

# Lecture 15

## EE 421 / CS 425

# Digital System Design

Fall 2024

Shahid Masud

# Topics

Midterm next  
Monday 28  
October

Booth Encoding and Booth Multiplication - Recap

Modified Booth / Radix 4 Conversion

Booth and Radix 4 Multiplication Process

Booth and Radix 4 Multiplication Examples

STG for Booth and Radix 4 Sequential Multipliers

# Booth Recoding of $-65_{10}$

$-65_{10} =$



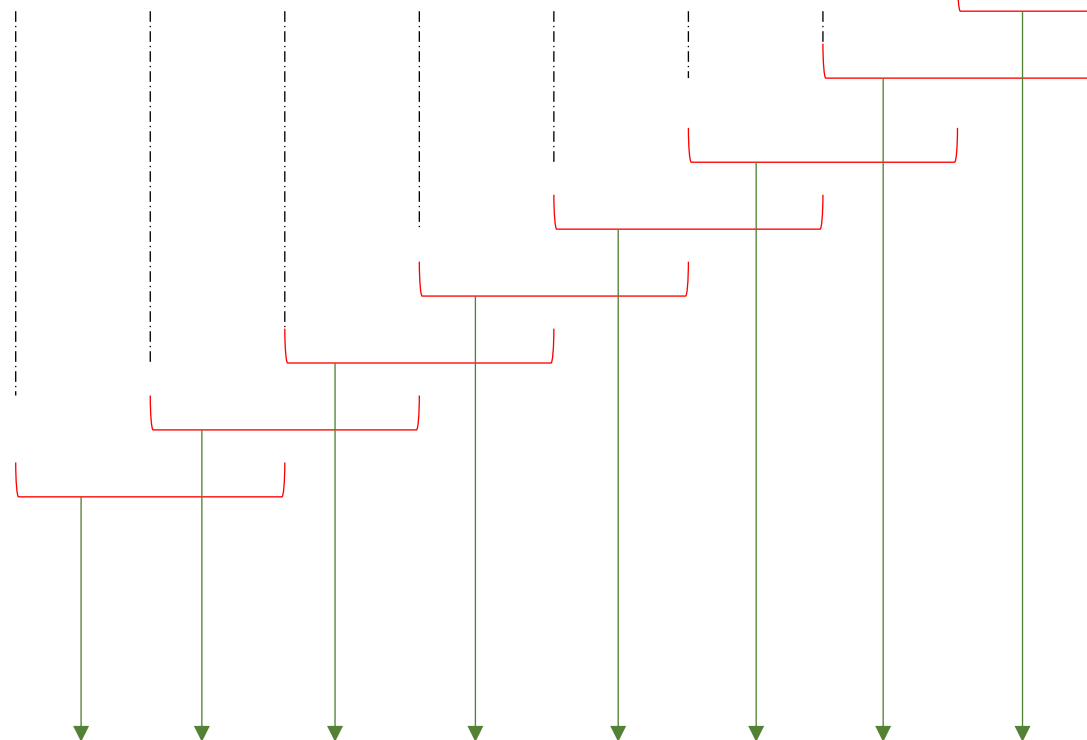
Append '0' on right, if LSB=1

2's Complement notation

$+65 = (01000001)$

2's Complement

$-65 = (10111111)$



$m_i$	$m_{i-1}$	Booth Recoded $C_i$
0	0	0
0	1	1
1	0	<u>1</u>
1	1	0

$-65_{10} =$



Or

Booth Recoded notation

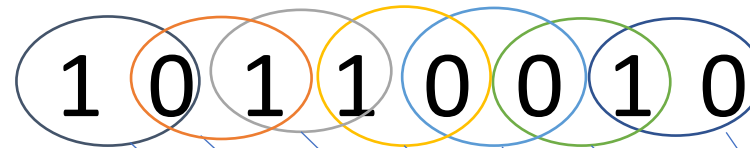
# Question?

Convert decimal number -78 to Booth Encoded format using 8 binary bits

+78 = 01001110

Take 2's Complement

-78 = 10110010



No need for extra '0' after LSB in this case

10 → -1

01 → 1

00 → 0

10 → -1

11 → 0

01 → 1

10 → -1

**Answer = -1 1 0 -1 0 1 -1**

After Booth Encoding

$m_i$	$m_{i-1}$	Booth Recoded $C_i$
0	0	0
0	1	1
1	0	-1
1	1	0

# Booth Multiplication – Example 1

Show Booth Encoded multiplication of 6 x 5, using 4 bits for both numbers

$m_i$	$m_{i-1}$	Booth Recoded $C_i$	Multiplication Use
0	0	0	Only shift
0	1	1	Add, shift
1	0	<u>1</u>	Sub, shift
1	1	0	Only shift

6 Multiplicand  
x5 Multiplier

<b>Extra bits</b>	0	0	0	0	0	1	1	0
	0	0	0	0	0	1	0	1
	1	1	1	1	1	0	1	0
0	0	0	0	0	1	1	0	X
1	1	1	1	1	0	1	0	X
0	0	0	0	1	1	0	X	X
					X	X	X	X
0	1	0	0	0	1	1	1	0

Imagine Zero bit if LSB = 1

Check 2-bits at a time, Right to Left  
Shift Left by 1 after every step

1[0] = Subtract = Add 2's Compl of Multiplicand to Acc

01 = Add Multiplicand to Acc

10 = Subtract = Add 2's Compl of Multiplicand to Acc

01 = Add Multiplicand to Acc

00 = No Op, Shift Left by 1

00 = No Op, Shift Left by 1

Answer = (0001 1110) = +(16 + 14) = +30<sub>10</sub>

# Booth Multiplication – Example 2

Show Booth Encoded multiplication of 6 x -5, using 4 bits for both numbers

6 Multiplicand  
X -5 Multiplier

		0	0	0	0	0	1	1	0
		1	1	1	1	1	0	1	1
		1	1	1	1	1	0	1	0
									X
0	0	0	0	0	1	1	0	X	X
1	1	1	1	0	1	0	X	X	X
						X	X	X	X
0	0	1	1	1	0	0	0	1	0

Imagine Zero bit if LSB = 1

Check 2-bits at a time, Right to Left  
Shift Left by 1 after every step

1[0] = Subtract = Add 2's Compl of Multiplicand to Acc

11 = No Op, Shift Left, Add 0 to Acc

01 = Add Multiplicand to Acc

10 = Subtract = Add 2's Compl of Multiplicand to Acc

11 = No Op, Shift Left by 1, Add 0 to Acc

Answer = (1110 0010) = Take 2's Comp = -( 0001 1110) = -30<sub>10</sub>

# Booth Multiplication – Example 3

Show Booth Encoded multiplication of B3 x C3, using 8 bits for both numbers

B3 = 1011 0011 = 2's Compl of = 0100 1101 =  $(4D)_{16} = 77_{10}$

C3 = 1100 0011 = 2's Compl of = 0011 1101 =  $(3D) = 61_{10}$

B3	Multiplicand								1	0	1	1	0	0	1	1
X C3	Multiplier								1	1	0	0	0	0	1	1
	0	0	0	0	0	0	0	0	0	1	0	0	1	1	0	1
																X
	1	1	1	1	1	1	1	0	1	1	0	0	1	1	X	X
												X	X	X	X	X
	0	0	0	1	0	0	1	1	0	1	X	X	X	X	X	X
1	0	0	0	1	0	0	1	0	0	1	0	1	1	0	0	1

Imaginary Zero bit if LSB = 1

Check 2-bits at a time, Right to Left  
Shift Left by 1 after every step

1[0] = Subtract = Add 2's Compl of Multiplicand to Acc

11 = No Op, Shift Left, Add 0 to Acc

01 = Add Multiplicand to Acc

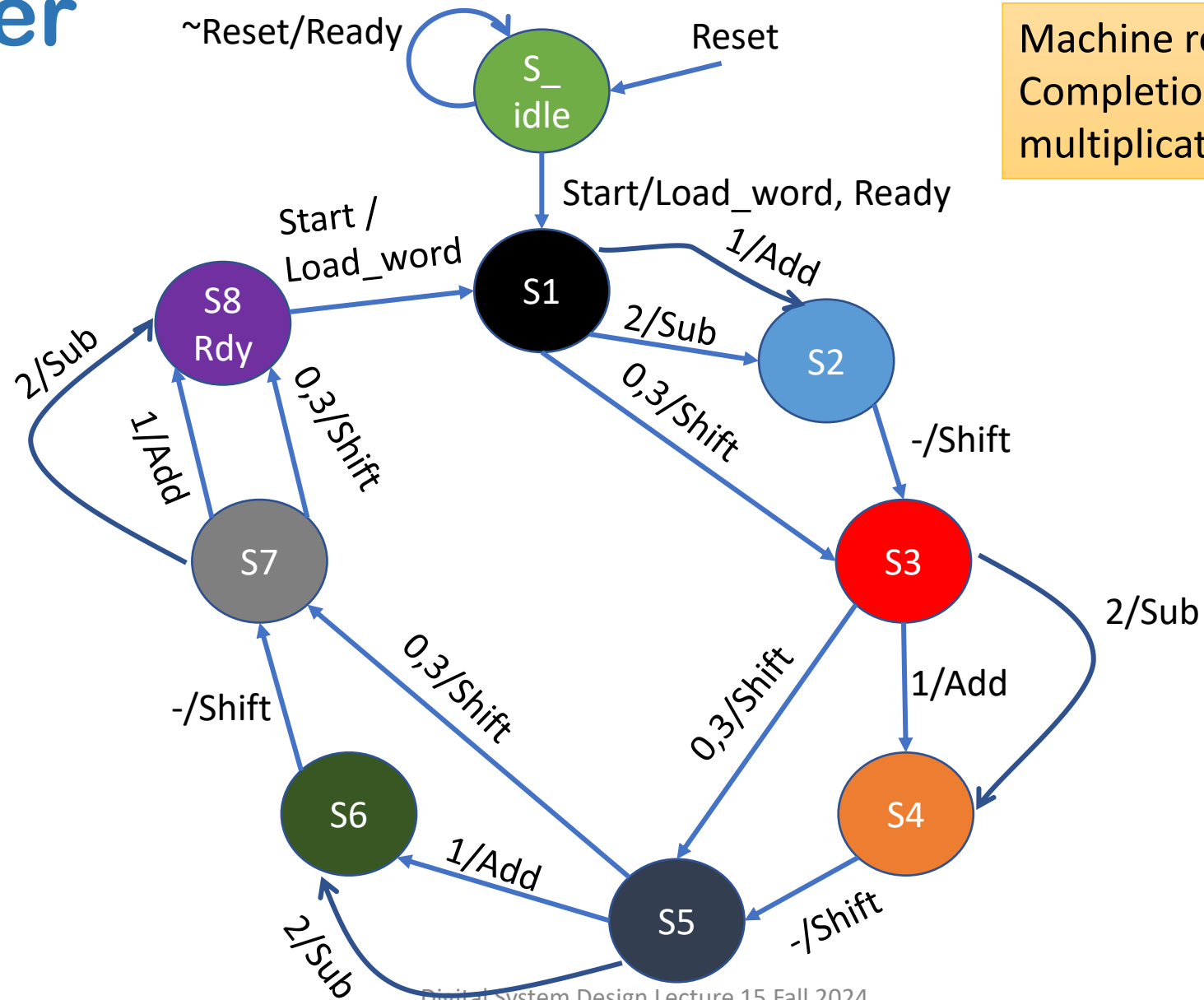
00, 00, 00 = No Op, Shift Left by 1

10 = Subtract = Add 2's Compl of Multiplicand to Acc

11 = No Op, Shift Left, Add 0 to Acc

Answer =  $(0001\ 0010\ 0101\ 1001)_2 = (1259)_{Hex} = (1 \times 16^3 + 2 \times 16^2 + 5 \times 16^1 + 9 \times 16^0) = 4697_{10}$

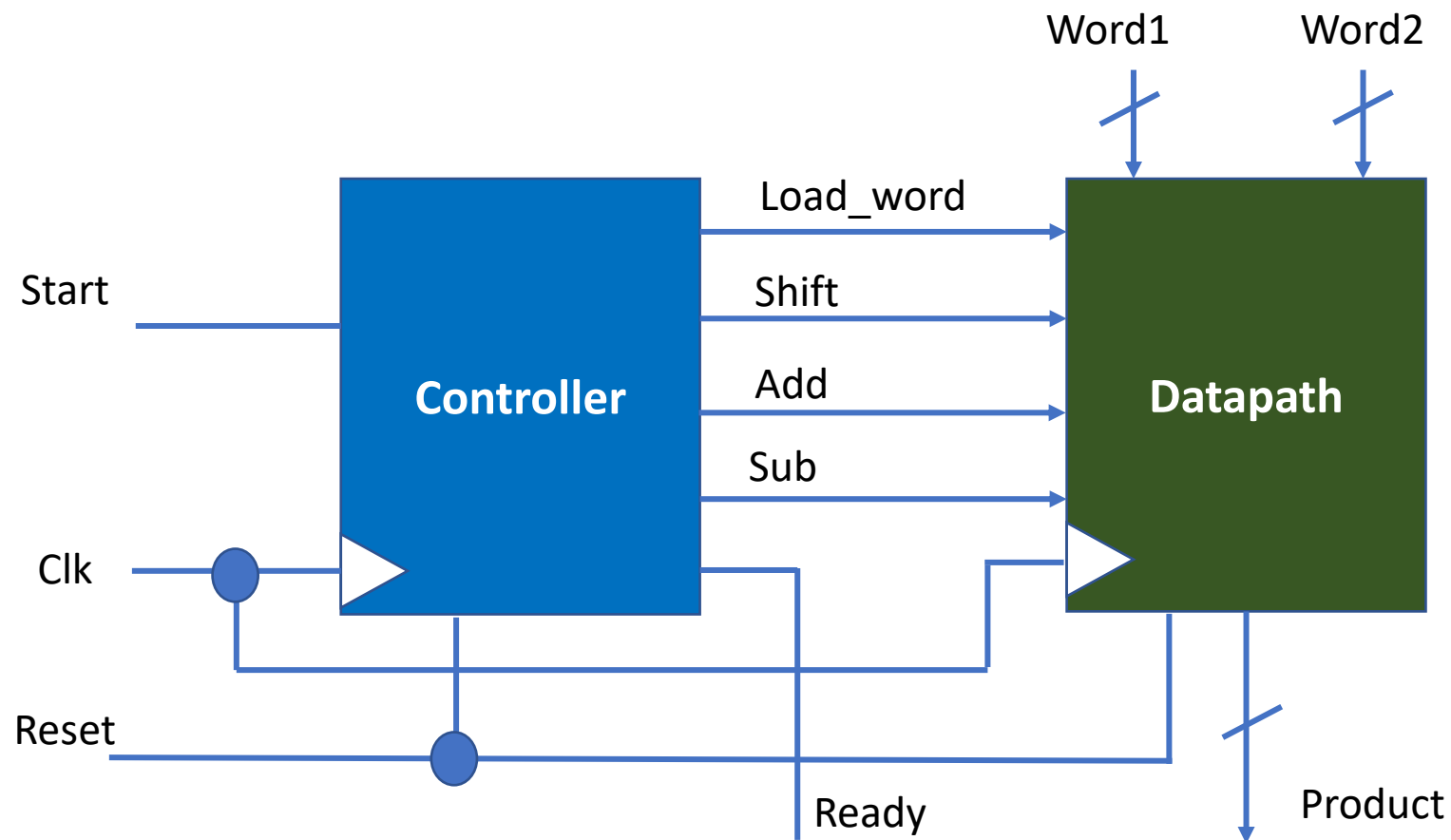
# STG for a 4 Bit Booth Encoded Sequential Multiplier



Machine returns to Idle state after Completion of 4 bit Booth Encoded multiplication



# Data Path Architecture of a Booth Sequential Multiplier



# Question?

Perform the following multiplication using Booth Encoding.

Multiplicand = 35, Multiplier = 19

How many Adds and Shifts are required in this multiplication?

How does this compare to a simple binary array multiplier?

# Bit-Pair Encoding

## Modified Booth Encoding

## Radix-4 Encoding

$m_{i+1}$	$m_i$	$m_{i-1}$	Code	$BRC_{i+1}$	$BRC_i$	Value	Status	Multiply Actions
0	0	0	0	0	0	0	String of 0s	Shift by 2
0	0	1	1	0	1	+1	End of string of 1s	Add
0	1	0	2	0	1	+1	Single 1	Add
0	1	1	3	1	0	+2	End of string of 1s	Shift by 1, Add, Shift by 1
1	0	0	4	<u>1</u>	0	-2	Begin of string of 1s	Shift by 1, Subtract, Shift by 1
1	0	1	5	0	<u>1</u>	-1	Single 0	Subtract
1	1	0	6	0	<u>1</u>	-1	Begin of string of 1s	Subtract
1	1	1	7	0	0	0	Midstring of 1s	Shift by 2

# Bit-Pair / Radix-4 Recoding of $-65_{10}$

$-65_{10} =$



Imaginary '0' if LSB=1

2's Complement notation

$+65 = (01000001)$

2's Complement

$-65 = (10111111)$

$-65_{10} =$



$m_{i+1}$	$m_i$	$m_{i-1}$	$BRC_{i+1}$	$BRC_i$	Value
0	0	0	0	0	0
0	0	1	0	1	+1
0	1	0	0	1	+1
0	1	1	1	0	+2
1	0	0	1	0	-2
1	0	1	0	1	-1
1	1	0	0	1	-1
1	1	1	0	0	0

Bit-Pair Recoded notation

# Question of Bit-Pair/Radix-4 Encoding

Express  $-75_{10}$  in Radix-4 Encoded format using 8 bits to express the given number

$m_{i+1}$	$m_i$	$m_{i-1}$	$BRC_{i+1}$	$BRC_i$	Value
0	0	0	0	0	0
0	0	1	0	1	+1
0	1	0	0	1	+1
0	1	1	1	0	+2
1	0	0	1	0	-2
1	0	1	0	1	-1
1	1	0	0	1	-1
1	1	1	0	0	0

$$+75_{10} = (64+8+2+1) = (0100\ 1011)_2$$

Thus 2's Complement

$$= (1011\ 0101)_2 = -75$$

1 0 1 1 0 1 0 1[0]

2; coded 01

2; coded 01

6; coded 0 -1

5; coded 0 -1

Radix 4 Encoded = 0 -1 0 -1 0 1 0 1

Radix 4 Encoded = 0 1 0 1 0 1

# Bit-Pair Encoding

## Modified Booth Encoding

## Radix-4 Encoding

Shifting by 2 in each step

$m_{i+1}$	$m_i$	$m_{i-1}$	Code	$BRC_{i+1}$	$BRC_i$	Value	Status	Multiply Actions
0	0	0	0	0	0	0	String of 0s	Shift Left by 2
0	0	1	1	0	1	+1	End of string of 1s	Add, Shift Left by 2
0	1	0	2	0	1	+1	Single 1	Add, Shift Left by 2
0	1	1	3	1	0	+2	End of string of 1s	Shift by 1, Add, Shift by 1
1	0	0	4	1	0	-2	Begin of string of 1s	Shift by 1, Subtract, Shift by 1
1	0	1	5	0	1	-1	Single 0	Subtract, Shift Left by 2
1	1	0	6	0	1	-1	Begin of string of 1s	Subtract, Shift Left by 2
1	1	1	7	0	0	0	Mid-string of 1s	Shift Left by 2

# Bit-Pair / Radix-4 Recoding of $-65_{10}$

$-65_{10} =$



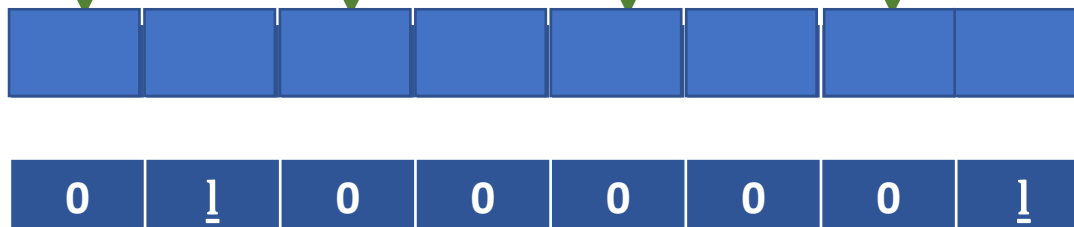
Imaginary '0' if LSB=1  
2's Complement notation

$+65 = (01000001)$

2's Complement

$-65 = (10111111)$

$-65_{10}$  RECODED =



$m_{i+1}$	$m_i$	$m_{i-1}$	$BRC_{i+1}$	$BRC_i$	Value
0	0	0	0	0	0
0	0	1	0	1	+1
0	1	0	0	1	+1
0	1	1	1	0	+2
1	0	0	1	0	-2
1	0	1	0	1	-1
1	1	0	0	1	-1
1	1	1	0	0	0

Bit-Pair Recoded notation

# Question of Bit-Pair/Radix-4 Encoding

Express  $-75_{10}$  in Radix-4 Encoded format using 8 bits to express the given number

$m_{i+1}$	$m_i$	$m_{i-1}$	$BRC_{i+1}$	$BRC_i$	Value
0	0	0	0	0	0
0	0	1	0	1	+1
0	1	0	0	1	+1
0	1	1	1	0	+2
1	0	0	1	0	-2
1	0	1	0	1	-1
1	1	0	0	1	-1
1	1	1	0	0	0

$$+75_{10} = (64+8+2+1) = (0100\ 1011)_2$$

Thus 2's Complement

$$= (1011\ 0101)_2 = -75$$

1 0 1 1 0 1 0 1[0]

2; coded 01

2; coded 01

6; coded 0 -1

5; coded 0 -1

Radix 4 Encoded = 0 -1 0 -1 0 1 0 1

Radix 4 Encoded = 0 1 0 1 0 1



# Radix 4 Coding for Multiplication

$m_{i+1}$	$m_i$	$m_{i-1}$	Code	Multiply Actions
0	0	0	0	Shift Left by 2
0	0	1	1	Add Multiplicand, Shift Left by 2
0	1	0	2	Add Multiplicand, Shift Left by 2
0	1	1	3	Shift by 1, Add Multiplicand, Shift by 1
1	0	0	4	Shift by 1, Subtract Multiplicand, Shift by 1
1	0	1	5	Subtract Multiplicand, Shift Left by 2
1	1	0	6	Subtract Multiplicand, Shift Left by 2
1	1	1	7	Shift Left by 2

# Radix 4 Multiplication – Example 1

Imagine Zero bit if LSB = 1

Show Radix 4 Encoded multiplication of 8 x 9, using 8 bits for both numbers

8 = 0000 1000

9 = 0000 1001

Convert 9 = 0000 1001 to Radix 4 Encoded bits

9 = 0 0 0 0 1 0 0 1 [0]

**RECODED**  
 010 → 01  
 100 → -1 0  
 001 → 01  
 000 → 00

8 = Multiplicand

X 9 = Recoded Multiplier

													0	0	0	0	1	0	0	0
													0	0	0	1	-1	0	0	1
	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
1	1	1	1	1	1	1	1	1	1	1	0	0	0	X	X	X				
0	0	0	0	0	0	0	0	0	0	1	0	0	0	X	X	X	X	X		
													X	X	X	X	X			
0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0			

0 1 = Add Multiplicand, Shl2

-1 0 = Shl 1, Sub, Shl1

0 1 = Add, Shl2

0 0 = Only Shl2, No op

Answer = (0100 1000) = +(64 + 8) = +72<sub>10</sub>

# Radix 4 Multiplication – Example 2

Show Radix 4 Encoded multiplication of **68 x -19**, using 8 bits for both numbers

68 = 0100 0100  
And 2's Compl is  
-68 = 1011 1100

19 = 0001 0011  
And 2's Compl is  
-19 = 1110 1101

Convert -19 = 1110 1101 to Radix 4 Encoded bits

-19 = 1 1 1 0 1 1 0 1 [0]

Imagine Zero

**RECODED**  
010 → 01  
110 → 0-1  
110 → 0-1  
111 → 00

68 = Multiplicand

X -19 = **Recoded** Multiplier

Result

										0	1	0	0	0	1	0	0
										0	0	0	-1	0	-1	0	1
	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0
1	1	1	1	1	1	1	1	0	1	1	1	1	0	0	X	X	
1	1	1	1	1	1	0	1	1	1	1	0	0	X	X	X	X	
													X	X	X	X	
1	1	1	1	1	1	0	1	0	1	1	1	1	0	1	0	0	

0 1 = Add Multiplicand, Shl2

0 -1 = Sub, Shl2

0 -1 = Sub, Shl2

0 0 = Only Shl2, No op

Take 2's Complement of Result = -(0101 0000 1100) = -(50C) Hex = -(1292)<sub>10</sub>

# Radix 4 Multiplication – Example 3

Show Radix 4 Encoded multiplication of **76 x 55**, using 8 bits for both numbers

76 = 0100 1100  
And 2's Compl is  
-76 = 1011 0100

55 = 0011 0111  
And 2's Compl is  
-55 = 1100 1001

Convert 55 = 0011 0111 to Radix 4 Encoded bits

Imagine Zero  
55 = 0 0 1 1 0 1 1 1 [0]

**RECODED**  
110 → 0-1  
011 → 10  
110 → 0-1  
001 → 01

76 = Multiplicand

X 55 = Recoded Multiplier

Partial Sum

Partial Sum

Result

										0	1	0	0	1	1	0	0
										0	1	0	-1	1	0	0	-1
	1	1	1	1	1	1	1	1	1	0	1	1	0	1	0	0	
	0	0	0	0	0	0	0	1	0	0	1	1	0	0	X	X	X
Partial Sum	0	0	0	0	0	0	0	1	0	0	0	0	1	0	1	0	0
	1	1	1	1	1	1	0	1	1	0	1	0	0	X	X	X	X
Partial Sum	1	1	1	1	1	1	1	0	1	0	1	0	1	0	1	0	0
	0	0	0	0	1	0	0	1	1	0	0	X	X	X	X	X	X
Result	0	0	0	0	1	0	0	0	0	0	1	0	1	0	1	0	0

0 -1 = Sub, Shl2

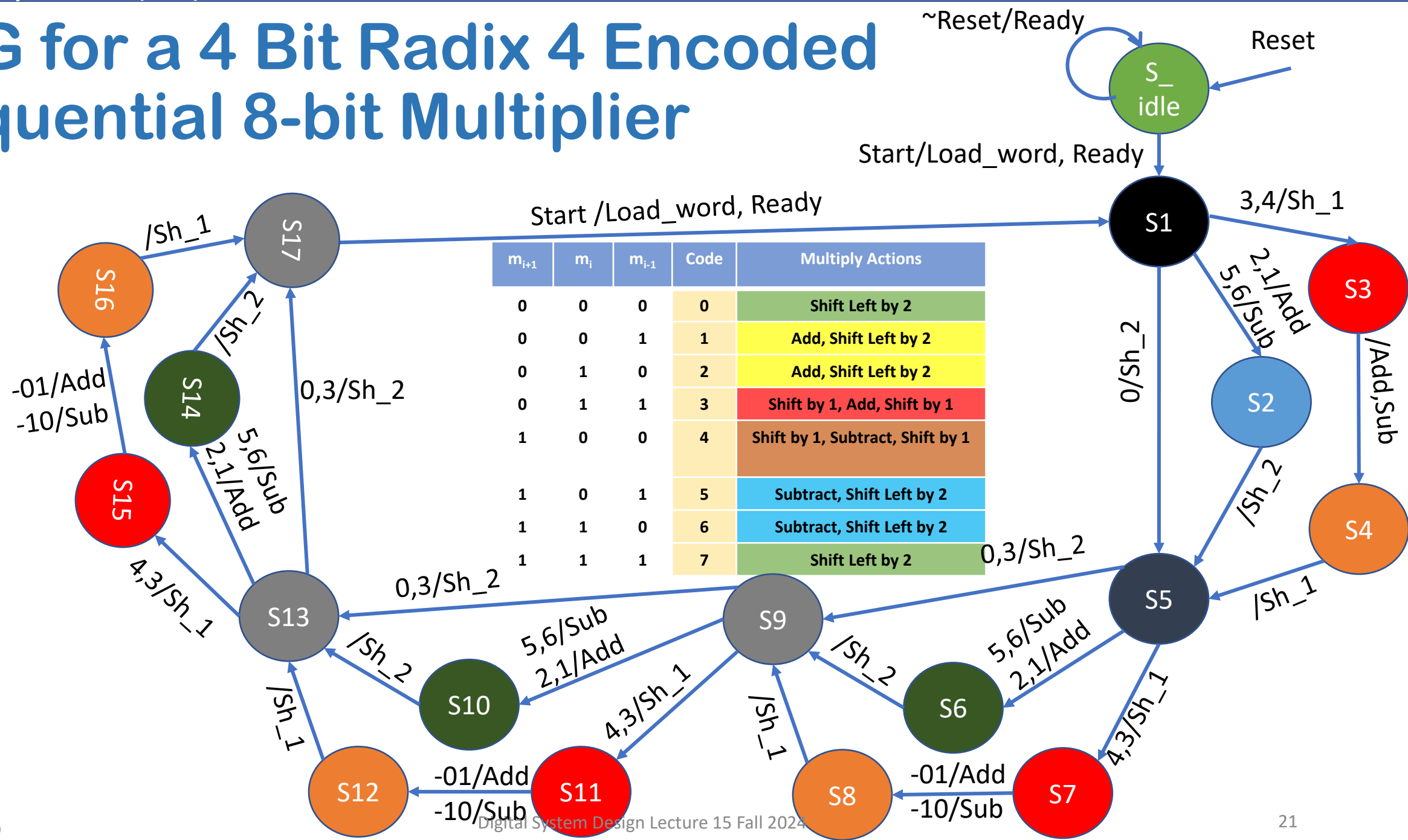
1 0 = Shl1, Add, Shl1

0 -1 = Sub, Shl2

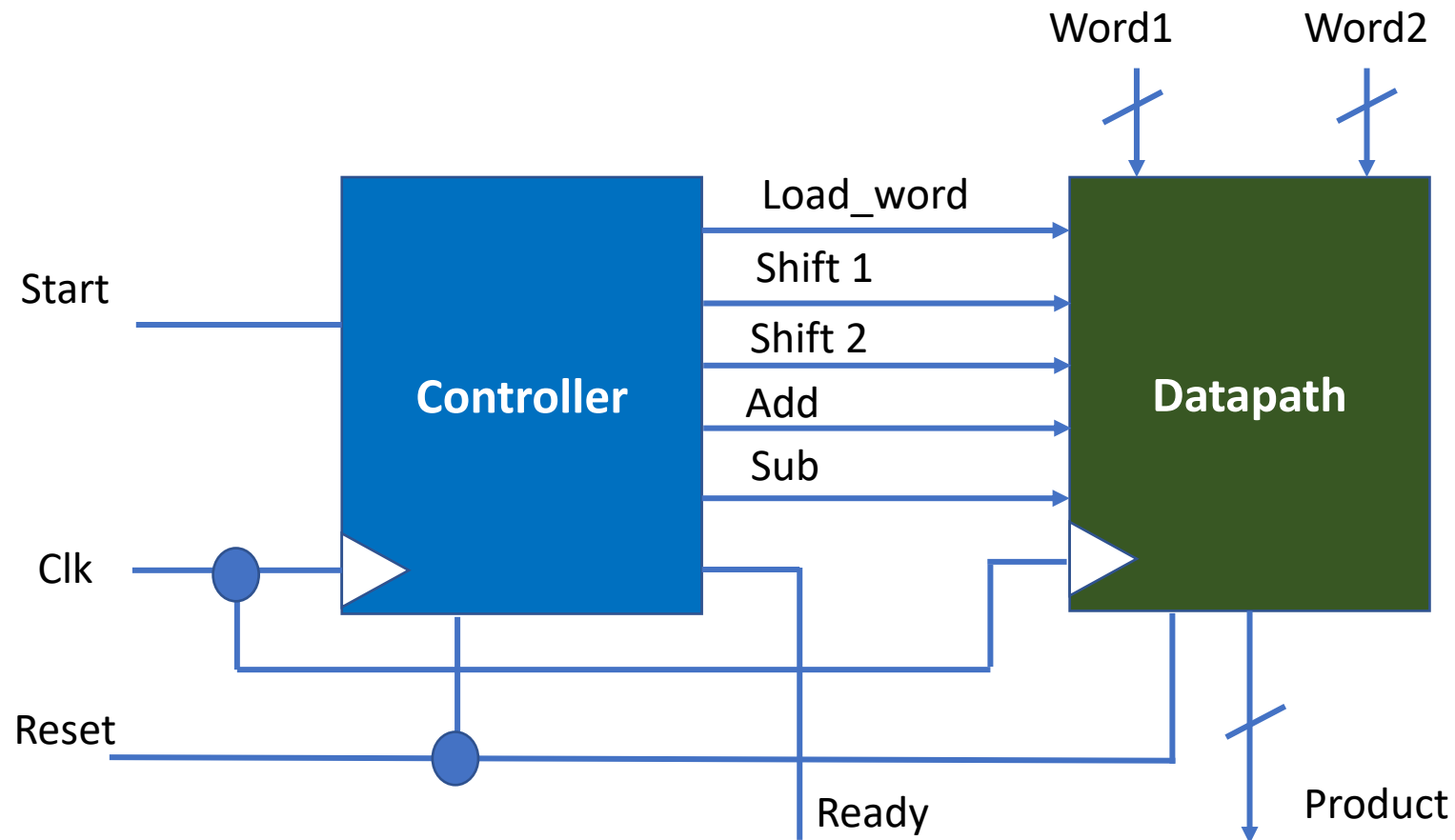
0 1 = Add, Shl2

**Answer = 0001 0000 0101 0100 = (4+16+64+4096) = (4180)<sub>10</sub>**

# STG for a 4 Bit Radix 4 Encoded Sequential 8-bit Multiplier



# Data Path Architecture of a Radix 4 Sequential Multiplier



# Question?

Perform the following multiplication using Radix 4 Encoding.

Multiplicand = 38, Multiplier = 23 (bits allocated?)

How many Adds and Shifts are required in this multiplication?

How does this compare to a simple binary array multiplier?

# Online Booth Encoding Simulator

<http://www.ecs.umass.edu/ece/koren/arith/simulator/Booth/>

Prof. Korean Page:

<http://www.ecs.umass.edu/ece/koren/>

Simulators Tab is on the left

Many simulators of Computer Arithmetic are available:

<http://www.ecs.umass.edu/ece/koren/arith/simulator/>

**Professor Emeritus Israel Koren**  
UMass Dept. of Electrical and Computer Engineering

**Biography**  
Israel Koren received the B.Sc., M.Sc., and D.Sc. degrees from the Technion - Israel Institute of Technology, Haifa, in 1967, 1970, and 1975, respectively, all in Electrical Engineering. He is currently a Professor Emeritus of Electrical and Computer Engineering at the University of Massachusetts, Amherst. He was a visiting Professor at the Instituto de Informatica, Universidade Federal do Rio Grande do Sul (UFGRS), Brazil and at Bar-Ilan University, Faculty of Engineering, Ramat-Gan, Israel. He is an Honorary Professor at Amity University, Uttar Pradesh, India. Previously he held positions with the Technion - Israel Institute of Technology, Haifa, the University of California at Berkeley, the University of Southern California, Los Angeles and the University of California, Santa Barbara. He has been a consultant to several companies including Analog Devices, AMD, Digital Equipment Corp., IBM, Intel, National Semiconductor and Tolerant Systems.

Dr. Koren's current research interests are Fault-Tolerant Systems, Cyber-Physical Systems, Secure Cryptographic Devices and Computer Arithmetic. He publishes extensively and has over 300 publications in refereed journals and conferences. He is an Associate Editor of the Sustainable Computing, Informatics and Systems Journal. He has been a Co-Guest Editor for several IEEE Transactions on Computers special issues on Fault Diagnosis and Tolerance in Cryptography, Sept. 2008, on Computer Arithmetic, July 2008, and on High Yield VLSI Systems, April 1998. He served on the Editorial Board of these Transactions during 1992-1997, the Editorial Board of the IEEE Transactions on VLSI Systems during 2001-2006, the Editorial Board of the IEEE Computer Architecture Letters during 2006-2010 and the Editorial Board of the VLSI Design Journal during 2008-2010. He also served as General Chair, Program Chair and Program Committee member for numerous conferences. He has edited and co-authored the book, Defect and Fault-Tolerance in VLSI Systems, Vol. 1, Plenum, 1989. He is the author of the textbook Computer Arithmetic Algorithms, Second Edition, A. K. Peters, Natick, MA, 2002. He is a co-author of the textbook Fault-Tolerant Systems, 2nd Edition, Morgan Kaufman, San Francisco, CA, 2002. Dr. Koren is a Fellow of the IEEE and a Member of the Dr. Tan Yee Yee Academy, China.

**Research Interests**  
Research is being conducted in the following directions (the pointers in brackets indicate the relevant publications in the list below or follow the link to Prior Publications):  
Multicore and SoC Architectures: [SARAK11], [DANK17], [RKK116], [RKK119], [RKK121], [RKK123], [SARAK116], [SARAK124], [RKK126], [RKK127], [RKK128], [RKK129], [RKK130], [RKK131], [RKK132], [RKK133], [RKK134], [RKK135], [RKK136], [RKK137], [RKK138], [RKK139], [RKK140], [RKK141], [RKK142], [RKK143], [RKK144], [RKK145], [RKK146], [RKK147], [RKK148], [RKK149], [RKK150], [RKK151], [RKK152], [RKK153], [RKK154], [RKK155], [RKK156], [RKK157], [RKK158], [RKK159], [RKK160], [RKK161], [RKK162], [RKK163], [RKK164], [RKK165], [RKK166], [RKK167], [RKK168], [RKK169], [RKK170], [RKK171], [RKK172], [RKK173], [RKK174], [RKK175], [RKK176], [RKK177], [RKK178], [RKK179], [RKK180], [RKK181], [RKK182], [RKK183], [RKK184], [RKK185], [RKK186], [RKK187], [RKK188], [RKK189], [RKK190], [RKK191], [RKK192], [RKK193], [RKK194], [RKK195], [RKK196], [RKK197], [RKK198], [RKK199], [RKK200], [RKK201], [RKK202], [RKK203], [RKK204], [RKK205], [RKK206], [RKK207], [RKK208], [RKK209], [RKK210], [RKK211], [RKK212], [RKK213], [RKK214], [RKK215], [RKK216], [RKK217], [RKK218], [RKK219], [RKK220], [RKK221], [RKK222], [RKK223], [RKK224], [RKK225], [RKK226], [RKK227], [RKK228], [RKK229], [RKK230], [RKK231], [RKK232], [RKK233], [RKK234], [RKK235], [RKK236], [RKK237], [RKK238], [RKK239], [RKK240], [RKK241], [RKK242], [RKK243], [RKK244], [RKK245], [RKK246], [RKK247], [RKK248], [RKK249], [RKK250], [RKK251], [RKK252], [RKK253], [RKK254], [RKK255], [RKK256], [RKK257], [RKK258], [RKK259], [RKK260], [RKK261], [RKK262], [RKK263], [RKK264], [RKK265], [RKK266], [RKK267], [RKK268], [RKK269], [RKK270], [RKK271], [RKK272], [RKK273], [RKK274], [RKK275], [RKK276], [RKK277], [RKK278], [RKK279], [RKK280], [RKK281], [RKK282], [RKK283], [RKK284], [RKK285], [RKK286], [RKK287], [RKK288], [RKK289], [RKK290], [RKK291], [RKK292], [RKK293], [RKK294], [RKK295], [RKK296], [RKK297], [RKK298], [RKK299], [RKK300], [RKK301], [RKK302], [RKK303], [RKK304], [RKK305], [RKK306], [RKK307], [RKK308], [RKK309], [RKK310], [RKK311], [RKK312], [RKK313], [RKK314], [RKK315], [RKK316], [RKK317], [RKK318], [RKK319], [RKK320], [RKK321], [RKK322], [RKK323], [RKK324], [RKK325], [RKK326], [RKK327], [RKK328], [RKK329], [RKK330], [RKK331], [RKK332], [RKK333], [RKK334], [RKK335], [RKK336], [RKK337], [RKK338], [RKK339], [RKK340], [RKK341], [RKK342], [RKK343], [RKK344], [RKK345], [RKK346], [RKK347], [RKK348], [RKK349], [RKK350], [RKK351], [RKK352], [RKK353], [RKK354], [RKK355], [RKK356], [RKK357], [RKK358], [RKK359], [RKK360], [RKK361], [RKK362], [RKK363], [RKK364], [RKK365], [RKK366], [RKK367], [RKK368], [RKK369], [RKK370], [RKK371], [RKK372], [RKK373], [RKK374], [RKK375], [RKK376], [RKK377], [RKK378], [RKK379], [RKK380], [RKK381], [RKK382], [RKK383], [RKK384], [RKK385], [RKK386], [RKK387], [RKK388], [RKK389], [RKK390], [RKK391], [RKK392], [RKK393], [RKK394], [RKK395], [RKK396], [RKK397], [RKK398], [RKK399], [RKK400], [RKK401], [RKK402], [RKK403], [RKK404], [RKK405], [RKK406], [RKK407], [RKK408], [RKK409], [RKK410], [RKK411], [RKK412], [RKK413], [RKK414], [RKK415], [RKK416], [RKK417], [RKK418], [RKK419], [RKK420], [RKK421], [RKK422], [RKK423], [RKK424], [RKK425], [RKK426], [RKK427], [RKK428], [RKK429], [RKK430], [RKK431], [RKK432], [RKK433], [RKK434], [RKK435], [RKK436], [RKK437], [RKK438], [RKK439], [RKK440], [RKK441], [RKK442], [RKK443], [RKK444], [RKK445], [RKK446], [RKK447], [RKK448], [RKK449], [RKK450], [RKK451], [RKK452], [RKK453], [RKK454], [RKK455], [RKK456], [RKK457], [RKK458], [RKK459], [RKK460], [RKK461], [RKK462], [RKK463], [RKK464], [RKK465], [RKK466], [RKK467], [RKK468], [RKK469], [RKK470], [RKK471], [RKK472], [RKK473], [RKK474], [RKK475], [RKK476], [RKK477], [RKK478], [RKK479], [RKK480], [RKK481], [RKK482], [RKK483], [RKK484], [RKK485], [RKK486], [RKK487], [RKK488], [RKK489], [RKK490], [RKK491], [RKK492], [RKK493], [RKK494], [RKK495], [RKK496], [RKK497], [RKK498], [RKK499], [RKK500], [RKK501], [RKK502], [RKK503], [RKK504], [RKK505], [RKK506], [RKK507], [RKK508], [RKK509], [RKK510], [RKK511], [RKK512], [RKK513], [RKK514], [RKK515], [RKK516], [RKK517], [RKK518], [RKK519], [RKK520], [RKK521], [RKK522], [RKK523], [RKK524], [RKK525], [RKK526], [RKK527], [RKK528], [RKK529], [RKK530], [RKK531], [RKK532], [RKK533], [RKK534], [RKK535], [RKK536], [RKK537], [RKK538], [RKK539], [RKK540], [RKK541], [RKK542], [RKK543], [RKK544], [RKK545], [RKK546], [RKK547], [RKK548], [RKK549], [RKK550], [RKK551], [RKK552], [RKK553], [RKK554], [RKK555], [RKK556], [RKK557], [RKK558], [RKK559], [RKK560], [RKK561], [RKK562], [RKK563], [RKK564], [RKK565], [RKK566], [RKK567], [RKK568], [RKK569], [RKK570], [RKK571], [RKK572], [RKK573], [RKK574], [RKK575], [RKK576], [RKK577], [RKK578], [RKK579], [RKK580], [RKK581], [RKK582], [RKK583], [RKK584], [RKK585], [RKK586], [RKK587], [RKK588], [RKK589], [RKK590], [RKK591], [RKK592], [RKK593], [RKK594], [RKK595], [RKK596], [RKK597], [RKK598], [RKK599], [RKK600], [RKK601], [RKK602], [RKK603], [RKK604], [RKK605], [RKK606], [RKK607], [RKK608], [RKK609], [RKK610], [RKK611], [RKK612], [RKK613], [RKK614], [RKK615], [RKK616], [RKK617], [RKK618], [RKK619], [RKK620], [RKK621], [RKK622], [RKK623], [RKK624], [RKK625], [RKK626], [RKK627], [RKK628], [RKK629], [RKK630], [RKK631], [RKK632], [RKK633], [RKK634], [RKK635], [RKK636], [RKK637], [RKK638], [RKK639], [RKK640], [RKK641], [RKK642], [RKK643], [RKK644], [RKK645], [RKK646], [RKK647], [RKK648], [RKK649], [RKK650], [RKK651], [RKK652], [RKK653], [RKK654], [RKK655], [RKK656], [RKK657], [RKK658], [RKK659], [RKK660], [RKK661], [RKK662], [RKK663], [RKK664], [RKK665], [RKK666], [RKK667], [RKK668], [RKK669], [RKK670], [RKK671], [RKK672], [RKK673], [RKK674], [RKK675], [RKK676], [RKK677], [RKK678], [RKK679], [RKK680], [RKK681], [RKK682], [RKK683], [RKK684], [RKK685], [RKK686], [RKK687], [RKK688], [RKK689], [RKK690], [RKK691], [RKK692], [RKK693], [RKK694], [RKK695], [RKK696], [RKK697], [RKK698], [RKK699], [RKK700], [RKK701], [RKK702], [RKK703], [RKK704], [RKK705], [RKK706], [RKK707], [RKK708], [RKK709], [RKK710], [RKK711], [RKK712], [RKK713], [RKK714], [RKK715], [RKK716], [RKK717], [RKK718], [RKK719], [RKK720], [RKK721], [RKK722], [RKK723], [RKK724], [RKK725], [RKK726], [RKK727], [RKK728], [RKK729], [RKK730], [RKK731], [RKK732], [RKK733], [RKK734], [RKK735], [RKK736], [RKK737], [RKK738], [RKK739], [RKK740], [RKK741], [RKK742], [RKK743], [RKK744], [RKK745], [RKK746], [RKK747], [RKK748], [RKK749], [RKK750], [RKK751], [RKK752], [RKK753], [RKK754], [RKK755], [RKK756], [RKK757], [RKK758], [RKK759], [RKK760], [RKK761], [RKK762], [RKK763], [RKK764], [RKK765], [RKK766], [RKK767], [RKK768], [RKK769], [RKK770], [RKK771], [RKK772], [RKK773], [RKK774], [RKK775], [RKK776], [RKK777], [RKK778], [RKK779], [RKK780], [RKK781], [RKK782], [RKK783], [RKK784], [RKK785], [RKK786], [RKK787], [RKK788], [RKK789], [RKK790], [RKK791], [RKK792], [RKK793], [RKK794], [RKK795], [RKK796], [RKK797], [RKK798], [RKK799], [RKK800], [RKK801], [RKK802], [RKK803], [RKK804], [RKK805], [RKK806], [RKK807], [RKK808], [RKK809], [RKK810], [RKK811], [RKK812], [RKK813], [RKK814], [RKK815], [RKK816], [RKK817], [RKK818], [RKK819], [RKK820], [RKK821], [RKK822], [RKK823], [RKK824], [RKK825], [RKK826], [RKK827], [RKK828], [RKK829], [RKK830], [RKK831], [RKK832], [RKK833], [RKK834], [RKK835], [RKK836], [RKK837], [RKK838], [RKK839], [RKK840], [RKK841], [RKK842], [RKK843], [RKK844], [RKK845], [RKK846], [RKK847], [RKK848], [RKK849], [RKK850], [RKK851], [RKK852], [RKK853], [RKK854], [RKK855], [RKK856], [RKK857], [RKK858], [RKK859], [RKK860], [RKK861], [RKK862], [RKK863], [RKK864], [RKK865], [RKK866], [RKK867], [RKK868], [RKK869], [RKK870], [RKK871], [RKK872], [RKK873], [RKK874], [RKK875], [RKK876], [RKK877], [RKK878], [RKK879], [RKK880], [RKK881], [RKK882], [RKK883], [RKK884], [RKK885], [RKK886], [RKK887], [RKK888], [RKK889], [RKK890], [RKK891], [RKK892], [RKK893], [RKK894], [RKK895], [RKK896], [RKK897], [RKK898], [RKK899], [RKK900], [RKK901], [RKK902], [RKK903], [RKK904], [RKK905], [RKK906], [RKK907], [RKK908], [RKK909], [RKK910], [RKK911], [RKK912], [RKK913], [RKK914], [RKK915], [RKK916], [RKK917], [RKK918], [RKK919], [RKK920], [RKK921], [RKK922], [RKK923], [RKK924], [RKK925], [RKK926], [RKK927], [RKK928], [RKK929], [RKK930], [RKK931], [RKK932], [RKK933], [RKK934], [RKK935], [RKK936], [RKK937], [RKK938], [RKK939], [RKK940], [RKK941], [RKK942], [RKK943], [RKK944], [RKK945], [RKK946], [RKK947], [RKK948], [RKK949], [RKK950], [RKK951], [RKK952], [RKK953], [RKK954], [RKK955], [RKK956], [RKK957], [RKK958], [RKK959], [RKK960], [RKK961], [RKK962], [RKK963], [RKK964], [RKK965], [RKK966], [RKK967], [RKK968], [RKK969], [RKK970], [RKK971], [RKK972], [RKK973], [RKK974], [RKK975], [RKK976], [RKK977], [RKK978], [RKK979], [RKK980], [RKK981], [RKK982], [RKK983], [RKK984], [RKK985], [RKK986], [RKK987], [RKK988], [RKK989], [RKK990], [RKK991], [RKK992], [RKK993], [RKK994], [RKK995], [RKK996], [RKK997], [RKK998], [RKK999], [RKK1000], [RKK1001], [RKK1002], [RKK1003], [RKK1004], [RKK1005], [RKK1006], [RKK1007], [RKK1008], [RKK1009], [RKK1010], [RKK1011], [RKK1012], [RKK1013], [RKK1014], [RKK1015], [RKK1016], [RKK1017], [RKK1018], [RKK1019], [RKK1020], [RKK1021], [RKK1022], [RKK1023], [RKK1024], [RKK1025], [RKK1026], [RKK1027], [RKK1028], [RKK1029], [RKK1030], [RKK1031], [RKK1032], [RKK1033], [RKK1034], [RKK1035], [RKK1036], [RKK1037], [RKK1038], [RKK1039], [RKK1040], [RKK1041], [RKK1042], [RKK1043], [RKK1044], [RKK1045], [RKK1046], [RKK1047], [RKK1048], [RKK1049], [RKK1050], [RKK1051], [RKK1052], [RKK1053], [RKK1054], [RKK1055], [RKK1056], [RKK1057], [RKK1058], [RKK1059], [RKK1060], [RKK1061], [RKK1062], [RKK1063], [RKK1064], [RKK1065], [RKK1066], [RKK1067], [RKK1068], [RKK1069], [RKK1070], [RKK1071], [RKK1072], [RKK1073], [RKK1074], [RKK1075], [RKK1076], [RKK1077], [RKK1078], [RKK1079], [RKK1080], [RKK1081], [RKK1082], [RKK1083], [RKK1084], [RKK1085], [RKK1086], [RKK1087], [RKK1088], [RKK1089], [RKK1090], [RKK1091], [RKK1092], [RKK1093], [RKK1094], [RKK1095], [RKK1096], [RKK1097], [RKK1098], [RKK1099], [RKK1100], [RKK1101], [RKK1102], [RKK1103], [RKK1104], [RKK1105], [RKK1106], [RKK1107], [RKK1108], [RKK1109], [RKK1110], [RKK1111], [RKK1112], [RKK1113], [RKK1114], [RKK1115], [RKK1116], [RKK1117], [RKK1118], [RKK1119], [RKK1120], [RKK1121], [RKK1122], [RKK1123], [RKK1124], [RKK1125], [RKK1126], [RKK1127], [RKK1128], [RKK1129], [RKK1130], [RKK1131], [RKK1132], [RKK1133], [RKK1134], [RKK1135], [RKK1136], [RKK1137], [RKK1138], [RKK1139], [RKK1140], [RKK1141], [RKK1142], [RKK1143], [RKK1144], [RKK1145], [RKK1146], [RKK1147], [RKK1148], [RKK1149], [RKK1150], [RKK1151], [RKK1152], [RKK1153], [RKK1154], [RKK1155], [RKK1156], [RKK1157], [RKK1158], [RKK1159], [RKK1160], [RKK1161], [RKK1162], [RKK1163], [RKK1164], [RKK1165], [RKK1166], [RKK1167], [RKK1168], [RKK1169], [RKK1170], [RKK1171], [RKK1172], [RKK1173], [RKK1174], [RKK1175], [RKK1176], [RKK1177], [RKK1178], [RKK1179], [RKK1180], [RKK1181], [RKK1182], [RKK1183], [RKK1184], [RKK1185], [RKK1186], [RKK1187], [RKK1188], [RKK1189], [RKK1190], [RKK1191], [RKK1192], [RKK1193], [RKK1194], [RKK1195], [RKK1196], [RKK1197], [RKK1198], [RKK1199], [RKK1200], [RKK1201], [RKK1202], [RKK1203], [RKK1204], [RKK1205], [RKK1206], [RKK1207], [RKK1208], [RKK1209], [RKK1210], [RKK1211], [RKK1212], [RKK1213], [RKK1214], [RKK1215], [RKK1216], [RKK1217], [RKK1218], [RKK1219], [RKK1220], [RKK1221], [RKK1222], [RKK1223], [RKK1224], [RKK1225], [RKK1226], [RKK1227], [RKK1228], [RKK1229], [RKK1230], [RKK1231], [RKK1232], [RKK1233], [RKK1234], [RKK1235], [RKK1236], [RKK1237], [RKK1238], [RKK1239], [RKK1240], [RKK1241], [RKK1242], [RKK1243], [RKK1244], [RKK1245], [RKK1246], [RKK1247], [RKK1248], [RKK1249], [RKK1250], [RKK1251], [RKK1252], [RKK1253], [RKK1254], [RKK1255], [RKK1256], [RKK1257], [RKK1258], [RKK1259], [RKK1260], [RKK1261], [RKK1262], [RKK1263], [RKK1264], [RKK1265], [RKK1266], [RKK1267], [RKK1268], [RKK1269], [RKK1270], [RKK1271], [RKK1272], [RKK1273], [RKK1274], [RKK1275], [RKK1276], [RKK1277], [RKK1278], [RKK1279], [RKK1280], [RKK1281], [RKK1282], [RKK1283], [RKK1284], [RKK1285], [RKK1286], [RKK1287], [RKK1288], [RKK1289], [RKK1290], [RKK1291], [RKK1292], [RKK1293], [RKK1294], [RKK1295], [RKK1296], [RKK1297], [RKK1298], [RKK1299], [RKK1300], [RKK1301], [RKK1302], [RKK1303], [RKK1304], [RKK1305], [RKK1306], [RKK1307], [RKK1308], [RKK1309], [RKK1310], [RKK1311], [RKK1312], [RKK1313], [RKK1314], [RKK1315], [RKK1316], [RKK1317], [RKK1318], [RKK1319], [RKK1320], [RKK1321], [RKK1322], [RKK1323], [RKK1324], [RKK1325], [RKK1326], [RKK1327], [RKK1328], [RKK1329], [RKK1330], [RKK1331], [RKK1332], [RKK1333], [RKK1334], [RKK1335], [RKK1336], [RKK1337], [RKK1338], [RKK1339], [RKK1340], [RKK1341], [RKK1342], [RKK1343], [RKK1344], [RKK1345], [RKK1346], [RKK1347], [RKK1348], [RKK1349], [RKK1350], [RKK1351], [RKK1352], [RKK1353], [RKK1354], [RKK1355], [RKK1356], [RKK1357], [RKK1358], [RKK1359], [RKK1360], [RKK1361], [RKK1362], [RKK1363], [RKK1364], [RKK1365], [RKK1366], [RKK1367], [RKK1368], [RKK1369], [RKK1370], [RKK1371], [RKK1372], [RKK1373], [RKK1374], [RKK1375], [RKK1376], [RKK1377], [RKK1378], [RKK1379], [RKK1380], [RKK1381], [RKK1382], [RKK1383], [RKK1384], [RKK1385], [RKK1386], [RKK1387], [RKK1388], [RKK1389], [RKK1390], [RKK1391], [RKK1392], [RKK1393], [RKK1394], [RKK1395], [RKK1396], [RKK1397], [RKK1398], [RKK1399], [RKK1400], [RKK1401], [RKK1402], [RKK1403], [RKK1404], [RKK1405], [RKK1406], [RKK1407], [RKK1408], [RKK1409], [RKK1410], [RKK1411], [RKK1412], [RKK1413], [RKK1414], [RKK1415], [RKK1416], [RKK1417], [RKK1418], [RKK1419], [RKK1420], [RKK1421], [RKK1422], [RKK1423], [RKK1424], [RKK1425], [RKK1426], [RKK1427], [RKK1428], [RKK1429], [RKK1430], [RKK1431], [RKK1432], [RKK1433], [RKK1434], [RKK1435], [RKK1436], [RKK1437], [RKK1438], [RKK1439], [RKK1440], [RKK1441], [RKK1442], [RKK1443], [RKK1444], [RKK1445], [RKK1446], [RKK1447], [RKK1448], [RKK1449], [RKK1450], [RKK1451], [RKK1452], [RKK1453], [RKK1454], [RKK1455], [RKK1456], [RKK1457], [RKK1458], [RKK1459], [RKK1460], [RKK1461], [RKK1462], [RKK1463], [RKK1464], [RKK1465], [RKK1466], [RKK1467], [RKK1468], [RKK1469], [RKK1470], [RKK1471], [RKK1472], [RKK1473], [RKK1474], [RKK1475], [RKK1476], [RKK1477], [RKK1478], [RKK1479], [RKK1480], [RKK1481], [RKK1482], [RKK1483], [RKK1484], [RKK1485], [RKK1486], [RKK1487], [RKK1488], [RKK1489], [RKK1490], [RKK1491], [RKK1492], [RKK1493], [RKK1494], [RKK1495], [RKK1496], [RKK1497], [RKK1498], [RKK1499], [RKK1500], [RKK1501], [RKK1502], [RKK1503], [RKK1504], [RKK1505], [RKK1506], [RKK1507], [RKK1508], [RKK1509], [RKK1510], [RKK1511], [RKK1512], [RKK1513], [RKK1514], [RKK1515], [RKK1516], [RKK1517], [RKK1518], [RKK1519], [RKK1520], [RKK1521], [RKK1522], [RKK1523], [RKK1524], [RKK1525], [RKK1526], [RKK1527], [RKK1528], [RKK1529], [RKK1530], [RKK1531], [RKK1532], [RKK1533], [RKK1534], [RKK1535], [RKK1536], [RKK1537], [RKK1538], [RKK1539], [RKK1540], [RKK1541], [RKK1542], [RKK1543], [RKK1544], [RKK1545], [RKK1546], [RKK1547], [RKK1548], [RKK1549], [RKK1550], [RKK1551], [RKK1552], [RKK1553], [RKK1554], [RKK1555], [RKK1556], [RKK1557], [RKK1558], [RKK1559], [RKK1560], [RKK1561], [RKK1562], [RKK1563], [RKK1564], [RKK1565], [RKK1566], [RKK1567], [RKK1568], [RKK1569], [RKK1570], [RKK1571], [RKK1572], [RKK1573], [RKK1574], [RKK1575], [RKK1576], [RKK1577], [RKK1578], [RKK1579], [RKK1580], [RKK1581], [RKK1582], [RKK1583], [RKK1584], [RKK1585], [RKK1586], [RKK1587], [RKK1588], [RKK1589], [RKK1590], [RKK1591], [RKK1592], [RKK1593], [RKK1594], [RKK1595], [RKK1596], [RKK1597], [RKK1598], [RKK1599], [RKK1600], [RKK1601], [RKK1602], [RKK1603], [RKK1604], [RKK1605], [RKK1606], [RKK1607], [RKK1608], [RKK1609], [RKK1610], [RKK1611], [RKK1612], [RKK1613], [RKK1614], [RKK1615], [RKK1616], [RKK1617], [RKK1618], [RKK1619], [RKK1620], [RKK1621], [RKK1622], [RKK1623], [RKK1624], [RKK1625], [RKK16



# Online Modified Booth (Radix 4) Encoding

<http://www.ecs.umass.edu/ece/koren/arith/simulator/ModBooth/>

Simulator available on Prof Koren's website

# Delay computation in binary array multiplier

Previous topic:

Delay computation in Array Multiplier (binary inputs):

<http://www.ecs.umass.edu/ece/koren/arith/simulator/ArrMlt/>