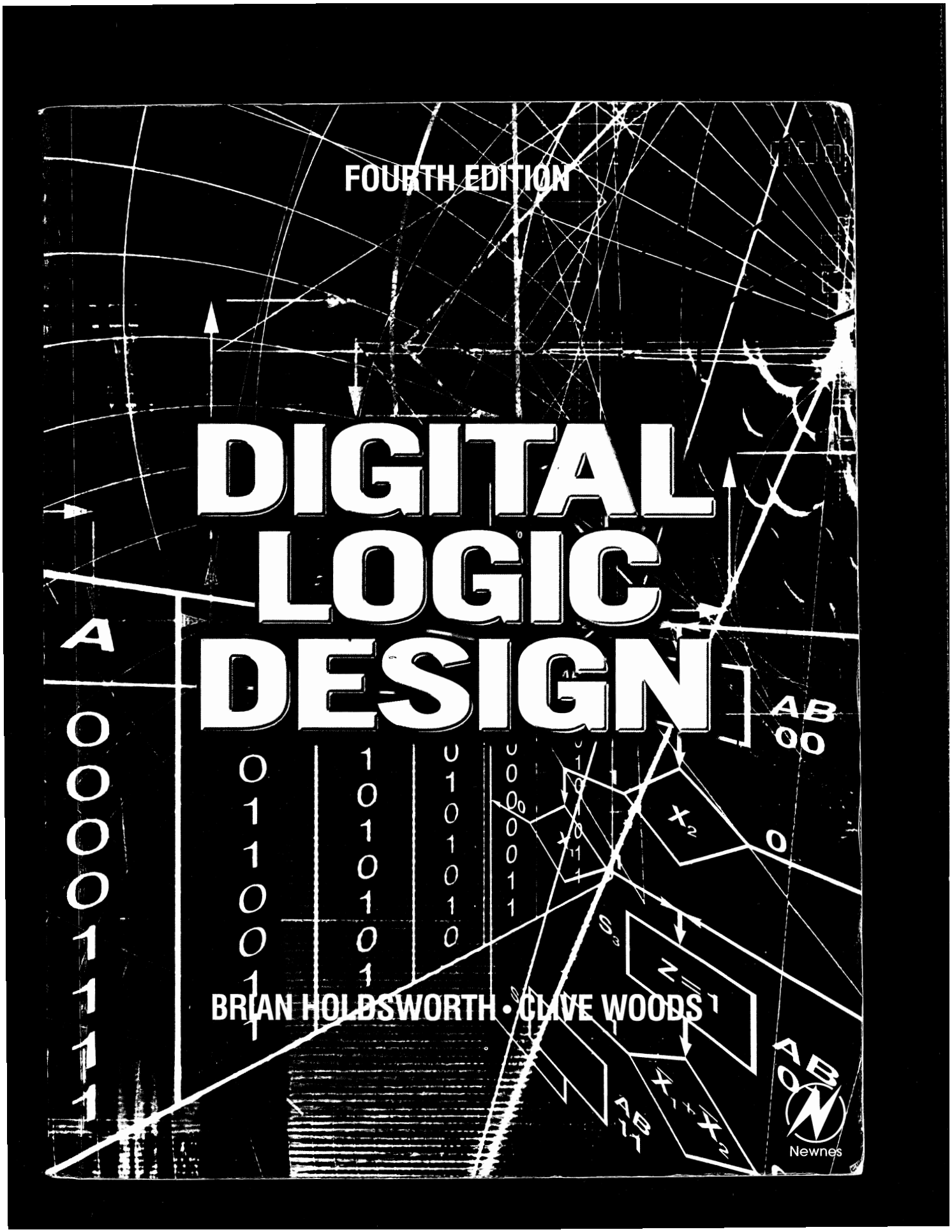


Ex.1007
CISCO SYSTEMS, INC. / Page 1 of 7

Newnes
An imprint of Elsevier Science
Linacre House, Jordan Hill, Oxford OX2 8DP
225 Wildwood Avenue, Woburn, MA 01801-2041

First published 1982
Second edition 1987
Third edition 1993
Fourth edition 2002

Copyright © Brain Holdsworth and Clive Woods, 1982, 1987, 1993, 2002

The right of Brian Holdsworth and Clive Woods to be identified as the authors of this work has been asserted in accordance with the Copyright, Designs and Patents Act 1988

No part of this publication may be reproduced in any material form (including photocopying or storing in any medium by electronic means and whether or not transiently or incidentally to some other use of this publication) without the written permission of the copyright holder except in accordance with the provisions of the Copyright, Designs and Patents Act 1988 or under the terms of a licence issued by the Copyright Licensing Agency Ltd, 90 Tottenham Court Road, London, England W1T 4LP. Applications for the copyright holder's written permission to reproduce any part of this publication should be addressed to the publishers

British Library Cataloguing in Publication Data
A catalogue record for this book is available from the British Library

ISBN 0 7506 45882

Typeset by Integra Software Services Pvt. Ltd, Pondicherry, India
www.integra-india.com

Printed and bound in Great Britain

Preface to the fourth edition

In this newly revised edition of *Digital Logic Design*, we have taken the opportunity to undertake extensive revisions of much material contained in the third edition, whilst retaining its comprehensive coverage of the subject. Like the previous editions, the current edition is intended to cover all the material that is needed in a typical undergraduate or Master's course on Digital Logic Systems, and also to act as a reference text for graduates working in this field. To this end, we have retained all elementary material assuming little or no background, but the advanced chapters have accordingly been revised to take account of recent trends in hardware availability. A number of additional problems have been set at the end of some of the chapters, sometimes without answers, in order to allow the reader to exercise his/her design capabilities without the luxury of being able to refer to worked solutions.

The chapter on instrumentation and interfacing is almost entirely new, and the chapters on programmable logic devices, and on fault diagnosis and testing, have been considerably enlarged as a result, on the one hand, of significant advances in the technology and the range of devices now available to the designer, and on the other hand to emphasise that logical fault-finding methods, far from being esoteric, impossible to apply in practice, trivial, or demeaning for a professional engineer to use, are actually worthy of serious study and application.

> Material enclosed in boxes in this manner is usually not needed later in this text, and is not as important as the main narrative, or sometimes summarises work in the main text. This material may be rather more demanding than the main text, or be unusual or obscure in some other manner; generally speaking, proofs of results in these sections and subsections are not given in detail, and are left as more of a challenge for the interested reader to work out in full. The first-time reader, or a reader not aiming for complete coverage of all the material in this text, may safely ignore these sections and subsections.

Throughout the main part of this edition, we have used the 'old' IEEE logic symbols rather than the 'new' BS3939 symbols; this is a result of a perceived shift in attitudes in the engineering profession, and the IEEE symbols are now recommended alongside the BS symbols. Modern CAD systems are capable of printing the 'old' symbols with ease, eliminating the major initial advantage of the 'new' symbols when first introduced. However, as an understanding of the 'new' symbols is also a useful accomplishment, a summary of the 'new' system is included as an Appendix.

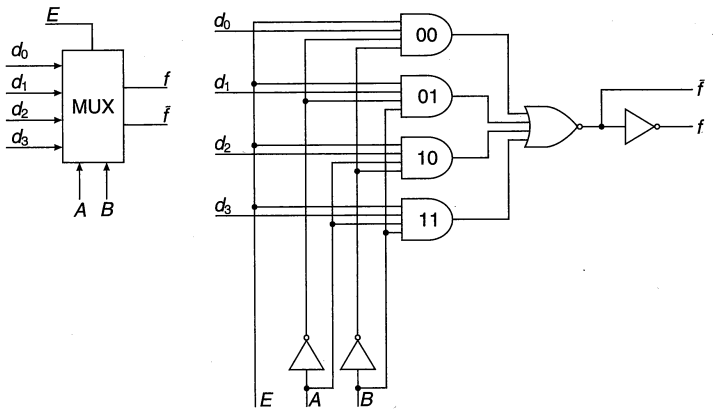


Figure 5.1 *(a) Block diagram of a 4-input multiplexer and (b) its gate implementation*

5.3 Available MSI multiplexers

The sizes of multiplexer available in the TTL family are limited by pin availability on standard MSI chips. With 16-pin chips, multiplexers having 2, 4, and 8 data lines are available and with 24-pin chips it is possible to provide a multiplexer with 16 data lines.

The 74157, a 16-pin chip, provides quadruple 2-to-1 multiplexers, where each multiplexer consists of a 2-wide, 3-input AO gate, with one input for data, one for selection, and the third which is the strobe or enable line. A logic diagram, the truth table and a traditional block diagram are shown in Figure 5.2. The numbers in parentheses on the input and output lines are the pin numbers.

The 16-pin 74353 is a dual 4-to-1 data selector/multiplexer. Each multiplexer consists of a 4-input 4-wide AOI gate with tri-state control (see Chapter 10) on the NOR output gates. The output control lines also act as separate enable inputs for the two devices, both of which are controlled by the common select lines. A logic diagram for the 74353, its function table and block diagram are shown in Figure 5.3.

At the upper end of the scale, the 74251 is an 8-to-1 multiplexer having complementary tri-state outputs. It consists of a 4-input 8-wide AOI gate with an enable strobe. There is also the 24-pin 74150, a 16-to-1 multiplexer/data selector which consists of a 5-input 16-wide AOI gate. The number of pins available on this chip limits the device to a single-output line.

5.4 Interconnecting multiplexers

Data within a digital system is normally processed in parallel form in order to increase the speed of operation. If the output of the system has to be transmitted over a relatively long distance then a parallel-to-serial conversion will take place so that the data can be transmitted serially over a single transmission line. This eliminates the requirement of individual transmission lines, one for each bit. The arrangement of Figure 5.4 can be regarded as an example of parallel-to-serial conversion. An 8-bit

multiplexing. It consists of first finding the residue functions at the inputs to the output and second level multiplexers and finally determining the input residue functions at the first level from the second level listings. Implementation of the function using three levels of multiplexing is shown in Figure 5.9(e). The residue functions at each level of multiplexing are marked on the diagram and it will be observed that M2, M3, M4 and M5 are redundant. The implementation requires a single quad 2-to-1 MUX and an inverter if the complement of variable A is not available. The number of pin connections used for this implementation is ten, but only one multiplexer package is needed. In general the possibility of redundancy in a multi-variable function is highest when the smallest multiplexer elements are used.

5.7 Demultiplexers

As the name infers, a demultiplexer performs the opposite function to that of a multiplexer. A single data line can be connected to any one of the output lines provided by the choice of an appropriate select signal. If there are s select inputs then the number of output lines to which the data can be routed is $n = 2^s$. The structure of a demultiplexer is identical to that of a decoder. A basic 2-to-4 line decoder and its associated enable line is shown in Figure 5.10. If the enable line is now used as a data input the data can be routed to any one of the outputs. If, for example, $A = B = 0$ and the data input $= 1$ then output $Y0 = 1$ and the remaining three outputs are 0. Any decoder having an enable line can function as a demultiplexer and for this reason they are listed as decoder/demultiplexers in manufacturers' catalogues.

A typical example of a 3-to-8 line decoder/demultiplexer, the 74138, is shown in Figure 5.11. The 74138 with three enable inputs has a flexible enabling system. If the package is to be used as a demultiplexer then input lines G2A and G2B can be grounded and G1 can be used as the input line for data. When the input G1 $= 1$ the eight output NAND gates are enabled. If the select signal is now $ABC = 000$, the output $Y0 = 0$ while all other results remain at 1.

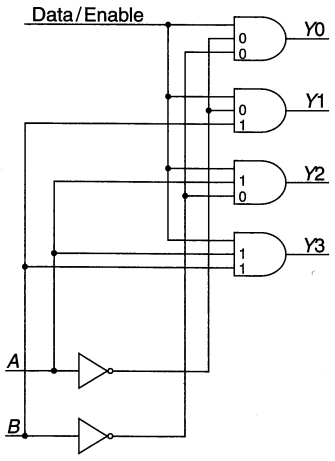
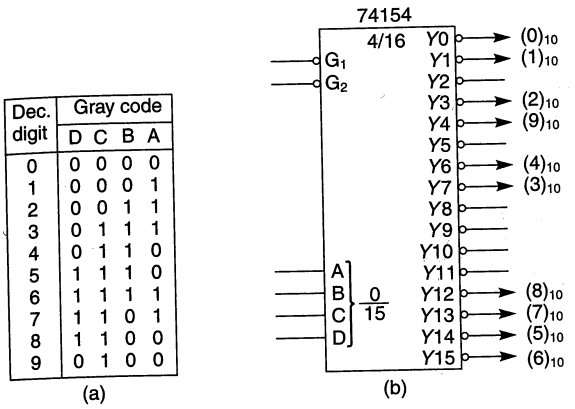


Figure 5.10 *Basic demultiplexer*



Dec. digit	Gray code D	C	B	A
0	0	0	0	0
1	0	0	0	1
2	0	0	1	1
3	0	1	1	1
4	0	1	1	0
5	1	1	1	0
6	1	1	1	1
7	1	1	0	1
8	1	1	0	0
9	0	1	0	0

Figure 5.15 *Gray code to decimal conversion with 74154 4-to-16 line coder (a) Code tabulation (b) Implementation*

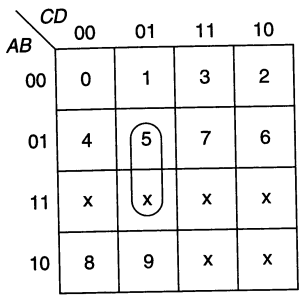


Figure 5.16 *K-map for NBCD/decimal conversion*

There are two types of NBCD/decimal decoders available. First, the decoder that implements the minterm corresponding to decimal digit five, so that $(5)_{10} = \bar{A}B\bar{C}D$. For the second type, the digit $(5)_{10}$ is simplified by combining the minterm $\bar{A}B\bar{C}D$ with the adjacent 'can't happen' term $AB\bar{C}D$ so that $(5)_{10} = B\bar{C}D$, as illustrated on the K-map in Figure 5.16. A typical example of a 4-to-10 line NBCD/decimal decoder in the TTL family is the 7442 which rejects all false data since it is implemented without minimisation. For the decimal decoder that employs simplification techniques the appearance at the input terminals of the minterm $AB\bar{C}D$ will be recognised as $(5)_{10}$ or $(9)_{10}$ depending upon the simplification.

5.10 Decoder networks

When a large decoding network is required it cannot be implemented in a single MSI package because of the large number of pins needed. For example, a 6-to-64 line decoder requires seventy pins for input and output in addition to those for enabling the package and the voltage supply. The decoding range can be extended by interconnecting decoder chips. Two possible schemes are available, (a) *tree decoding* and (b) *coincident* or *2-dimensional decoding*.

A block diagram of a 4-to-16 line tree decoder is shown in Figure 5.17. It consists of the interconnection of five 2-to-4 line decoders. This requires three dual 2-to-4 line chips with four interconnections between the two levels of decoding where the select lines in the second level of decoding are commoned. An extension of the scheme would require extra levels of decoding and a 6-to-64 line decoder is obtained by the addition of one extra level of decoding consisting of a bank of sixteen 2-to-4 line decoders. The alternative approach of coincident decoding is also illustrated in Figure 5.17. Here the 4-to-16 line decoder consists of two 2-to-4 line decoders, available on a single

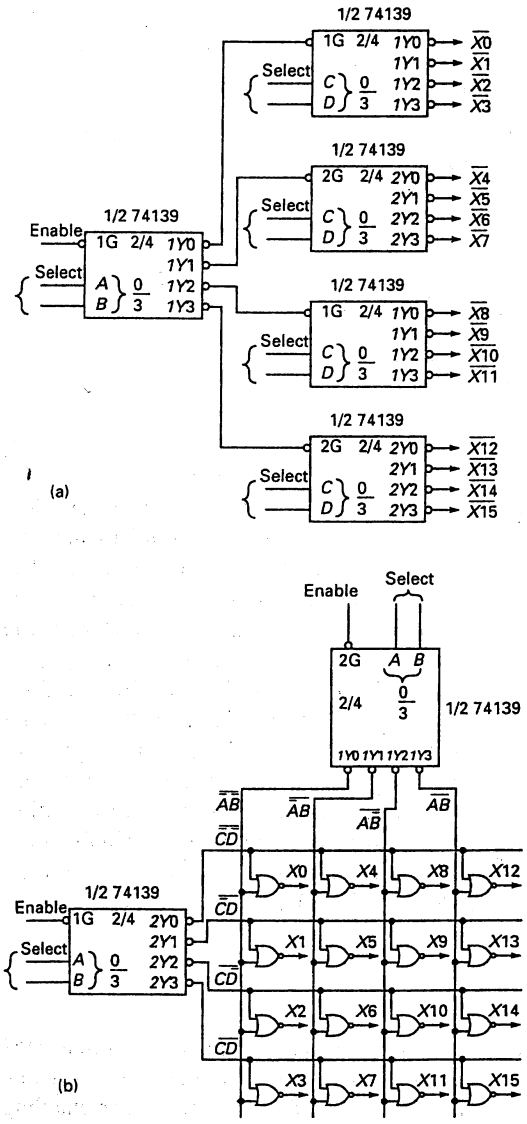


Figure 5.17 *(a) 4-to-16 line tree decoder (b) 4-to-16 line coincident decoder*

chip, and, in addition, sixteen NOR gates. Assuming that four 2-input NOR gates are available on a single chip, a total of five chips are needed for the implementation of this scheme. The coincidence scheme clearly requires more chips and more interconnections than the corresponding tree decoder and as the number of inputs increases, the superiority of the tree decoder becomes more marked.

In spite of this disadvantage, coincident decoders are widely used in conjunction with memory arrays because the NOR gates can be incorporated in the array. The choice

Ex.1007
CISCO SYSTEMS, INC. / Page 7 of 7