IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Patent of:	Jentsung Lin	
U.S. Pat. No.:	7,693,002	Attorney Docket No.:39521-0054IP1
Issue Date:	April 6, 2010	
Appl. Serial No.:	11/548,132	
Filing Date:	October 10, 2006	
Title:	DYNAMIC WORD LINE DRIVERS AND DECODERS FOR MEMORY ARRAYS	

DECLARATION OF EDWARD G. FAETH

I, Edward G. Faeth, of Rockville, MD, declare:

1. My name is Edward G. Faeth. I am over the age of 21, and I have personal knowledge of the facts contained herein unless otherwise indicated. I am an employee at Fish & Richardson P.C.

2. On March 27, 2018, I visited the Jefferson Building of the Library of Congress in Washington, D.C., where I previously requested a copy of the book identified below to compare the electronic copy of the book, submitted as an exhibit with this petition, with the physical copy maintained by the Library of Congress.

3. Using the Library of Congress record for APPLE-1007 ("Itoh") (Appendix A), located through the title and author of APPLE-1007, I requested a copy of CALL NUMBER TK7895.M4 I876 2001 from the library system. I compared the contents of APPLE-1007 to the physical copy provided to me by the Library of Congress and the appearance, contents, substance, and other characteristics of

APPLE-1007 appeared to be the same as the copy maintained by the Library of Congress.

1. On April 10, 2018, I again visited the Jefferson Building of the Library of Congress in Washington, D.C., where I previously requested a copy of the book identified below to compare the electronic copy of the book, submitted as an exhibit with this petition, with the physical copy maintained by the Library of Congress.

2. Using the Library of Congress record for APPLE-1009 ("Brown") (Appendix B), located through the title and author of APPLE-1009, I requested a copy of CALL NUMBER TK7868.L6 B76 2002 from the library system. I compared the contents of APPLE-1009 to the physical copy provided to me by the Library of Congress and the appearance, contents, substance, and other characteristics of APPLE-1009 appeared to be the same as the copy maintained by the Library of Congress.

I declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under 18 U.S.C. § 1001.

Date: April 10, 2018 By: /Edward G. Faeth/
Edward G. Faeth, Rockville, MD

APPENDIX A

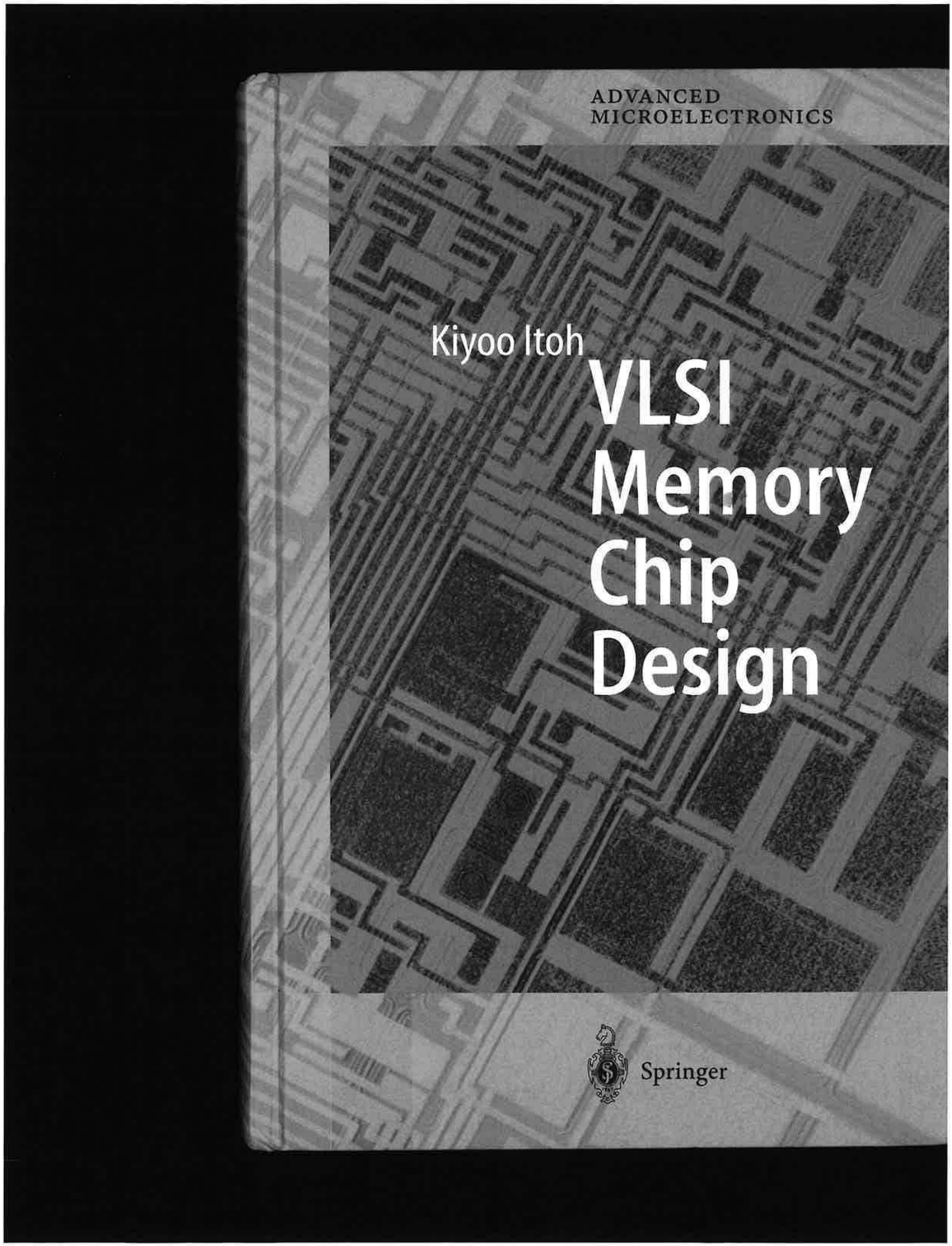
LIBRARY OF CONGRESS
ONLINE CATALOG

1. VLSI memory chip design

LCCN	00068735
Type of material	Book
Personal name	Itoh, Kiyoo, 1941-
Main title	VLSI memory chip design / Kiyoo Itoh.
Published/Created	Berlin ; New York : Springer, 2001.
Description	xi, 495 p. : ill. ; 25 cm.
Links	Publisher description http://www.loc.gov/catdir/enhancements/fy0816/00068735-d.html Table of contents only http://www.loc.gov/catdir/enhancements/fy0816/00068735-t.html
ISBN	3540678204 (alk. paper)
LC classification	TK7895.M4 I876 2001
Subjects	Semiconductor storage devices--Design and construction. Integrated circuits--Very large scale integration--Design and construction.
Notes	Includes bibliographical references (p. [473]-488) and index.
Series	Springer series in advanced microelectronics ; 5
Dewey class no.	621.39/732
CALL NUMBER	TK7895.M4 I876 2001 Copy 1
Request in	Jefferson or Adams Building Reading Rooms

LIBRARY OF CONGRESS ONLINE CATALOG
Library of Congress
101 Independence Ave., SE
Washington, DC 20540

Questions? Ask a Librarian:
https://www.loc.gov/rr/askalib/ask-contactus.html


ADVANCED
MICROELECTRONICS
Kiyoo Itoh
VLSI
Memory
Chip
Design
Springer

RONICS

rovides systematic information on
manufacturing of microelectronic
rchers or engineers in their fields,
:h as wafer processing, materials,
/LSI implementation, and subsys-
ı physics and engineering and the
as research scientists.

ss Communication

ooks/ssam/

Kiyoo Itoh

VLSI Memory Chip Design

With 416 Figures and 26 Tables

Springer

Dr. Kiyoo Itoh
Hitachi Ltd., Central Research Laboratory
1-280, Higashi-Koigakubo
Kokubunji-shi
Tokyo185-8601
Japan
e-mail: k-itoh@crl.hitachi.co.jp

Series Editors:

Dr. Kiyoo Itoh
Hitachi Ltd., Central Research Laboratory
1-280 Higashi-Koigakubo
Kokubunji-shi
Tokyo 185-8601
Japan

Professor Takayasu Sakurai
Center for Collaborative Research
University of Tokyo
7-22-1 Roppongi, Minato-ku,
Tokyo 106-8558
Japan

Library of Congress Cataloging-in-Publication Data

Itoh, Kiyoo, 1941-
VLSI memory chip design / Kiyoo Itoh.
p. cm. -- (Springer series in advanced microelectronics ; 5)
Includes bibliographical references and index.
ISBN 3540678204 (alk. paper)
1. Semiconductor storage devices--Design and construction. 2. Integrated circuits--Very large scale integration--Design and construction. I. Title. II. Series.

TK7895.M4 I876 2001
621.39'732--dc21

00-068735

ISSN 1437-0387
ISBN 3-540-67820-4 Springer-Verlag Berlin Heidelberg New York

Springer-Verlag Berlin Heidelberg New York
a member of Springer Science+Business Media

http://www.springer.de

Printed in Germany

Typesetting by the author.
Data conversion and figure processing by LE-TEX Jelonek, Schmidt & Vöckler GbR, 04229 Leipzig.

Cover concept by eStudio Calmar Steinen using a background picture from Photo Studio "SONO". Courtesy of Mr. Yukio Sono, 3-18-4 Uchi-Kanda, Chiyoda-ku, Tokyo.
Cover design: *design & production* GmbH, Heidelberg

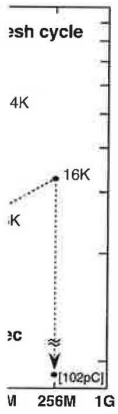
Printed on acid-free paper SPIN: 12187942 57/3180 - 5 4 3

ıance, the basic technologies
: peripheral circuits are des-
cy are explained. Chapter 4
.M which strongly influences
, in the chip. The relation-
ving/sensing is explained in
: on-chip voltage generators
tors are essential for power-
apter 6 discusses subsystem-
mportant in providing wide
S. Chapters 7 and 8 describe
sizing the importance of the
f lowering power-supply vol-
ıold-current reduction which

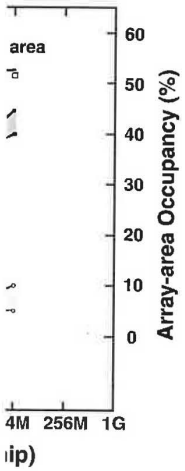
leagues and the office admi-
Ohta, at Hitachi Ltd. They
to finalize my work. Special
inuing support and patience

Kiyoo Itoh

Contents



f data lines [3.3, 3.4]. A C_D of 200 fF



se amplifiers in a DRAM chip [3.4]

y of a word line made of a resistive
r of divisions has been determined
ltant area penalty. In the 64 Kb
short and the speed requirement
e, as shown in Fig. 3.40a [3.9], was
, poly-Si was replaced by a lower-

resistance polycide, in order to double or quadruple the number of memory cells to be driven. To further reduce the RC delay, a poly-Si or polycide word line strapped with a low-resistance aluminum line in each string of 64–128 cells [3.17], as shown in Fig. 3.40b, has been widely accepted in commercial chips since the 1 Mb generation. In the 64 Mb generation, even a hybrid division of the two types shown in Fig. 3.40 has been proposed. However, the aluminum-strapped word-line structure suffers from some drawbacks. It becomes difficult to achieve fine patterning of aluminum at a tight pitch of word lines on a hilly surface, while still connecting to the poly-Si line at the bottom, as the memory cell is miniaturized. In addition, even the word-line structure starts to create quite a large RC delay as many memory cells are connected to a word or subword line. A multidivided word-line scheme using a hierarchical word-line structure [3.18, 3.19], as shown in Fig. 3.25, solves these problems, and is discussed later in detail.

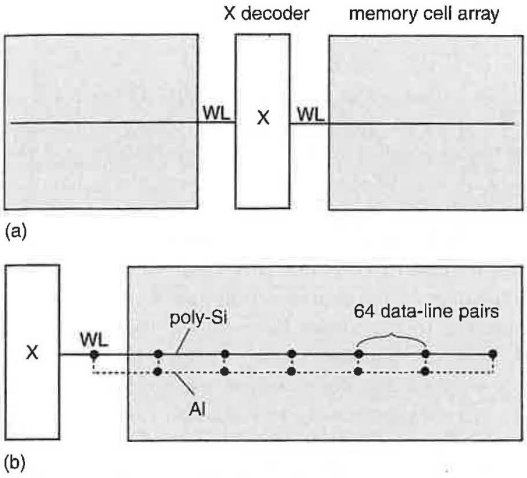


Fig. 3.40. Reductions in word-line delay [3.4, 3.9, 3.17]. (**a**) 64 Kb–256 Kb. 64 Kb: poly-Si (30 Ω/□), 64 data-line pairs per WL. 256 Kb: polycide (1–5 Ω/□), 256 data-line pairs per WL. (**b**) 1 Mb and beyond: poly-Si (50 Ω/□), Al (0.1 Ω/□)

3.6 Read and Relevant Circuits

3.6.1 The Address Buffer

In the NMOS era of the 16–256 Kb generations, a differential address buffer using an on-chip reference voltage [3.14] was widely used. Since the 1 Mb

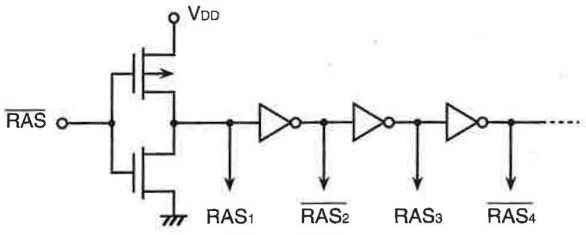


Fig. 3.41. The $\overline{\text{RAS}}$ clock buffer [3.21]

generation, various simple CMOS address buffers have been proposed. In principle, the same circuit configuration is applicable to both the row and column address buffers. Recently, however, a high-speed column address buffer has been especially important to enhance the data throughput by shortening the column address access time t_{AA}, which is dominated by the address buffer (about 40% of t_{AA} [3.20]).

Figure 3.41 shows a typical $\overline{\text{RAS}}$-clock buffer [3.21] to control the row address buffers. To discriminate between the TTL logic levels of over 2.4 V or below 0.8 V for an input $\overline{\text{RAS}}$ signal, the logical threshold voltage is adjusted to be about 1.6 V by tuning the channel-width ratios of the NMOS and the PMOS at the first input stage. A multistage CMOS circuit controls the row internal circuits, using pulses with differing polarities and delays.

Figure 3.42 shows an address buffer that features a cross-coupled differential amplifier [3.21]. It enables an almost constant speed due to a differential circuit configuration, independently of the power-supply noise. Address buffers consisting of inverters, similar to the above $\overline{\text{RAS}}$ buffer, have also been widely accepted. In general, however, the operation of inverter-type buffers is susceptible to power-supply noise [3.20]. For example, as soon as the input logic signals to many buffers are simultaneously switched to the other logic state so that the ground (V_{SS}) line voltage is instantaneously raised and maximized, the speed of each buffer is degraded, with a reduced NMOS gate–source voltage. For a 64 Mb design [3.20], a simultaneous switching of 13 address buffers caused a V_{SS} noise of 0.4 V and a speed difference of 2.3 ns between the inverter and cross-coupled types, although the difference depended on the quality of the V_{SS} layout. In the figure, the input voltage of address Ai is compared with a reference V_{REF} (1.6 V) to discriminate between a high logic level (H) and a low logic level. The resultant differential signal developed between N_1 and N_2 is quickly amplified to a full V_{DD} by the cross-coupled amplifier, as a result of the application of $\overline{\text{RAS}}_2$. After that, the complementary addresses a_i and $\overline{a}_i$ are generated by the application of $\overline{\text{RAS}}_4$ to address latches (ALCs). Note that during standby periods (i.e. $\overline{\text{RAS}}$: H) both a_i and $\overline{a}_i$ are kept low and there is no current path in the buffer.

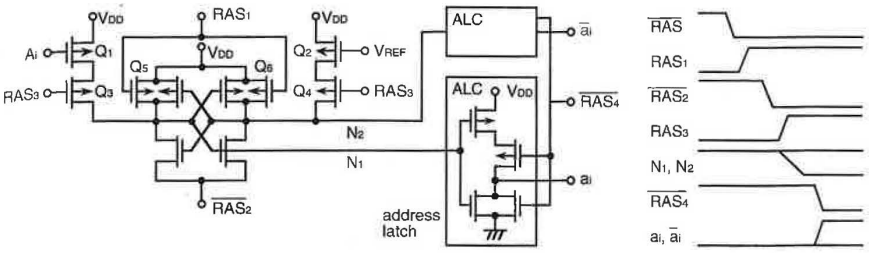


Fig. 3.42. The cross-coupled address buffer [3.21]

Figure 3.43 shows a typical address transition detector (ATD) [3.22], although the variations are shown in Fig. 7.18. Exclusive OR of a'_0 and the address delayed by τ generates a short pulse every a'_0 transition. All the short pulses generated from all the address input transitions are summed up to one ATD pulse, $\overline{EQ}$. Thus, an ATD pulse is generated at any address transition so as to control internal circuits instead of external clocks. If any address transition is quickly detected by ATD and the resultant ATD signal precharges the I/O line in advance, a data line will be selected using addresses just after the transition, so that a data on the data line is outputted on the I/O line without waiting for the I/O precharging. Thus, a long I/O precharging time can be concealed. The ATD signal reduces the power dissipation of main amplifier by cutting the dc current during periods when it is not needed.

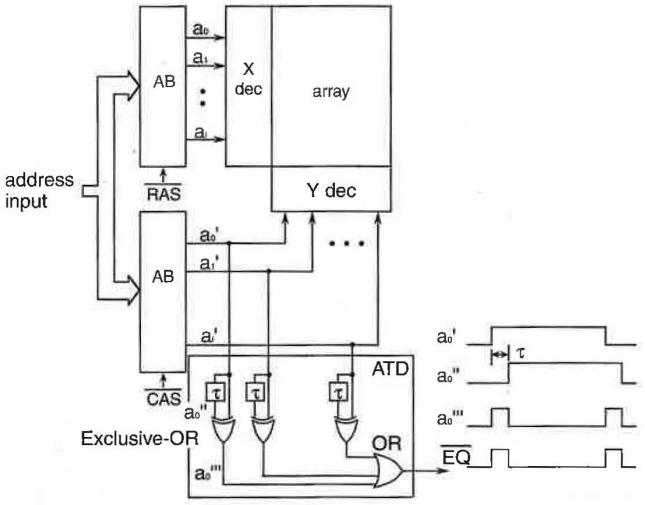


Fig. 3.43. The address transition detector (ATD) scheme [3.22]. AB, address buffers

3.6.2 The Address Decoder

Major concerns for the address-decoder block are power dissipation, speed, and area, because the block includes a huge number of circuits and occupies quite a large segment of the chip.

There are two kinds of decoder; row decoders and column decoders. In DRAM design, unlike SRAM designs, the circuit configurations of the two are totally different. Each row decoder must be a dynamic circuit, while each column decoder can be a static circuit, as explained previously. Note that to precharge all the row decoders without any dc current path, all of the complementary addresses are fixed at a low level during a precharge period, as shown in Fig. 3.42.

Figure 3.44 shows dynamic and static decoders, exemplified by two-bit address signals. There are two kinds of dynamic decoder in DRAM applications; NOR and NAND decoders. First, all of the output nodes (X_0–X_3) are precharged to V_{DD} by transistors Q_P, while keeping all addresses low. Then, according to the succeeding valid address signals, the output nodes are discharged or kept high. Obviously, in NOR decoders all output nodes except for a selected one are discharged, while in NAND decoders all output nodes except for a selected one are kept high. Thus, NOR decoders suffer from a drawback of the large charging and discharging power. The power increases with memory capacity, because an increased number of the nodes – for example, a few thousands, for multimegabit DRAMs – is involved. On the other hand, in NAND decoders only one node is discharged or charged up, independently of memory capacity. NAND decoders, however, suffer the drawback of a slower speed, because the node is discharged by serially connected (i.e. stacked) transistors. The number of stacked transistors is also limited by the body effect of the transistor. Static NAND decoders for the column are simple, as shown in Fig. 3.44c.

Figure 3.45 shows applications of dynamic decoders to the row and static decoders to the column. In dynamic decoders, a word line WL is activated by an RX pulse that is applied after the decoder output Xi has been settled. In the selected decoder, the Q_W gate–drain (i.e. RX terminal) capacitance C_{GD} is large, because the Q_W gate stays at the high level of $V_{DD} - V_T$. Thus, an RX pulse positively going from 0 V to V_{DD} can boost the Q_W gate voltage. The boost ratio is large, because a diode Q_D isolates the Q_W gate from the node Xi capacitance. Due to the resulting boosted gate voltage, to higher than $V_{DD} + V_T$, the word line is quickly driven to V_{DD}. In the non-selected decoders, an RX pulse application never raises the Q_W gate voltages, since the gate voltages are 0 V and thus their C_{GD} values are almost zero. For NOR decoders, even the heavily capacitive Q_W gate is quickly discharged by at least one transistor of the decoder. For NAND decoders, however, to accomplish rapid decoding, Q_W is driven with the help of a small CMOS inverter, whose input capacitance is small enough to be quickly driven even by stacked transistors. Despite the area penalty, the inverter added to each

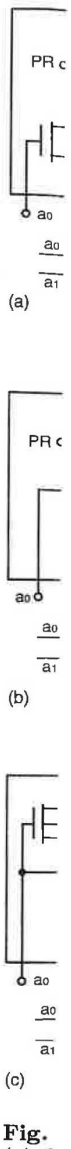


Fig.
(**a**) D

deco
reasc
ders

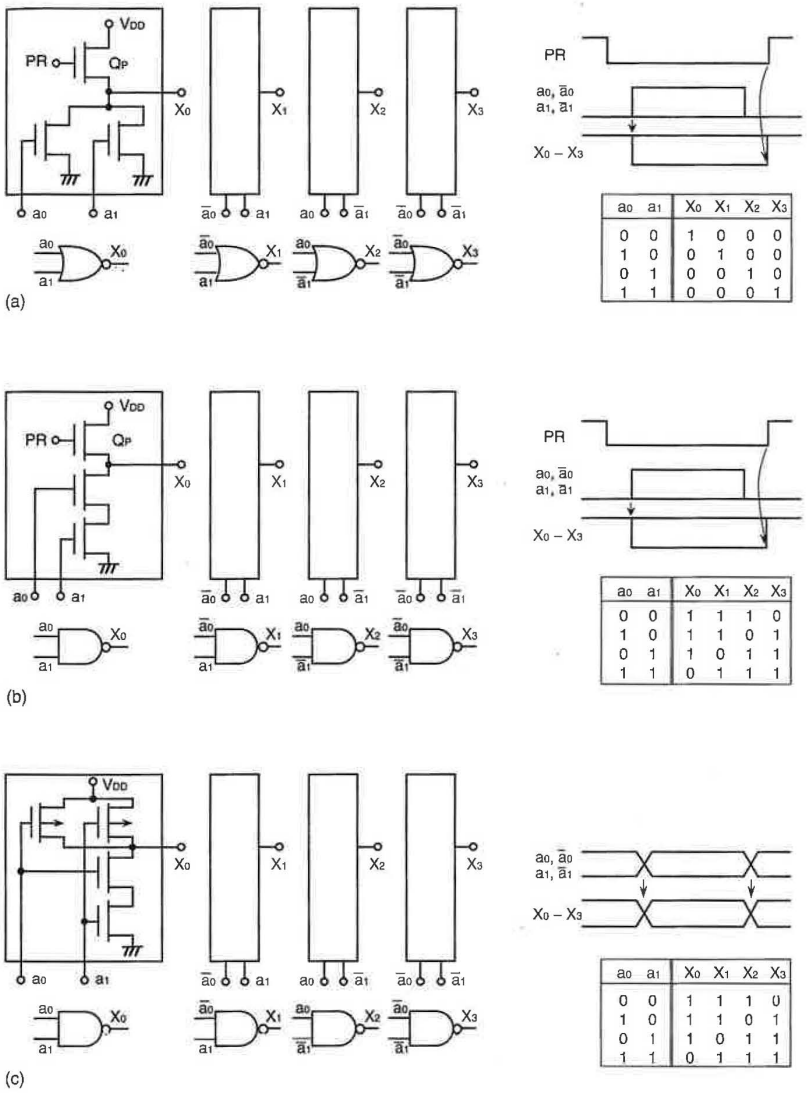


Fig. 3.44. Decoders and operations, exemplified by two address bits [3.4]. (**a**) Dynamic NOR; (**b**) dynamic NAND; (**c**) static NAND

decoder never increases the decoder power because it is a CMOS inverter. The reason why NMOS NOR decoders have been replaced by CMOS NAND decoders since the 1 Mb generation is just to reduce the ever-increasing decoder

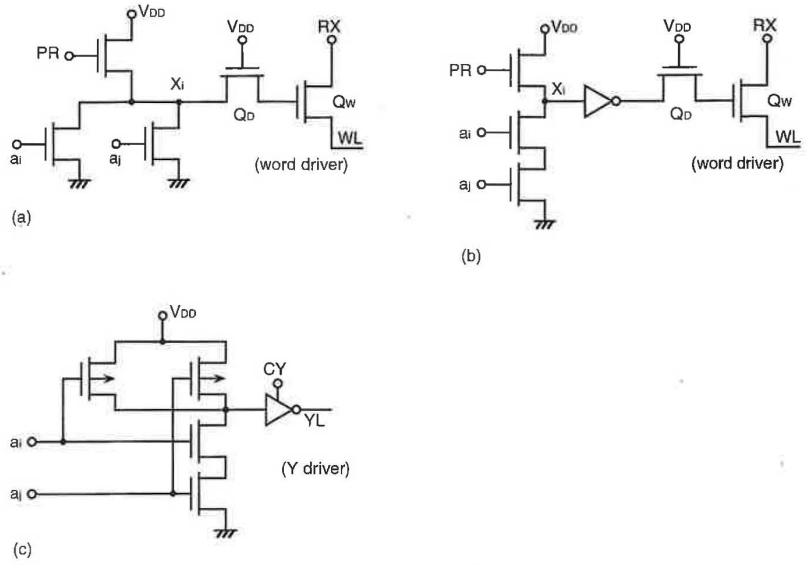
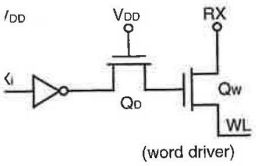


Fig. 3.45. Applications of decoders to word and column drivers [3.4]. (**a**) Dynamic NOR; (**b**) dynamic NAND; (**c**) static NAND

power. Even for a small memory capacity chip of 1 Mb, CMOS NAND decoders have reduced the decoder power down to 4% of that needed for NMOS NOR decoders [3.7, 3.23] as shown in Fig. 3.17. However, to improve the performance of CMOS NAND decoders further, it is essential to reduce the number of stacked transistors. This is realized by predecoding schemes [3.21], as follows.

A predecoding scheme achieves a faster decoding and area reduction of a decoder while reducing the number of stacked transistor in a CMOS NAND decoder. In addition, it reduces the input capacitance and the necessary address input lines of the decoder. Figure 3.46 compares predecoding schemes [3.4]. Direct decoding, 2 bit predecoding, and 3 bit predecoding are shown in Figs. 3.46a–c, respectively. A circle in the figures denotes a transistor connection. For example, when a high level is applied to a_0 and a_1 in Fig. 3.46a, decoders X_0, X_4, and X_8 are selected as a result of NAND decoding. Each 2 bit predecoder can select one of four address input lines coming to the decoders by using two sets of complementary addresses from two address buffers, while each 3 bit predecoder can select one of eight address input lines by using three sets of complementary addresses. Here, let us cite an example of a total external address bits of 6 bits (A_0–A_5). The numbers of address lines to the decoders for direct decoding, 2 bit decoding, and 3 bit decoding are 12, 12, and 16, respectively. The numbers of transistors



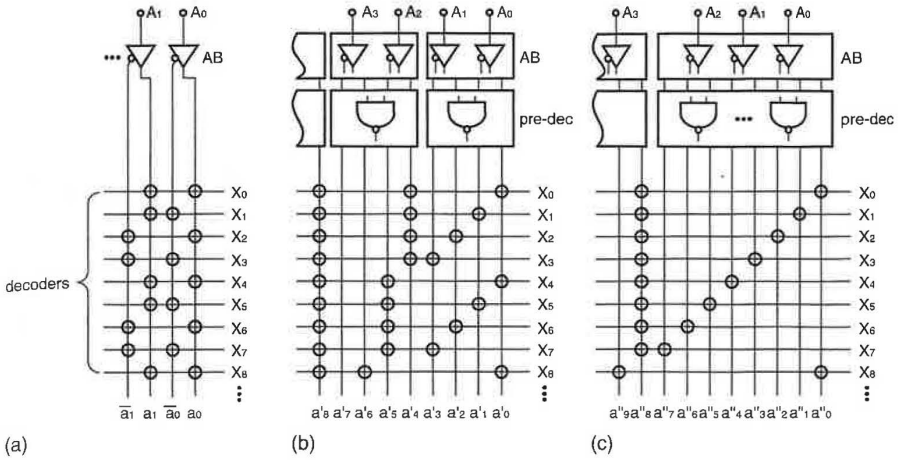


Fig. 3.46. Predecoding [3.4]. (**a**) no predecoding; (**b**) 2-bit predecoding; (**c**) 3-bit predecoding

connected to each address line are 32, 16, and 8, and the numbers of transistors consisting of each decoder are six, three, and two, in the same order. Hence, the predecoding schemes achieve a higher speed with reduction of the address-line capacitance and the number of stacked transistors needed for each NAND decoder, given an acceptable number of address lines. They also reduce the decoder area. The resulting improvement in speed offsets an additional delay caused by the predecoders. Here, predecoding schemes with more than 4 bits are not practical because of a rapid increase in the number of address lines.

Figure 3.47 shows a reduction in the delay of an address line [3.24], which is an example of the buffer insertions shown in Fig. 3.20. Quite a long delay time is developed, despite the aluminum address line, because the line becomes resistive and capacitive due to fine patterning, and is loaded by the distributed gate capacitances of the decoder transistors, as shown in Fig. 3.47a. However, the delay is reduced by the multidivided decoder shown in Fig. 3.47b. The resulting block decoder is driven by a buffer. Each block is constructed so as to correspond to a subarray and only one block is selectively activated by the subarray activation pulse Φ_i.

3.6.3 The Word Driver

A word driver needs to be designed carefully – more so than a column driver – because its load has the following unique features:

1. *A Boosted Word Voltage.* The need for a boosted word voltage, for full write and read operations, means that row decoders and word drivers

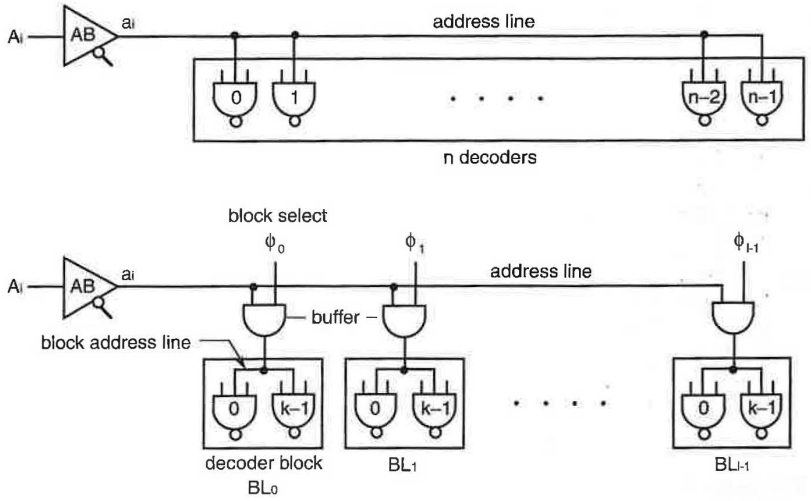
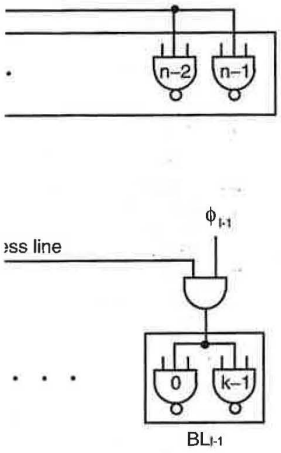


Fig. 3.47. The delay reduction of an address line running on decoders (*upper*) by insertion of buffers (*lower*) [3.4, 3.24]

have complicated designs. On the other hand, column-relevant circuits, such as column decoders and drivers, do not need any boosting. Even without boosting, an amplified signal voltage from a data line can be transmitted to the I/O line, and a data-input voltage of V_{DD} can be fully transmitted from the I/O line to the data line with the help of the CMOS sense amplifier.

2. *A Large Word-Line Capacitance and Resistance.* The electrical characteristics of a word line differ from those of column line YL in the shared Y decoder scheme. Word-line capacitance is quite heavy, because of connections with many memory cells. On the other hand, column-line capacitance is light, because a small number of transistors, equal to the number of data-line divisions (i.e. q in Fig. 3.14), are connected to a YL line. Thus, a larger word driver is needed. As for line resistance, there is also a large difference between the two. A word line made of poly-Si or polycide is resistive for a folded data-line cell, while a column line is made of aluminum.
3. *Loss of Stored Information.* If the stored charges of a non-selected cell are allowed to escape to a data line through the transfer transistor, the refresh time or S/N ratio of the cell is degraded, as discussed in Chap. 4. Thus, noise suppression is essential on non-selected word lines. Moreover, to avoid loss of stored information, data-line precharging must be started after completely turning off the word pulse. Such considerations are not



ning on decoders (*upper*) by

, column-relevant circuits,
need any boosting. Even
from a data line can be
oltage of V_{DD} can be fully
vith the help of the CMOS

ice. The electrical charac-
olumn line YL in the sha-
is quite heavy, because of
e other hand, column-line
of transistors, equal to the
.4), are connected to a YL
for line resistance, there is
rd line made of poly-Si or
while a column line is made

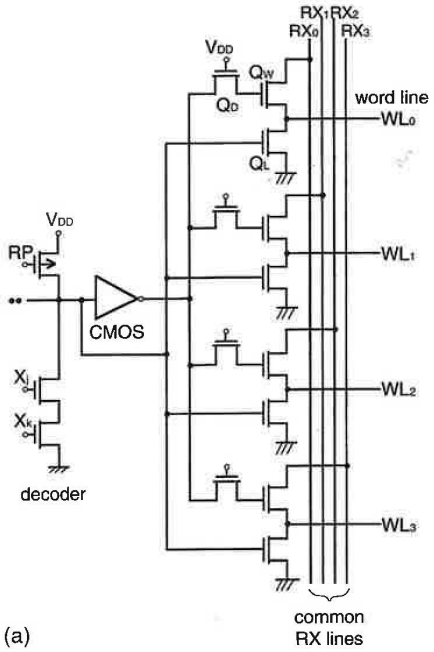
rges of a non-selected cell
he transfer transistor, the
d, as discussed in Chap. 4.
cted word lines. Moreover,
recharging must be started
uch considerations are not

needed for the column. Here, an explanation of the column driver is omitted in what follows, because the driver is almost the same as in Fig. 3.45.

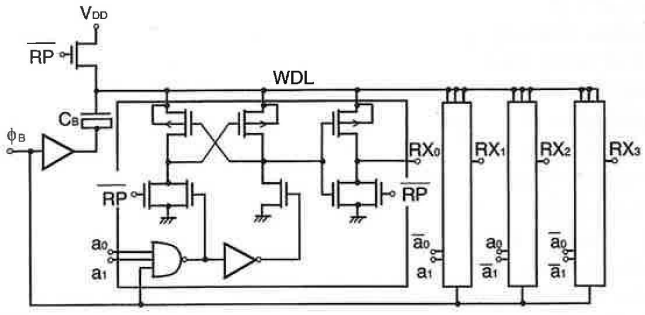
A Basic Word Driver. Figure 3.48 shows the basic unit of conventional word drivers [3.13, 3.21, 3.23]. Each word line is divided into two, to reduce word-line delay (see Fig. 3.40), and the resulting divided word line has its own word driver, Q_W. Since a CMOS NAND decoder cannot be placed at a tight word-line pitch, it is shared with two sets (left and right) of four word drivers, although only the right section is shown in the figure. Address signals X_j and X_k are inputted from 2-bit predecoders to the decoder. Each of the four word drivers selected by the decoder is selectively driven by decoded row select lines RX (RX_0–RX_3), enabling the corresponding word line to be driven. The RX drivers in Fig. 3.48b provide a boost word voltage to one of RXs as a result of two-address bit (a_0, a_1) decoding, as follows. At first, node WDL is precharged almost to V_{DD} during the precharge period (i.e. $\overline{RP}$: H) and all address signals and thus all RX lines are fixed at 0 V. When the addresses have been valid after starting activation with $\overline{RP}$: L, a clock Φ_B generated by the $\overline{RAS}$ buffer is applied, so that only one selected RX line is driven at a high enough voltage, boosted by C_B.

The latch transistor Q_L in the word driver suppresses noise on each non-selected word line while fixing the word line at 0 V as follows. During precharge period all of the non-selected word lines are fixed at 0 V because the Q_Ls are turned on. When one set of four word lines is selected by a decoder, the gate voltages of the corresponding four Q_Ls are changed from V_{DD} to 0 V and the Q_Ls are thus cut off. At this moment, the gate voltages of the corresponding four word-driver transistors Q_W are increased from 0 V to $V_{DD} - V_T$. During this process, the word lines are not at any floating voltage that easily couples noise to the lines, because the four word lines WL_0–WL_3 are fixed at the voltages (i.e. 0 V) of RX_0–RX_3 through the respective Q_Ws. After that, for example, RX_0 is selected and a V_{DH} pulse is sent to WL_0, fixing the remaining non-selected word lines at 0 V. Note that all of the word lines belonging to the non-selected decoders continue to be fixed at 0 V, because the Q_Ls are turned on. Thus, the noise coupled to each non-selected word line, even during signal amplifications performed at a large voltage swing of V_{DD} or a half-V_{DD} on data lines, can be sufficiently suppressed. To further reduce noise on the word lines, another scheme of an additional transistor, which is controlled by address signals, on each word line has been proposed [3.25].

High-speed driving of the RX line is also important, because a long delay is developed by the heavy loading of the large Q_Ws and the long RX line running along a memory array. An RX driver placed at the end of the subarrays in Fig. 3.49a increases the line delay. However, a RX driver for each subarray in Fig. 3.49b [3.26], which is similar to the scheme in Fig. 3.47, shortens the delay. In this scheme, only one subarray is selected by the address signals and a subarray selection signal Φ_i. The node WDL of the RX line in Fig. 3.48 corresponds to a line WDL that is common to a number of subarrays in



(a)



(b)

Fig. 3.48. The configuration of word drivers and relevant circuits [3.4, 3.13, 3.21, 3.23]. (**a**) Word drivers; (**b**) RX drivers

Fig. 3.49b. As soon as $\overline{\text{RAS}}$ activation starts, the heavily capacitive WDL line is boosted. After that, the addresses are valid and a subarray is thus selected so that a V_{DH} pulse is applied to a common RX line belonging to the subarray. Consequently, the boosting time of the WDL line can be concealed and the RX line that the RX driver must drive is shortened to one subarray. Thus, the driving speed is improved.

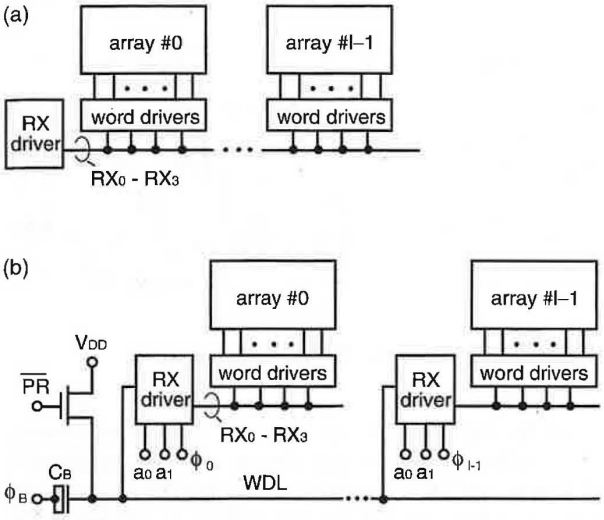
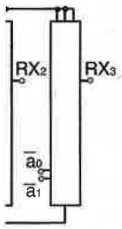


Fig. 3.49. Driving schemes of RX lines [3.4, 3.26]. (**a**) Direct driving of RX lines; (**b**) selective driving of multidivided RX lines

The Voltage-Stress Relaxed Word Driver. This is the word driver that must operate at the highest voltage in a DRAM chip. Therefore, many circuits have been proposed to relax the voltage stress applied at normal and/or burn-in tst operations. They are categorized as the use of PMOS transistor in the word driver instead of NMOS transistor, the changing of the boost ratio according to V_{DD}, and the use of a well-regulated boosted dc voltage.

The PMOS Output Transistor [3.4]. The stress voltages applied to the word-driver output transistor are high, because the word-line voltage must be higher than $V_{DD} + V_{TM}$ (where V_{TM} is the threshold voltage of the memory-cell transistor). However, the PMOS transistor can relax the voltage stress, ensuring high reliability, as follows. For the NMOS transistor in Fig. 3.50a, in order that an increased voltage V_{DH} ($> V_{DD} + V_{TM}$) boosted by C_B at the drain (i.e. RX) is available at the source (i.e. word line), the gate voltage must be boosted to a level that is somewhat higher than $V_{DH} + V_{TD}$ by utilizing C_{GD}. Here, V_{TD} is threshold voltage of the output transistor, which is usually smaller than V_{TM} due to the narrow channel effect, and so on. This dual boosting raises the gate voltage to a considerably high voltage. The PMOS output transistor in Fig. 3.50b does not need dual boosting, since the activation is performed only by lowering the gate voltage from V_{DH} to 0 V. Hence, the PMOS transistor reduces the gate voltage by at least V_{TD}. In actual practice, the difference in gate voltage between the PMOS and NMOS transistors becomes larger, due to variations in the boost ratio at the



rs and relevant circuits

e heavily capacitive WDL
id and a subarray is thus
n RX line belonging to the
VDL line can be concealed
shortened to one subarray.

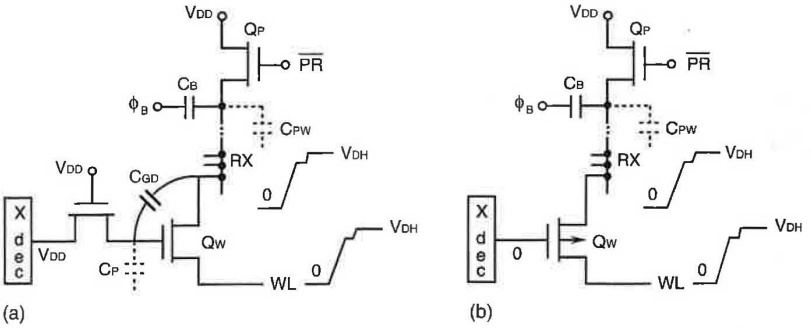
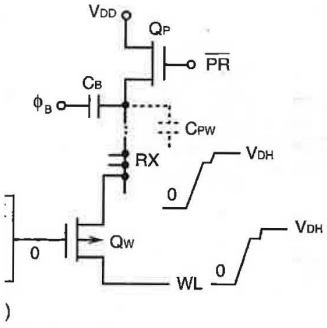


Fig. 3.50. Word boosting for NMOS (**a**) and PMOS (**b**) drivers [3.4].

NMOS transistor caused by variations in the V_{DD} and fabrication-process. The PMOS transistor may allow the gate–source voltage to be large enough to achieve a faster speed despite the lower conductance of the PMOS, compared to the NMOS transistor, in which the gate–source voltage is usually insufficient. Moreover, for lower-V_{DD} operation, the boost ratio must be larger, since there is a minimal threshold voltage for V_{TD} to prevent a degradation of the V_{DH} level caused by a subthreshold current; this is discussed in Chap 8. Thus, the PMOS transistor has been widely used instead of the NMOS transistor.

The Varied Boost-Ratio Driver. There are some operating modes in which MOS devices must not break down even when the operating voltage varies widely. These are the burn-in test mode, which ensures device reliability by applying an increased voltage, and battery operations, which require a wide voltage margin. Figure 3.51a shows a word driver for boosting a well-regulated low-voltage V_{DL}, which is lowered from V_{DD} (= 3.3 V) using an on-chip voltage down-converter [3.27]. The node WDL, which corresponds to WDL in Figs. 3.48 and 3.49, is boosted by activating $\overline{\Phi}_B$ to a low level after it has been precharged to V_{DL}. An excessively high stress voltage is applied to devices in the burn-in test mode, since the boost ratio is almost constant. A voltage down-converter dedicated to the test mode relaxes the stress voltage in a high-voltage region as follows. The reference voltage V_{RN} increases when V_{DD} is increased. Eventually, however, it is clamped at a voltage of two MOS-diode drops and thus V_{DL} becomes equal to $r_1 V_{RN}$. Here, r_1 is a resistance division ratio. Although V_{DL} increases slightly with V_{DD} because of a slight increase in the diode drops caused by the increased diode current, V_{DL} is determined in turn by the other voltage down-converter for the burn-in mode. The converter generates a boosted voltage, monitoring the threshold voltage V_{TM} of the memory-cell transistor as follows. The input voltage V_{RB} of the comparator is given by



IOS (**b**) drivers [3.4].

V_{DD} and fabrication-process. ce voltage to be large enough to tance of the PMOS, compared urce voltage is usually insuffi- oost ratio must be larger, since o prevent a degradation of the is is discussed in Chap 8. Thus, tead of the NMOS transistor.

me operating modes in which n the operating voltage varies h ensures device reliability by erations, which require a wide er for boosting a well-regulated) (= 3.3 V) using an on-chip L, which corresponds to WDL ing $\overline{\Phi}_B$ to a low level after it igh stress voltage is applied to oost ratio is almost constant. mode relaxes the stress voltage ce voltage V_{RN} increases when nped at a voltage of two MOS- $_1V_{RN}$. Here, r_1 is a resistance with V_{DD} because of a slight creased diode current, V_{DL} is converter for the burn-in mode. nitoring the threshold voltage The input voltage V_{RB} of the

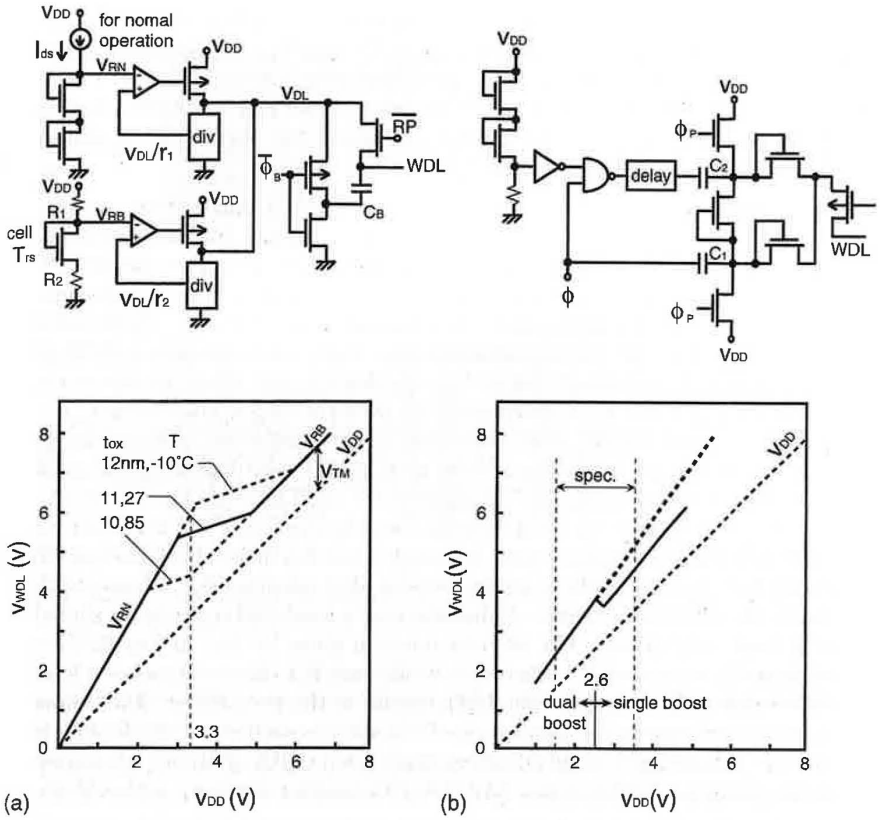


Fig. 3.51. Varied boost ration word drivers [3.4, 3.27, 3.28]. Switching of raised voltages (**a**) and boost number (**b**)

$$V_{RB} = \frac{R_2}{R_1 + R_2}\left\{V_{DD} + \frac{R_1 + R_2}{R_2} V_{TM}\right\} ,$$
$$\therefore\ V_{WDL} = BV_{DL} = B(r_2 V_{RB}) ,$$

where $V_{DD} \gg V_{TM}$, r_2 is a resistance division ratio, and B is a boost ratio. Hence,

$$V_{WDL} = V_{DD} + r_2 B\, V_{TM} ,$$

for $Br_2R_2/(R_1 + R_2) = 1$. An additional voltage r_2BV_{TM} is thus V_{TM} for $r_2B = 1$, which minimizes the necessary boosted voltage. The boosted word voltage V_{WDL} can track variations of V_{TM} because a memory-cell transistor is used.

Figure 3.51b shows a voltage up-converter for a battery operation [3.28]. It features dual boosting in the low-V_{DD} region, but single boosting in the

high-V_{DD} region. Thus, a raised S/N ratio even in the low-V_{DD} region, and the suppression of an excessive stress voltage in the high-V_{DD} region are obtained. This is essential to ensure the wide operational range of $V_{DD} = 2.6 \pm 1$ V that is inherent in battery operations. A voltage up-converter [3.29] described in Chap. 5 is a variation of this concept, in which the boost ratio is almost continuously changed over a wide range of V_{DD} values.

The Raised dc Supply-Voltage Driver [3.29–3.32]. A quasi-static supply-voltage (V_{DH}) generator, discussed in Chap. 5, would not only minimize the stress voltage to devices, but also enable fast operation, as follows. Figure 3.52a shows a word driver using the dc power supply V_{DH}. One decoder is shared with four word lines through transistors controlled by the decoded signals Φ_{X0}–Φ_{X3}. At the beginning of activation, the non-selected three of the four signals, which were all at V_{DD}, go down to 0 V while the remaining selected one goes to V_{DD}. After that, the decoder output goes down to 0 V and the selected PMOS word transistor is thus turned on, with a change in the gate voltage from V_{DH} to 0 V, so that the word line is activated at V_{DH}. During this process, the remaining three word lines are latched at 0 V. This driver eliminates the need to drive a heavily capacitive RX line and thus eventually realizes a higher speed, although it needs a little additional time to ensure the sequence of the enabling decoder after enabling Φ_{Xi}. Figure 3.52b shows the other word driver. A decoder and a level shifter are both shared with four word drivers. The decoder function given by Φ_{Xi} in Fig. 3.52a is replaced by a decoded RX scheme, in which each RX driver comprises a level shifter and an inverter (see Fig. 3.54), similar to the word driver. The timing requirement between the application of RX and the enabling of the decoder is relaxed, permitting even an advanced application of RX. Without a boosting effect, which is available at the NMOS word transistor combined with a MOS-

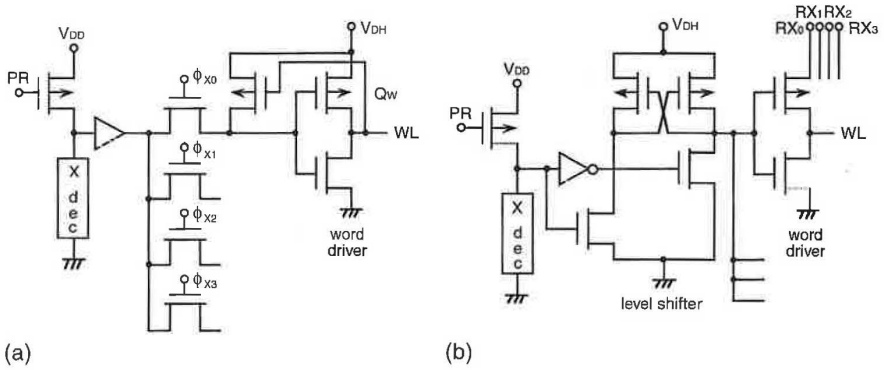
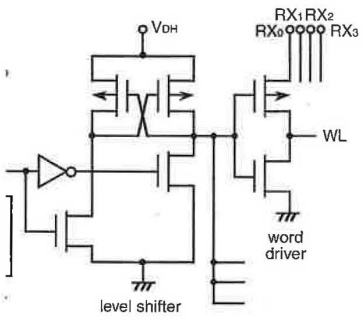


Fig. 3.52. A word driver using a raised dc supply voltage [3.4, 3.29–3.32]. (**a**) Static driver; (**b**) dynamic driver

in the low-V_{DD} region, and the high-V_{DD} region are obtained. range of $V_{DD} = 2.6 \pm 1$ V that ›-converter [3.29] described in ich the boost ratio is almost › values.

3.32]. A quasi-static supply- 5, would not only minimize fast operation, as follows. Fi- ›wer supply V_{DH}. One decoder ›ors controlled by the decoded ion, the non-selected three of vn to 0 V while the remaining ›der output goes down to 0 V us turned on, with a change the word line is activated at word lines are latched at 0 V. ly capacitive RX line and thus eeds a little additional time to ›er enabling Φ_{Xi}. Figure 3.52b , level shifter are both shared given by Φ_{Xi} in Fig. 3.52a is ›h RX driver comprises a level o the word driver. The timing the enabling of the decoder is on of RX. Without a boosting nsistor combined with a MOS-



›oltage [3.4, 3.29–3.32]. (**a**) Static

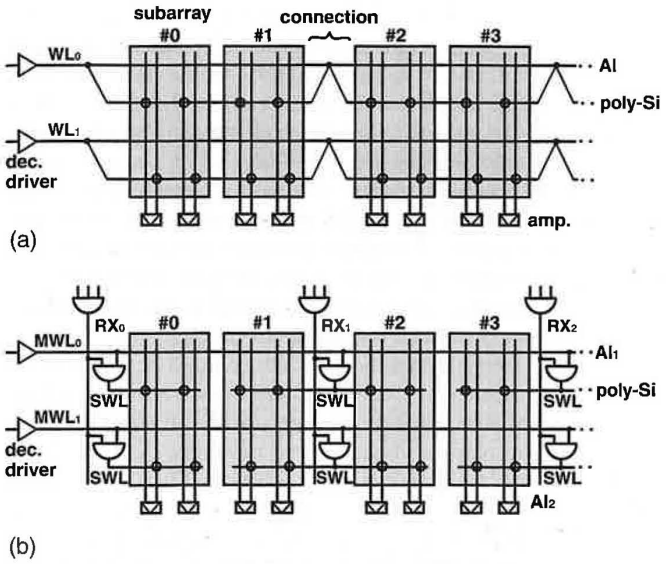


Fig. 3.53. The concept behing partial activation of a multidivided word line [3.18, 3.19, 3.33]. (**a**) No division; (**b**) multidivision

diode shown in Fig. 3.48, an increased word voltage is quickly outputted. This is due to the use of a V_{DH} level shifter.

The Reduction of Word-Line Delay. Even if word-driver speed is improved by using a PMOS transistor, the need to reduce a large word-line delay remains. Although the aluminum-strapped poly-Si or polycide word line, as shown in Fig. 3.53a, has been popular since the 1 Mb generation, a large *RC* word-line delay has been prominent in the 64 Mb generation and beyond. This is because of the ever-finer and longer aluminum line and the ever-increasing number of memory cells connected to a word line. Unfortunately, an increase in the number of word-line divisions causes an area penalty, with a resulting increased number of word drivers.

One solution is the hierarchical word-line structure [3.18, 3.19, 3.33], as conceptually shown in Fig. 3.53b. This is an example of the delay reduction scheme in Fig. 3.19d. One word line is divided into several by the small subword-line (SWL) drivers. All of the subword lines in a row are commonly controlled by a main word line (MWL). Therefore, they are simultaneously activated by selecting a main word line and all of the row select lines (RX_0, RX_1, etc.). The choice of aluminum lines for MWL and RX would improve the speed of the simple aluminum strapping in Fig. 3.53a, despite poly-Si or polycide subword lines. In simple strapping, a word line is heavily loaded with a huge number of memory cells in a row. The resulting heavy capacitance

develops a long delay even on a low-resistance aluminum line. In this case the total delay is approximately the sum of a large MWL delay and the subword-line delay. On the other hand, in the hierarchical structure a main word line is loaded by only quite a small number of AND logic gates, which is the same as the number of word-line divisions. Considering that the number of memory cells connected to one subword line ranges from 256 to 512, the loading for the main word line is quite light, which enables an extremely small delay compared with the subword-line delay. An RX line also enables a small delay because of the aluminum material. The delay could be further shortened if it is driven by a RX driver located at each data-line division. Eventually, the total delay of the hierarchical structure is almost confined to the subword-line delay. Thus, the delay is less than that for simple strapping.

Figure 3.54 shows an actual hierarchical structure applied to a 256 Mb chip [3.18, 3.19, 3.33]. It relaxes the MWL pitch to one-quarter so that aluminum wiring is enabled even on the top surface of the substrate, and it allows SWL drivers to be placed alternately to meet the tight word-line layout pitch. Eight-row word lines, each of which is divided into a number of subword lines (SWL_{00}, $\overline{SWL_{10}}$, ...), are controlled by a set of complimentary main word lines (MWL, $\overline{MWL}$), which are driven by a row decoder and a set of word drivers. One set of four SWL drivers is located at each end of each subarray. One of four SWL drivers is activated by one of the decoded RX lines from a RX driver. The full use of NMOS transistors realizes a small

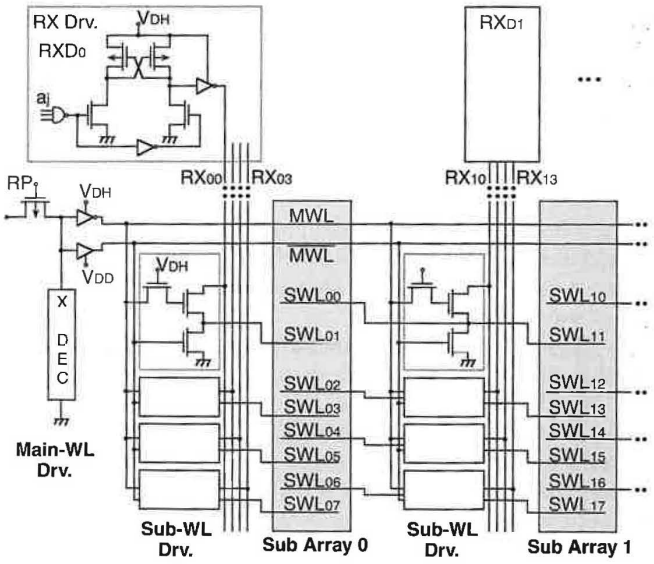


Fig. 3.54. Hierarchical word-line architecture [3.18, 3.19]

APPENDIX B

LIBRARY OF CONGRESS
ONLINE CATALOG

1. Fundamentals of digital logic with Verilog design

LCCN	2002071439
Type of material	Book
Personal name	Brown, Stephen D.
Main title	Fundamentals of digital logic with Verilog design / Stephen Brown and Zvonko Vranesic.
Edition	1st ed.
Published/Created	Boston : McGraw-Hill, c2003.
Description	xx, 844 p. : ill. ; 25 cm. + 1 CD-ROM (4 3/4 in.)
Links	Table of contents only http://www.loc.gov/catdir/enhancements/fy0702/2002071439-t.html
ISBN	0072823151 (alk. paper)
LC classification	TK7868.L6 B76 2003
Related names	Vranesic, Zvonko G.
Subjects	Logic circuits--Design and construction--Data processing. Verilog (Computer hardware description language) Computer-aided design.
Notes	Includes bibliographical references and index.
Series	McGraw-Hill series in electrical and computer engineering
Dewey class no.	621.39/2
CALL NUMBER	TK7868.L6 B76 2002 FT MEADE Copy 1
Request in	Machine Readable Collections - STORED OFFSITE
CALL NUMBER	TK7868.L6 B76 2003 FT MEADE Copy 2
Request in	Machine Readable Collections - STORED OFFSITE

LIBRARY OF CONGRESS ONLINE CATALOG
Library of Congress
101 Independence Ave., SE
Washington, DC 20540

Questions? Ask a Librarian:
https://www.loc.gov/rr/askalib/ask-contactus.html

FUNDAMENTALS OF DIGITAL LOGIC WITH VERILOG DESIGN

Stephen Brown and Zvonko Vranesic
Department of Electrical and Computer Engineering
University of Toronto

Boston Burr Ridge, IL Dubuque, IA Madison, WI New York San Francisco St. Louis
Bangkok Bogotá Caracas Kuala Lumpur Lisbon London Madrid Mexico City
Milan Montreal New Delhi Santiago Seoul Singapore Sydney Taipei Toronto

McGraw-Hill Higher Education
A Division of The McGraw-Hill Companies

FUNDAMENTALS OF DIGITAL LOGIC WITH VERILOG DESIGN

This book is printed on acid-free paper.

International 1 2 3 4 5 6 7 8 9 0 QPF/QPF 0 9 8 7 6 5 4 3 2
Domestic 1 2 3 4 5 6 7 8 9 0 QPF/QPF 0 9 8 7 6 5 4 3 2

ISBN 0-07-282315-1
ISBN 0-07-121322-8 (ISE)

Publisher: *Elizabeth A. Jones*
Senior sponsoring editor: *Carlise Paulson*
Administrative assistant: *Michaela M. Graham*
Executive marketing manager: *John Wannemacher*
Senior project manager: *Jill R. Peter*
Production supervisor: *Kara Kudronowicz*
Lead media project manager: *Judi David*
Senior media technology producer: *Phillip Meek*
Coordinator of freelance design: *Michelle D. Whitaker*
Cover designer: *Rokusek Design*
Cover image: *Stephen Brown and Zvonko Vranesic*
Senior photo research coordinator: *Lori Hancock*
Compositor: *Techsetters, Inc.*
Typeface: *10/12 Times Roman*
Printer: *Quebecor World Fairfield, PA*

Library of Congress Cataloging-in-Publication Data

Brown, Stephen D.
Fundamentals of digital logic with Verilog design / Stephen D. Brown, Zvonko G. Vranesic.—1st ed.
p. cm. (McGraw-Hill Series in electrical and computer engineering)
Includes index.
ISBN 0-07-282315-1
1. Logic circuits—Design and construction—Data processing. 2. Verilog (Computer hardware description language). 3. Computer-aided design. I. Vranesic, Zvonko G. II. Title. III. Series.

TK7868.L6 B76 2003
621.39'2—dc21 2002071439
CIP

INTERNATIONAL EDITION ISBN 0-07-121322-8

www.mhhe.com

To Susan and Anne

About the Authors

Stephen Brown received his B.A.Sc. degree in Electrical Engineering from the University of New Brunswick, Canada, and the M.A.Sc. and Ph.D. degrees in Electrical Engineering from the University of Toronto. He joined the University of Toronto faculty in 1992, where he is now an Associate Professor in the Department of Electrical & Computer Engineering. He is also Director of Software Development at the Altera Toronto Technology Center.

His research interests include field-programmable VLSI technology and computer architecture. He won the Canadian Natural Sciences and Engineering Research Council's 1992 Doctoral Prize for the best Ph.D. thesis in Canada.

He has won four awards for excellence in teaching electrical engineering, computer engineering, and computer science courses. He is a coauthor of two other books: *Fundamentals of Digital Logic with VHDL Design* and *Field-Programmable Gate Arrays*.

Zvonko Vranesic received his B.A.Sc., M.A.Sc., and Ph.D. degrees, all in Electrical Engineering, from the University of Toronto. From 1963–1965, he worked as a design engineer with the Northern Electric Co. Ltd. in Bramalea, Ontario. In 1968 he joined the University of Toronto, where he is now a Professor in the Departments of Electrical & Computer Engineering and Computer Science. During the 1978–79 academic year, he was a Senior Visitor at the University of Cambridge, England, and during 1984–85 he was at the University of Paris, 6. From 1995 to 2000 he served as Chair of the Division of Engineering Science at the University of Toronto. He is also involved in research and development at the Altera Toronto Technology Center.

His current research interests include computer architecture, field-programmable VLSI technology, and multiple-valued logic systems.

He is a coauthor of four other books: *Computer Organization*, 5th ed.; *Fundamentals of Digital Logic with VHDL Design*; *Microcomputer Structures*; and *Field-Programmable Gate Arrays*. In 1990, he received the Wighton Fellowship for "innovative and distinctive contributions to undergraduate laboratory instruction."

He has represented Canada in numerous chess competitions. He holds the title of International Master.

$$f = \overline{w}_2 f_{\overline{w}_2} + w_2 f_{w_2}$$
$$= \overline{w}_2(w_3 + w_1\overline{w}_4) + w_2(\overline{w}_1\overline{w}_3 + \overline{w}_3 w_4)$$

Observe that $\overline{f}_{\overline{w}_2} = f_{w_2}$; hence only two 3-LUTs are needed, as illustrated in Figure 6.14*b*. The LUT on the right implements the two-variable function $\overline{w}_2 f_{\overline{w}_2} + w_2 \overline{f}_{\overline{w}_2}$.

Since it is possible to implement any logic function using multiplexers, general-purpose chips exist that contain multiplexers as their basic logic resources. Both Actel Corporation [2] and QuickLogic Corporation [3] offer FPGAs in which the logic block comprises an arrangement of multiplexers. Texas Instruments offers gate array chips that have multiplexer-based logic blocks [4].

6.2 DECODERS

Decoder circuits are used to decode encoded information. A binary decoder, depicted in Figure 6.15, is a logic circuit with n inputs and 2^n outputs. Only one output is asserted at a time, and each output corresponds to one valuation of the inputs. The decoder also has an enable input, *En*, that is used to disable the outputs; if $En = 0$, then none of the decoder outputs is asserted. If $En = 1$, the valuation of $w_{n-1} \cdots w_1 w_0$ determines which of the outputs is asserted. An n-bit binary code in which exactly one of the bits is set to 1 at a time is referred to as *one-hot encoded*, meaning that the single bit that is set to 1 is deemed to be "hot." The outputs of a binary decoder are one-hot encoded.

A 2-to-4 decoder is given in Figure 6.16. The two data inputs are w_1 and w_0. They represent a two-bit number that causes the decoder to assert one of the outputs $y_0, \ldots, y_3$. Although a decoder can be designed to have either active-high or active-low outputs, in Figure 6.16 active-high outputs are assumed. Setting the inputs $w_1 w_0$ to 00, 01, 10, or 11 causes the output y_0, y_1, y_2, or y_3 to be set to 1, respectively. A graphical symbol for the decoder is given in part (*b*) of the figure, and a logic circuit is shown in part (*c*).

Larger decoders can be built using the sum-of-products structure in Figure 6.16*c*, or else they can be constructed from smaller decoders. Figure 6.17 shows how a 3-to-8 decoder is built with two 2-to-4 decoders. The w_2 input drives the enable inputs of the two decoders. The top decoder is enabled if $w_2 = 0$, and the bottom decoder is enabled if $w_2 = 1$. This concept can be applied for decoders of any size. Figure 6.18 shows how five 2-to-4 decoders can be used to construct a 4-to-16 decoder. Because of its treelike structure, this type of circuit is often referred to as a *decoder tree*.

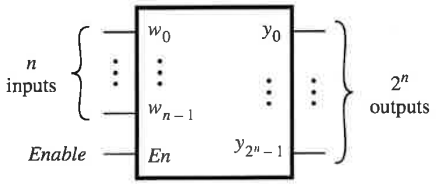
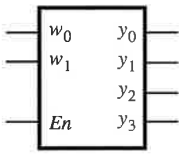


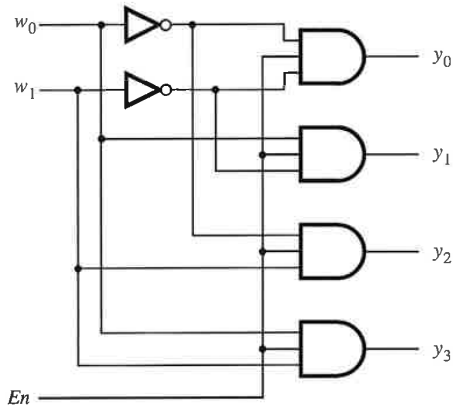
Figure 6.15 An n-to-2^n binary decoder.

En	w_1	w_0	y_0	y_1	y_2	y_3
1	0	0	1	0	0	0
1	0	1	0	1	0	0
1	1	0	0	0	1	0
1	1	1	0	0	0	1
0	x	x	0	0	0	0

(a) Truth table



(b) Graphical symbol



(c) Logic circuit

Figure 6.16 A 2-to-4 decoder.

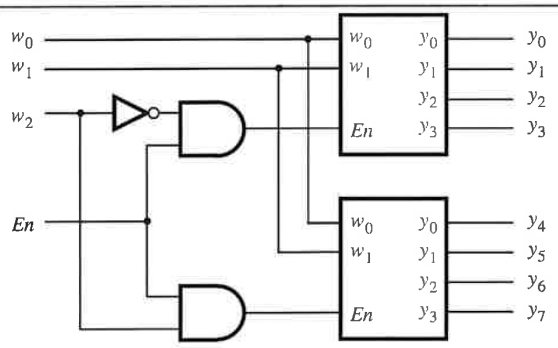


Figure 6.17 A 3-to-8 decoder using two 2-to-4 decoders.

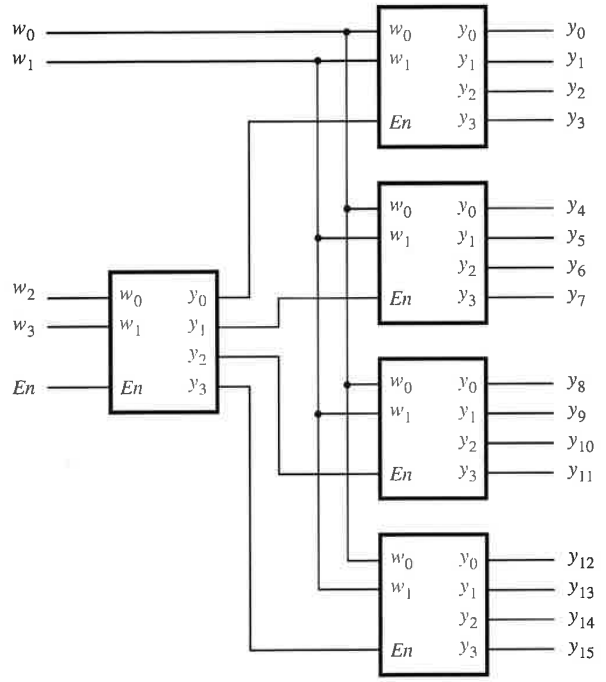


Figure 6.18 A 4-to-16 decoder built using a decoder tree.

Example 6.9

Decoders are useful for many practical purposes. In Figure 6.2*c* we showed the sum-of-products implementation of the 4-to-1 multiplexer, which requires AND gates to distinguish the four different valuations of the select inputs s_1 and s_0. Since a decoder evaluates the values on its inputs, it can be used to build a multiplexer as illustrated in Figure 6.19. The enable input of the decoder is not needed in this case, and it is set to 1. The four outputs of the decoder represent the four valuations of the select inputs.

Example 6.10

In Figure 3.59 we showed how a 2-to-1 multiplexer can be constructed using two tri-state buffers. This concept can be applied to any size of multiplexer, with the addition of a decoder. An example is shown in Figure 6.20. The decoder enables one of the tri-state buffers for each valuation of the select lines, and that tri-state buffer drives the output, f, with the selected data input. We have now seen that multiplexers can be implemented in various ways. The choice of whether to employ the sum-of-products form, transmission gates, or tri-state buffers depends on the resources available in the chip being used. For

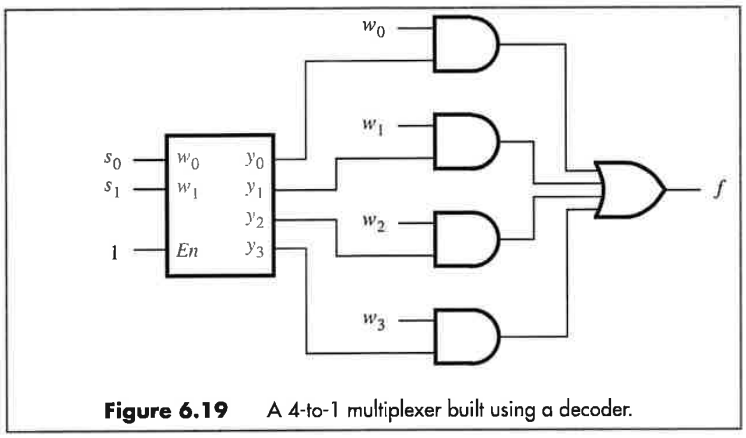


Figure 6.19 A 4-to-1 multiplexer built using a decoder.

instance, most FPGAs that use lookup tables for their logic blocks do not contain tri-state buffers. Hence multiplexers must be implemented in the sum-of-products form using the lookup tables (see problem 6.15).

6.2.1 DEMULTIPLEXERS

We showed in section 6.1 that a multiplexer has one output, n data inputs, and $\lceil \log_2 n \rceil$ select inputs. The purpose of the multiplexer circuit is to *multiplex* the n data inputs onto the single data output under control of the select inputs. A circuit that performs the opposite function, namely, placing the value of a single data input onto multiple data outputs, is

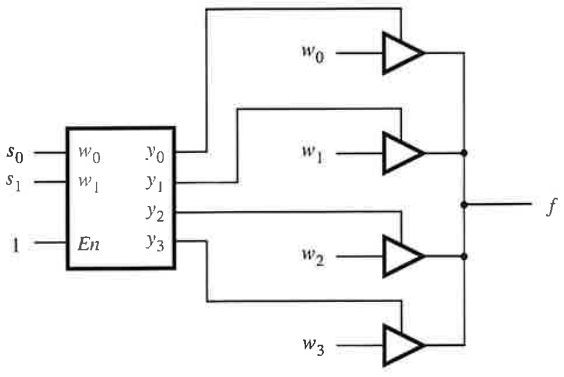


Figure 6.20 A 4-to-1 multiplexer built using a decoder and tri-state buffers.

called a *demultiplexer*. The demultiplexer can be implemented using a decoder circuit. For example, the 2-to-4 decoder in Figure 6.16 can be used as a 1-to-4 demultiplexer. In this case the *En* input serves as the data input for the demultiplexer, and the y_0 to y_3 outputs are the data outputs. The valuation of w_1w_0 determines which of the outputs is set to the value of *En*. To see how the circuit works, consider the truth table in Figure 6.16*a*. When $En = 0$, all the outputs are set to 0, including the one selected by the valuation of w_1w_0. When $En = 1$, the valuation of w_1w_0 sets the appropriate output to 1.

In general, an n-to-2^n decoder circuit can be used as a 1-to-n demultiplexer. However, in practice decoder circuits are used much more often as decoders rather than as demultiplexers. In many applications the decoder's *En* input is not actually needed; hence it can be omitted. In this case the decoder always asserts one of its data outputs, $y_0, \ldots, y_{2^n-1}$, according to the valuation of the data inputs, $w_{n-1} \ldots w_0$. Example 6.11 uses a decoder that does not have the *En* input.

Example 6.11

One of the most important applications of decoders is in memory blocks, which are used to store information. Such memory blocks are included in digital systems, such as computers, where there is a need to store large amounts of information electronically. One type of memory block is called a *read-only memory* (ROM). A ROM consists of a collection of storage cells, where each cell permanently stores a single logic value, either 0 or 1. Figure 6.21 shows an example of a ROM block. The storage cells are arranged in 2^m rows with n

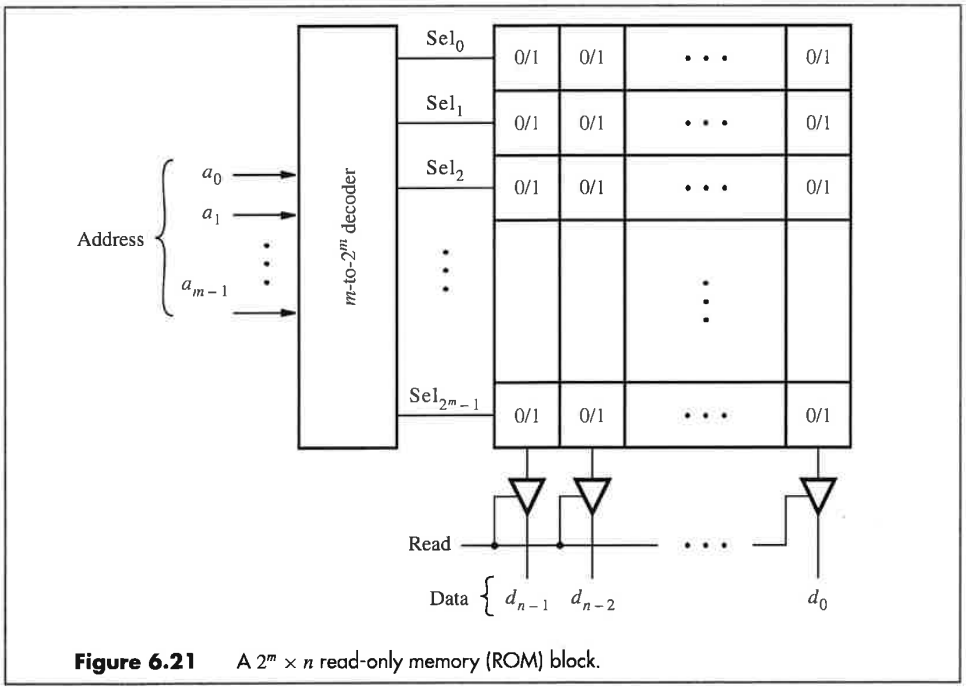


Figure 6.21 A $2^m \times n$ read-only memory (ROM) block.

cells per row. Thus each row stores n bits of information. The location of each row in the ROM is identified by its *address*. In the figure the row at the top of the ROM has address 0, and the row at the bottom has address $2^m - 1$. The information stored in the rows can be accessed by asserting the select lines, Sel_0 to Sel_{2^m-1}. As shown in the figure, a decoder with m inputs and 2^m outputs is used to generate the signals on the select lines. Since the inputs to the decoder choose the particular address (row) selected, they are called the *address* lines. The information stored in the row appears on the data outputs of the ROM, $d_{n-1}, \ldots, d_0$, which are called the *data* lines. Figure 6.21 shows that each data line has an associated tri-state buffer that is enabled by the ROM input named *Read*. To access, or *read*, data from the ROM, the address of the desired row is placed on the address lines and *Read* is set to 1.

Many different types of memory blocks exist. In a ROM the stored information can be read out of the storage cells, but it cannot be changed (see problem 6.31). Another type of ROM allows information to be both read out of the storage cells and stored, or *written*, into them. Reading its contents is the normal operation, whereas writing requires a special procedure. Such a memory block is called a programmable ROM (PROM). The storage cells in a PROM are usually implemented using EEPROM transistors. We discussed EEPROM transistors in section 3.10 to show how they are used in PLDs. Other types of memory blocks are discussed in section 10.1.

6.3 ENCODERS

An encoder performs the opposite function of a decoder. It encodes given information into a more compact form.

6.3.1 BINARY ENCODERS

A *binary encoder* encodes information from 2^n inputs into an n-bit code, as indicated in Figure 6.22. Exactly one of the input signals should have a value of 1, and the outputs present the binary number that identifies which input is equal to 1. The truth table for a 4-to-2 encoder is provided in Figure 6.23*a*. Observe that the output y_0 is 1 when either input w_1 or w_3 is 1, and output y_1 is 1 when input w_2 or w_3 is 1. Hence these outputs can be generated by the circuit in Figure 6.23*b*. Note that we assume that the inputs are one-hot

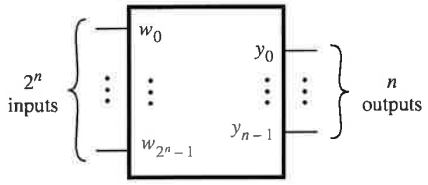


Figure 6.22 A 2^n-to-n binary encoder.