse 20,
Tel. 051/44 22 11
CH-8027 ZUERICH

Turkey
Türk Philips Ticaret A.S.
EMET Department
Gümüssuyu Cad. 78-80
Tel. 45.32.50
Beyoglü, ISTANBUL

United Kingdom
Mullard Ltd.
Mullard House
Torrington Place
Tel. 01-580 6633
LONDON W.C. 1

United States
Amperex Electronic Corp.

Electron Tubes Div.
Tel. 516 WE 1-6200
HICKSVILLE N.Y

Sem. and Microcircuits Div.
Tel. 401-762-9000
SLATERSVILLE R.I. 02876

Electronic Components Div.
Tel. 516-234-7000
HAUPPAGE N.Y.

Uruguay
Luzilectron S.A.
Rondeau 1576, piso 5
Tel. 9 43 21
MONTEVIDEO

Venezuela
C.A. Philips Venezolana
Elcoma Department
Colinas de Bello Monte
Tel. 72.01.51
CARACAS

Printed in The Netherlands ③⑤⑥⑦ 9399 303 05601