

Fundamentals of Computer Systems

Memory

Martha A. Kim

Columbia University

Fall 2013

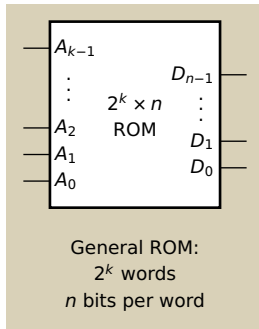
Memory Interface

Data stored in *word* units

A word is several bytes (powers of two are typical)

write operations store data to memory

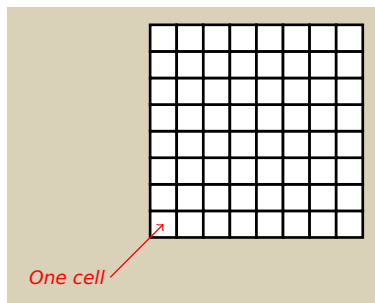
read operations retrieve data from memory



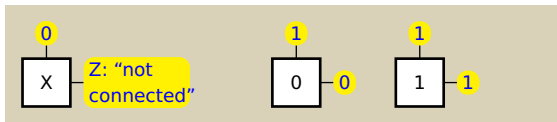
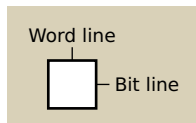
Conceptual View of Memory

Memory is an array of *cells*.

Each cell stores a single bit.

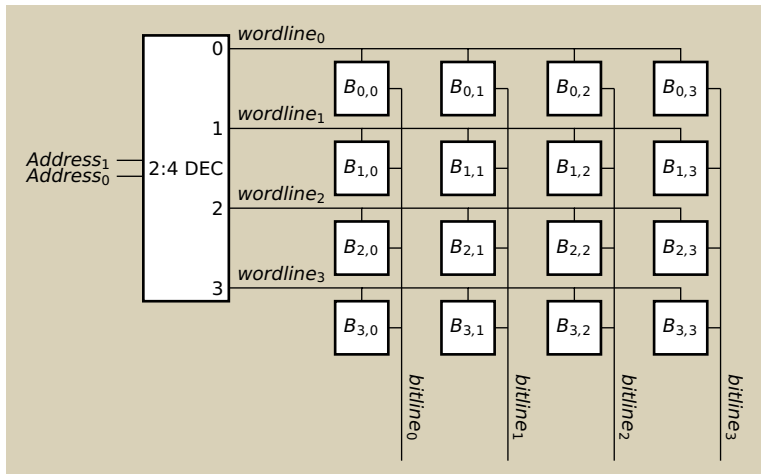


Cell Behavior

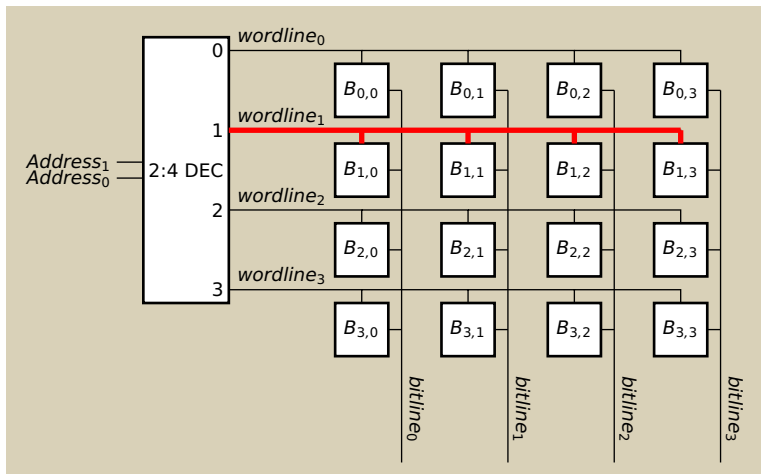


Implementation of cell depends on type of memory.

Generic Memory Array Architecture



Generic Memory Array Architecture

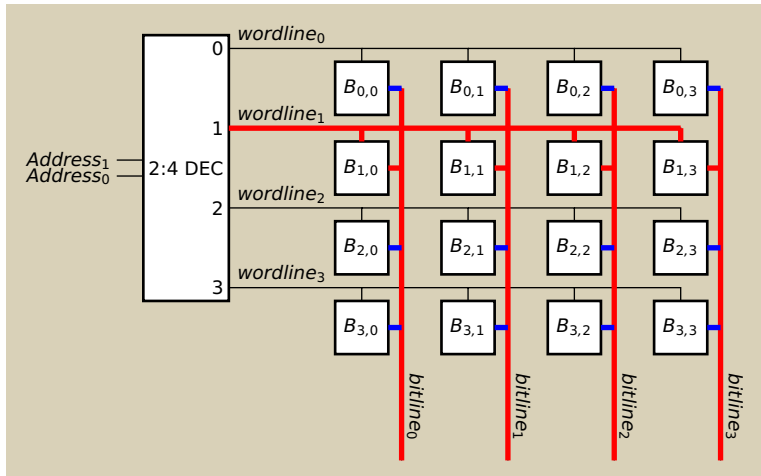


Address is decoded into set of wordlines.

Wordlines select row to be read/written.

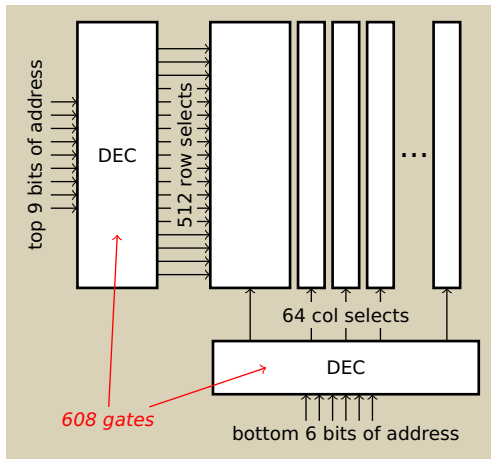
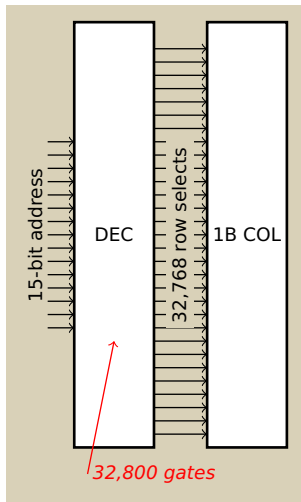
Only one wordline=1 at a time.

Generic Memory Array Architecture

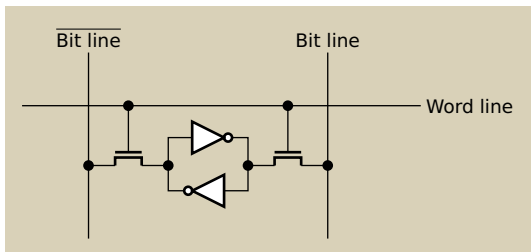


Multiple cells read in parallel, setting values of multiple bitlines.

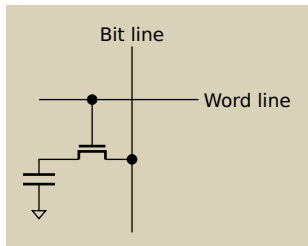
Coincident Selection Saves Decode Logic



Static Random-Access Memory Cell (SRAM)

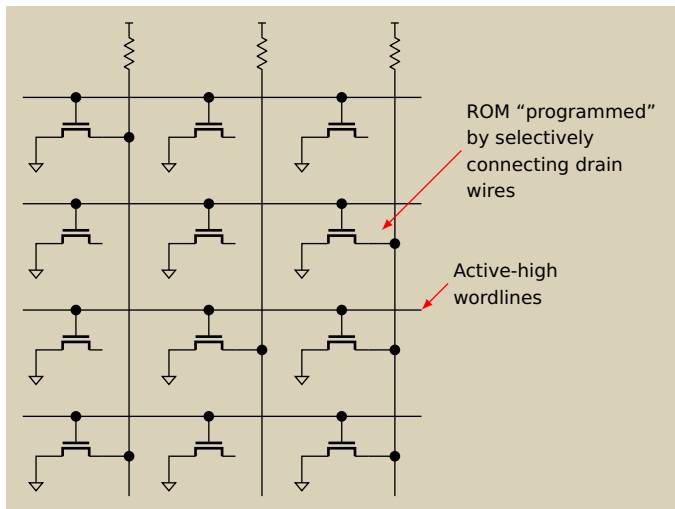


Dynamic RAM Cell

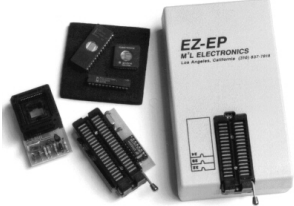
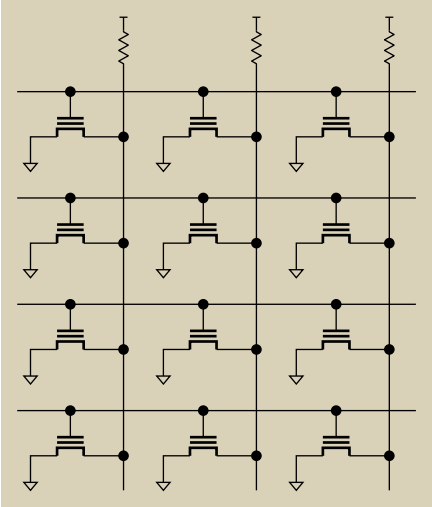


CMOS Mask-Programmed ROMs

Add. Data	
00	011
01	110
10	100
11	010



EPROMs and FLASH use Floating-Gate MOSFETs

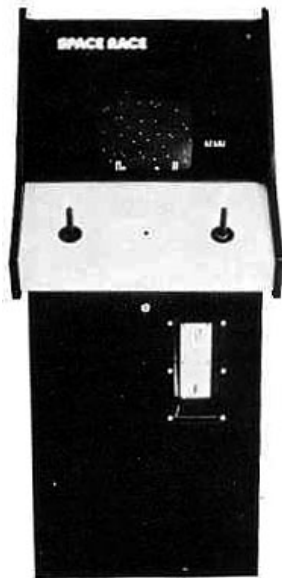


Volatile Storage Comparisons

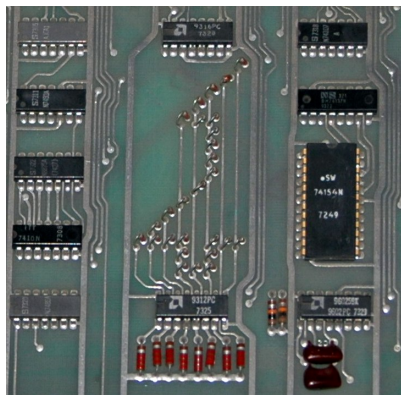
	Flip-Flop	SRAM	DRAM
Transistors/Bit Density	Approx. 20 Low	6 Medium	1 High
Access Time	Fast	Medium	Slow
Destructive Read?	No	No	Yes ¹
Power	High	Medium	Low

¹Therefore refresh required

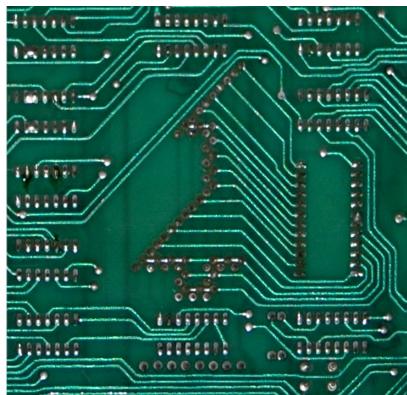
Atari Space Race, 1973



Atari Space Race PCB

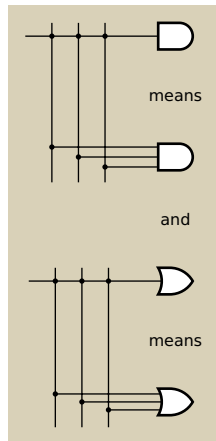
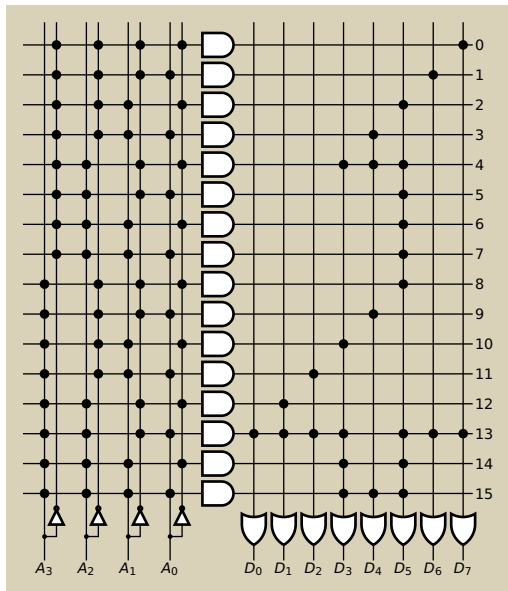


Front

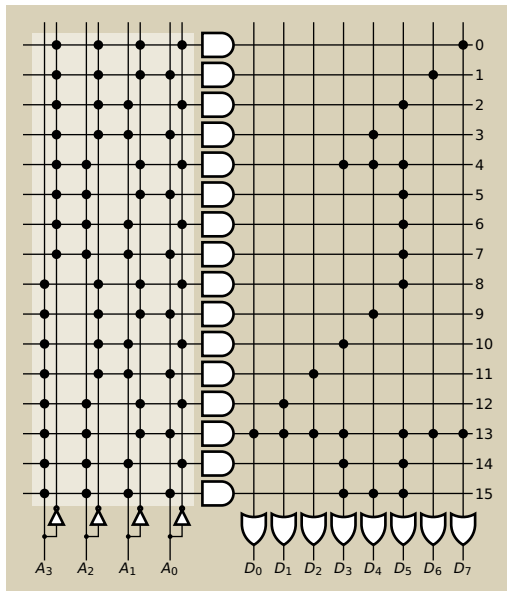


Back (mirrored)

The Space Race ROM



The Space Race ROM

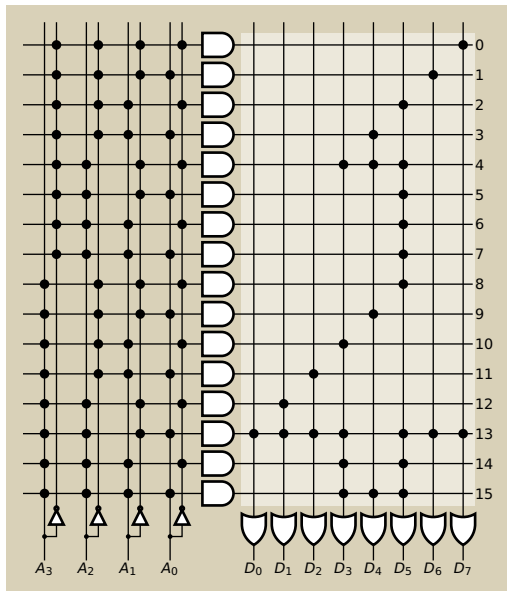


The decoder or
“AND plane”

In a RAM or ROM,
computes every
minterm

Pattern is not
programmable

The Space Race ROM

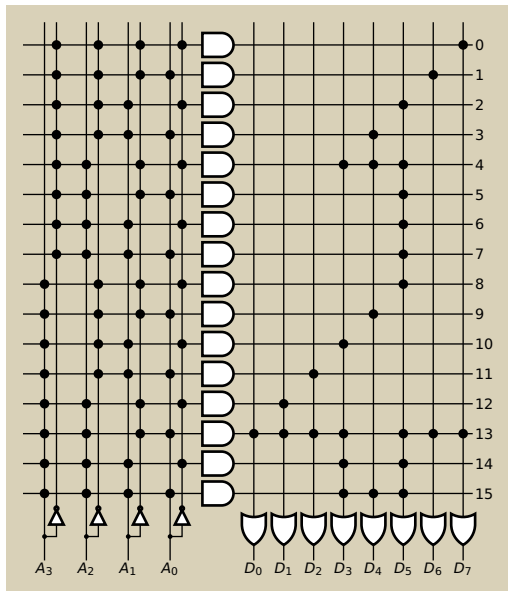


The decoder or
"OR plane"

One term for every
output

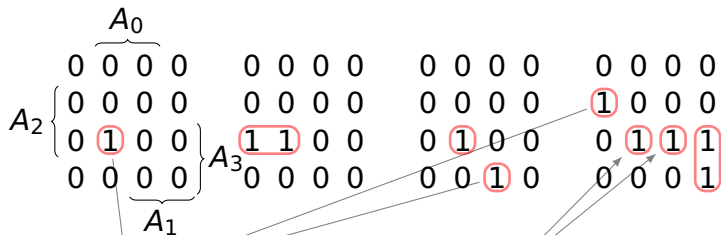
Pattern is
programmable =
the contents of the
ROM

The Space Race ROM



Can we do better?

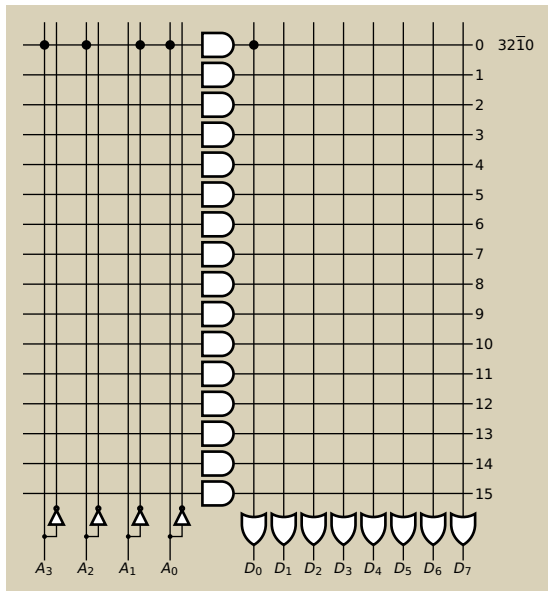
Simplifying the Space Race ROM



Essential minterms mean don't expand these

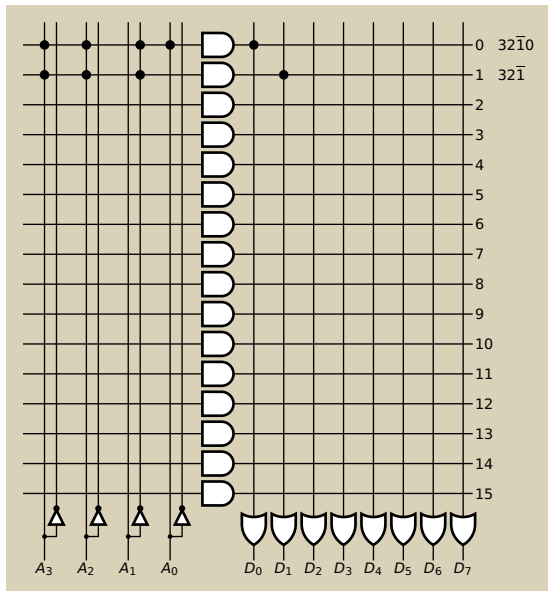


The Space Race ROM – using PAL



$$D_0 = 32\bar{1}0$$

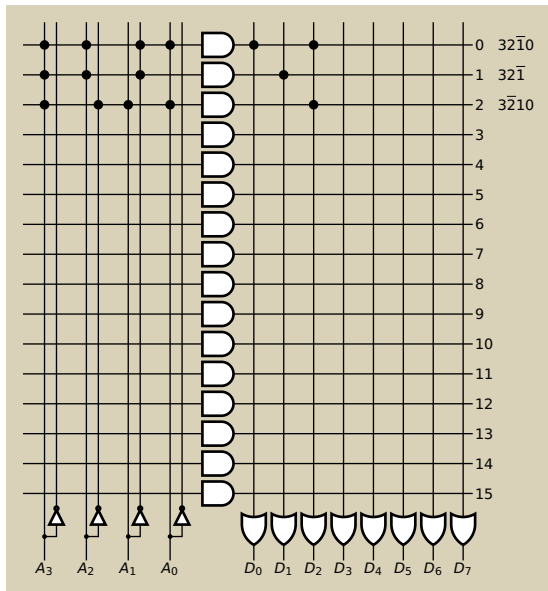
The Space Race ROM – using PAL



$$D_0 = 32\bar{1}0$$

$$D_1 = 32\bar{1}$$

The Space Race ROM – using PAL

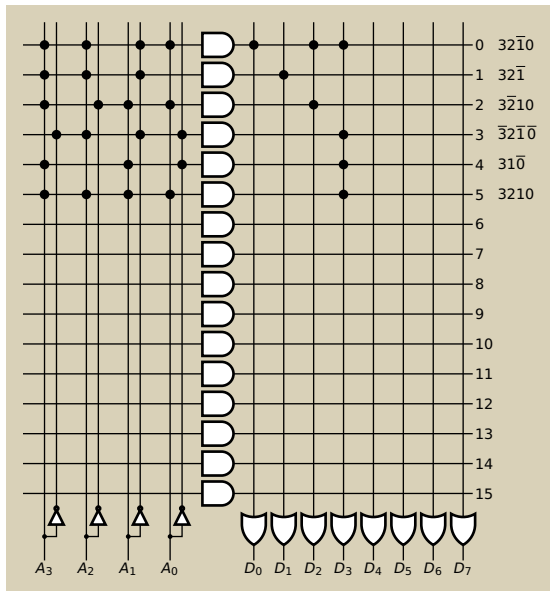


$$D_0 = 32\bar{1}0$$

$$D_1 = 32\bar{1}$$

$$D_2 = 3\bar{2}10 + 32\bar{1}0$$

The Space Race ROM – using PAL



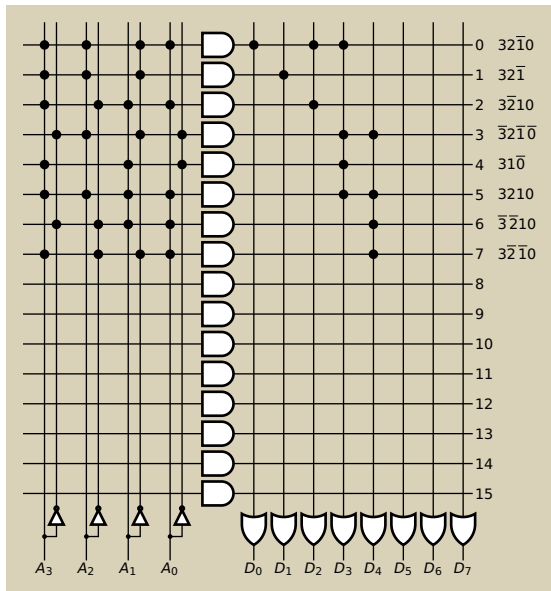
$$D_0 = 32\bar{1}0$$

$$D_1 = 32\bar{1}$$

$$D_2 = 3\bar{2}10 + 32\bar{1}0$$

$$D_3 = \bar{3}2\bar{1}0 + 31\bar{0} + 32\bar{1}0 + 3210$$

The Space Race ROM – using PAL



$$D_0 = 32\bar{1}0$$

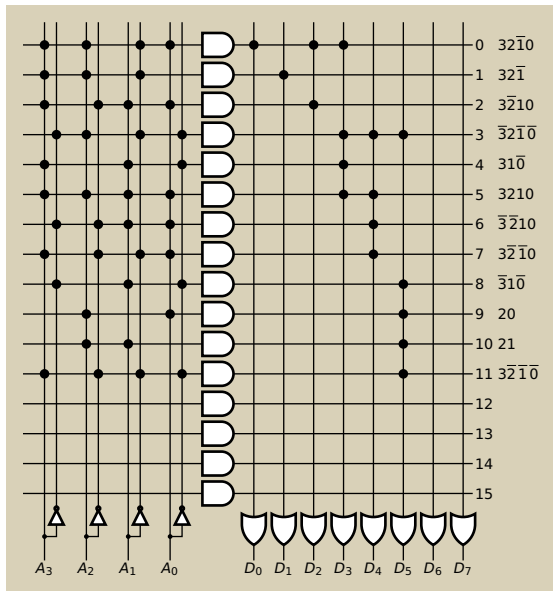
$$D_1 = 32\bar{1}$$

$$D_2 = 3\bar{2}10 + 32\bar{1}0$$

$$D_3 = \bar{3}\bar{2}\bar{1}0 + 31\bar{0} + 32\bar{1}0 + 3210$$

$$D_4 = \bar{3}\bar{2}10 + \bar{3}\bar{2}\bar{1}0 + 3\bar{2}\bar{1}0 + 3210$$

The Space Race ROM – using PAL



$$D_0 = 32\bar{1}0$$

$$D_1 = 32\bar{1}$$

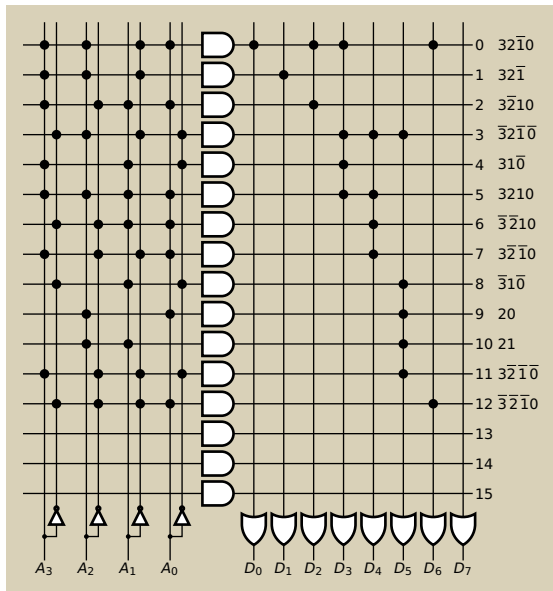
$$D_2 = 3\bar{2}10 + 32\bar{1}0$$

$$D_3 = \bar{3}2\bar{1}0 + 31\bar{0} + 32\bar{1}0 + 3210$$

$$D_4 = \bar{3}\bar{2}10 + \bar{3}2\bar{1}0 + 3\bar{2}\bar{1}0 + 3210$$

$$D_5 = \bar{3}1\bar{0} + 20 + 21 + \bar{3}2\bar{1}0 + 3\bar{2}\bar{1}0$$

The Space Race ROM – using PAL



$$D_0 = 32\bar{1}0$$

$$D_1 = 32\bar{1}$$

$$D_2 = \bar{3}\bar{2}10 + 32\bar{1}0$$

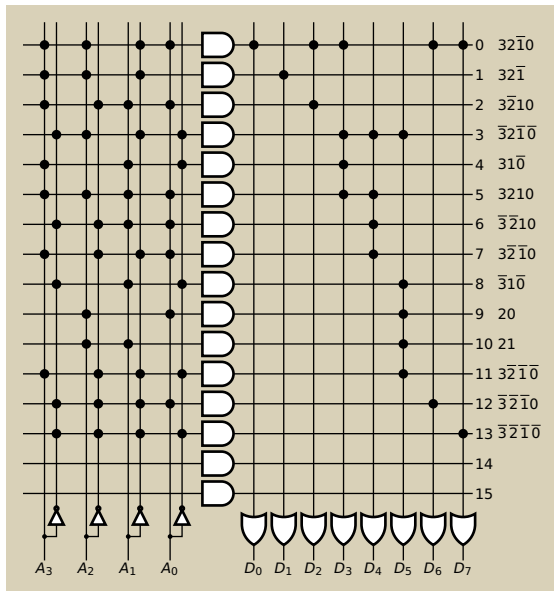
$$D_3 = \bar{3}\bar{2}\bar{1}\bar{0} + 31\bar{0} + 32\bar{1}0 + 3210$$

$$D_4 = \bar{3}\bar{2}10 + \bar{3}\bar{2}\bar{1}\bar{0} + 32\bar{1}0 + 3210$$

$$D_5 = \bar{3}1\bar{0} + 20 + 21 + \bar{3}\bar{2}\bar{1}\bar{0} + \bar{3}\bar{2}\bar{1}0$$

$$D_6 = \bar{3}\bar{2}\bar{1}0 + 32\bar{1}0$$

The Space Race ROM – using PAL



$$D_0 = 32\bar{1}0$$

$$D_1 = 32\bar{1}$$

$$D_2 = 3\bar{2}10 + 32\bar{1}0$$

$$D_3 = \bar{3}\bar{2}\bar{1}0 + 31\bar{0} + 3\bar{2}\bar{1}0 + 3210$$

$$D_4 = \bar{3}\bar{2}10 + \bar{3}\bar{2}\bar{1}0 + 3\bar{2}\bar{1}0 + 3210$$

$$D_5 = \bar{3}1\bar{0} + 20 + 21 + \bar{3}\bar{2}\bar{1}0 + 3\bar{2}\bar{1}0$$

$$D_6 = \bar{3}\bar{2}\bar{1}0 + 32\bar{1}0$$

$$D_7 = \bar{3}\bar{2}\bar{1}0 + 32\bar{1}0$$

Saved two ANDs

Field-Programmable Gate Arrays (FPGAs)

