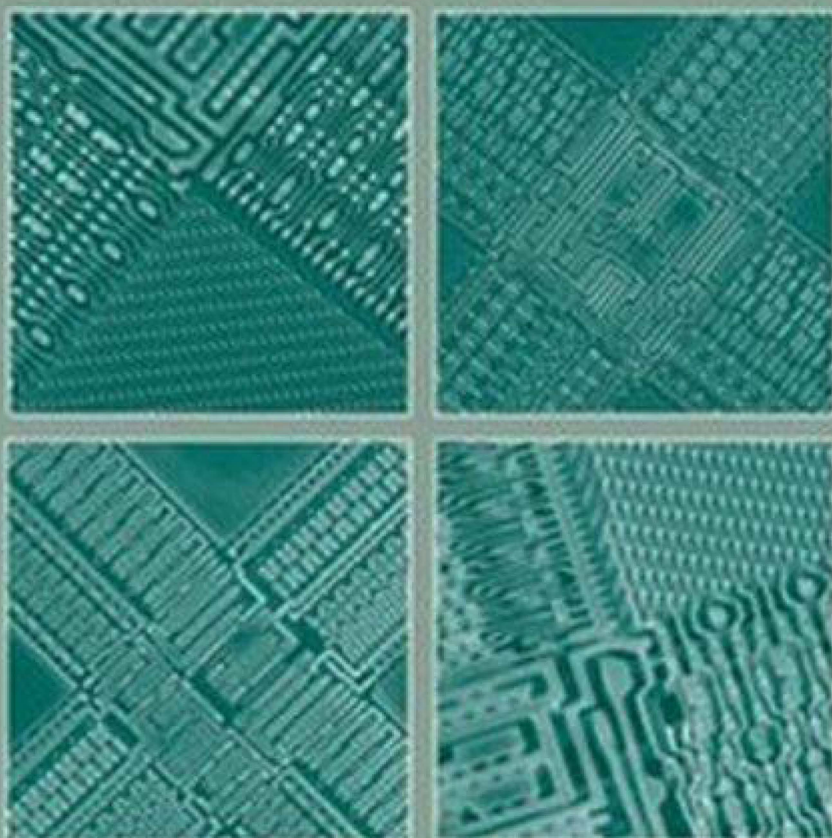


DRAM

CIRCUIT DESIGN

Fundamental and High-Speed Topics



Brent Keeth • R. Jacob Baker
Brian Johnson • Feng Lin

DRAM Circuit Design

IEEE Press
445 Hoes Lane
Piscataway, NJ 08854

IEEE Press Editorial Board
Mohamed E. El-Hawary, *Editor in Chief*

R. Abari	T. G. Croda	S. V. Kartalopoulos
S. Basu	S. Farshchi	M. S. Newman
A. Chatterjee	B. M. Hammerli	
T. Chen	R. J. Herrick	

Kenneth Moore, *Director of IEEE Book and Information Services (BIS)*
Catherine Faduska, *Senior Acquisitions Editor*
Jeanne Audino, *Project Editor*

IEEE Solid-State Circuits Society, Sponsor
IEEE SSCS Liaison to the IEEE Press, Stuart K. Tewksbury

Technical Editing and Illustrations
Mary J. Miller, Micron Technology, Inc.

Technical Reviewers
Jim Frenzel, University of Idaho
Zhao Zhang, Iowa State University

DRAM Circuit Design

Fundamental and High-Speed Topics

**Brent Keeth
R. Jacob Baker
Brian Johnson
Feng Lin**

IEEE Press Series on Microelectronic Systems
Stuart K. Tewksbury and Joe E. Brewer, *Series Editors*

IEEE Solid-State Circuits Society, *Sponsor*



IEEE PRESS



WILEY-INTERSCIENCE
A John Wiley & Sons, Inc., Publication

Copyright © 2008 by the Institute of Electrical and Electronic Engineers, Inc. All rights reserved.

Published by John Wiley & Sons, Inc., Hoboken, New Jersey.
Published simultaneously in Canada.

No part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, electronic, mechanical, photocopying, recording, scanning, or otherwise, except as permitted under Section 107 or 108 of the 1976 United States Copyright Act, without either the prior written permission of the Publisher, or authorization through payment of the appropriate per-copy fee to the Copyright Clearance Center, Inc., 222 Rosewood Drive, Danvers, MA 01923, (978) 750-8400, fax (978) 750-4470, or on the web at www.copyright.com. Requests to the Publisher for permission should be addressed to the Permissions Department, John Wiley & Sons, Inc., 111 River Street, Hoboken, NJ 07030, (201) 748-6011, fax (201) 748-6008, or online at <http://www.wiley.com/go/permission>.

Limit of Liability/Disclaimer of Warranty: While the publisher and author have used their best efforts in preparing this book, they make no representations or warranties with respect to the accuracy or completeness of the contents of this book and specifically disclaim any implied warranties of merchantability or fitness for a particular purpose. No warranty may be created or extended by sales representatives or written sales materials. The advice and strategies contained herein may not be suitable for your situation. You should consult with a professional where appropriate. Neither the publisher nor author shall be liable for any loss of profit or any other commercial damages, including but not limited to special, incidental, consequential, or other damages.

For general information on our other products and services or for technical support, please contact our Customer Care Department within the United States at (800) 762-2974, outside the United States at (317) 572-3993 or fax (317) 572-4002.

Wiley also publishes its books in a variety of electronic formats. Some content that appears in print may not be available in electronic format. For information about Wiley products, visit our web site at www.wiley.com.

Wiley Bicentennial Logo: Richard J. Pacifico

Library of Congress Cataloging-in-Publication Data is available.

ISBN 978-0-470-18475-2

Printed in the United States of America.

10 9 8 7 6 5 4 3 2 1

more bitlines we use in an array, the longer the delay through the wordline (Figure 1.13).

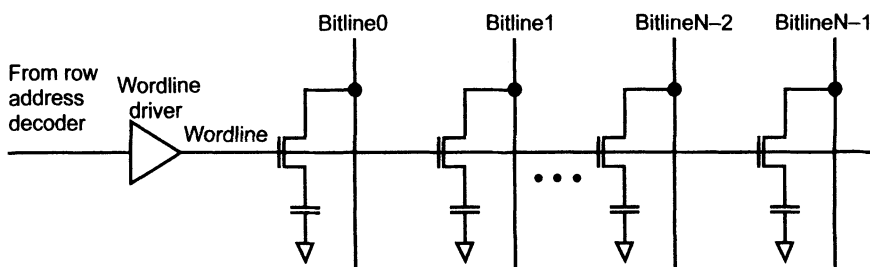


Figure 1.13 Row of N dynamic memory elements.

If we drive the wordline on the left side of Figure 1.13 HIGH, the signal will take a finite time to reach the end of the wordline (the wordline on the right side of Figure 1.13). This is due to the distributed resistance/capacitance structure formed by the resistance of the polysilicon wordline and the capacitance of the MOSFET gates. The delay limits the speed of DRAM operation. To be precise, it limits how quickly a row can be opened and closed. To reduce this RC time, a polycide wordline is formed by adding a silicide, for example, a mixture of a refractory metal such as tungsten with polysilicon, on top of polysilicon. Using a polycide wordline will have the effect of reducing the wordline resistance. Also, additional drivers can be placed at different locations along the wordline, or the wordline can be *stitched* at various locations with metal.

The limitations on the additional number of wordlines can be understood by realizing that by adding more wordlines to the array, more parasitic capacitance is added to the bitlines. This parasitic capacitance becomes important when sensing the value of data charge stored in the memory element. We'll discuss this in more detail in the next section.

1.1.2.3 Memory Array Size. A comment is in order about memory array size and how addressing can be used for setting word and page size. (We'll explain what this means in a moment.) If we review the block diagram of the 4k DRAM shown in Figure 1.9, we see that two 2k-DRAM memory arrays are used. Each 2k memory is composed of 64 wordlines and 32 bitlines for 2,048 memory elements/address locations per array. In the block diagram, notice that a single bit, coming from the column decoder, can be used to select data, via the bitlines, from Array0 or Array1.

From our discussion earlier, we can open a row in Array0 while at the same time opening a row in Array1 by simply applying a row address to the input address pins and driving RAS^* LOW. Once the rows are open, it is a simple matter of changing the column address to select different data asso-

ciated with the same open row from either array. If our word size is 1 bit, we could define a page as being 64 bits in length (32 bits from each array). We could also define our page size as 32 bits with a 2-bit word for input/output. We would then say that the DRAM is a 4k DRAM organized as 2k x 2. Of course, in the 4k DRAM, in which the number of bits is small, the concepts of page reads or size aren't too useful. We present them here simply to illustrate the concepts. Let's consider a more practical and modern configuration.

Suppose we have a 64-Meg DRAM organized as 16 Meg x 4 (4 bits input/output) using 4k row address locations and 4k column address locations (12 bits or pins are needed for each 4k of addressing). If our (sub) memory array size is 256kbits, then we have a total of 256 memory arrays on our DRAM chip. We'll assume that there are 512 wordlines and 512 bitlines (digitlines), so that the memory array is logically square. (However, physically, as we shall see, the array is not square.) Internal to the chip, in the address decoders, we can divide the row and column addresses into two parts: the lower 9 bits for addressing the wordlines/bitlines in a 256k memory array and the upper 3 bits for addressing one of the 64 "group-of-four" memory arrays (6 bits total coming from the upper 3 bits of the row and column addresses).

Our 4-bit word comes from the group-of-four memory arrays (one bit from each memory array). We can define a page of data in the DRAM by realizing that when we open a row in each of the four memory arrays, we are accessing 2k of data (512 bits/array x 4 arrays). By simply changing the column address without changing the row address and thus opening another group-of-four wordlines, we can access the 2k "page" of data. With a little imagination, we can see different possibilities for the addressing. For example, we could open 8 group-of-four memory arrays with a row address and thus increase the page size to 16k, or we could use more than one bit at a time from an array to increase word size.

1.1.2.4 Refreshing the DRAM. Refreshing the DRAM is accomplished by sequentially opening each row in the DRAM. (We'll discuss how the DRAM cell is refreshed in greater detail later in the book.) If we use the 64-Meg example in the last section, we need to supply 4k row addresses to the DRAM by changing the external address inputs from 000000000000 to 111111111111 while clocking the addresses into the DRAM using the falling edge of *RAS**. In some DRAMs, an internal row address counter is present to make the DRAM easier to refresh. The general specification for 64-Meg DRAM Refresh is that all rows must be refreshed at least every 64 ms, which is an average of 15.7 μ s per row. This means, that if the Read cycle time t_{RC} is 100 ns (see Figure 1.11), it will take $4,096 \cdot 100$ ns or