

Switching Circuits & Logic Design

Jie-Hong Roland Jiang
江介宏

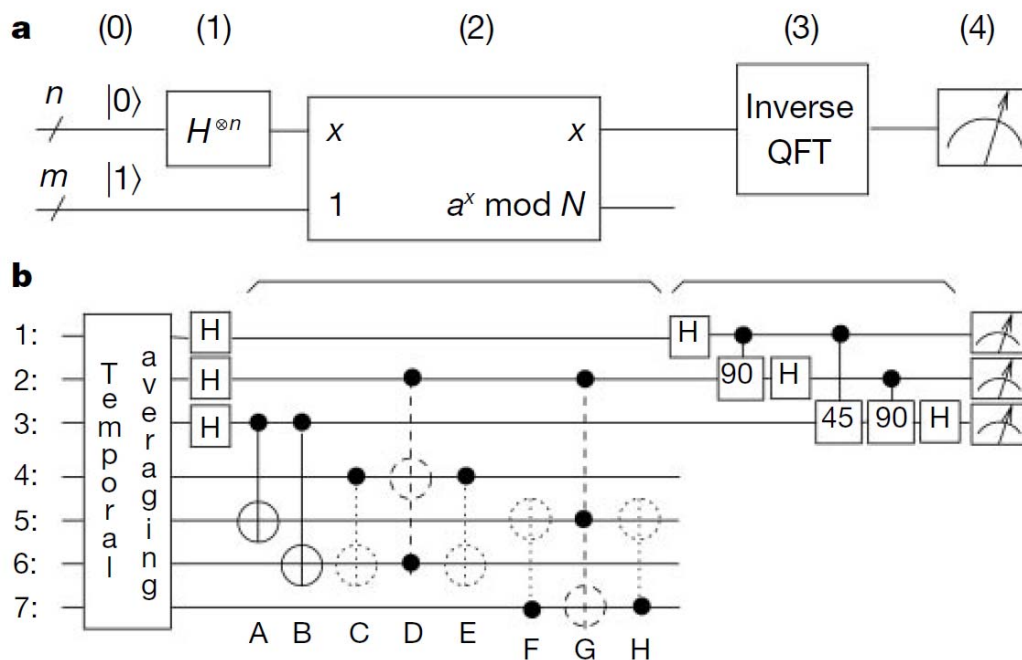
Department of Electrical Engineering
National Taiwan University



Fall 2012

1

§18 Circuits for Arithmetic Operations



Outline

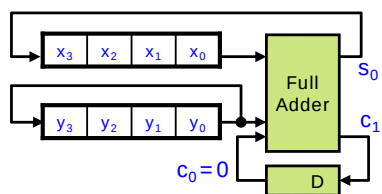
How to use a sequential circuit to control a sequence of operations in a digital system

- Serial adder with accumulator
- Design of a parallel multiplier

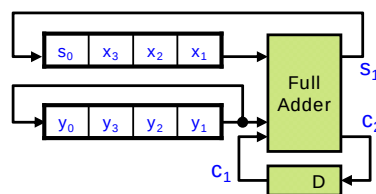
3

Serial Adder with Accumulator Operation

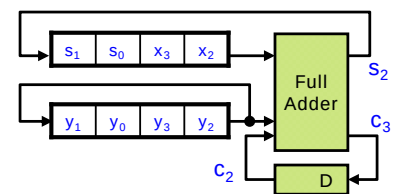
- 4-bit example



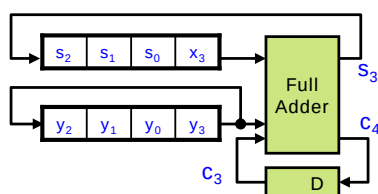
(a) At time t_0



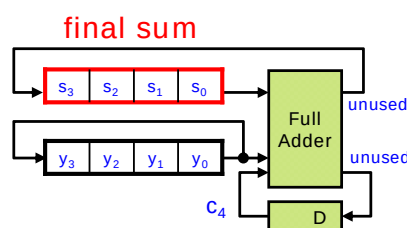
(b) At time t_1



(c) At time t_2



(d) At time t_3



(e) At time t_4

Clear D F/F before next use

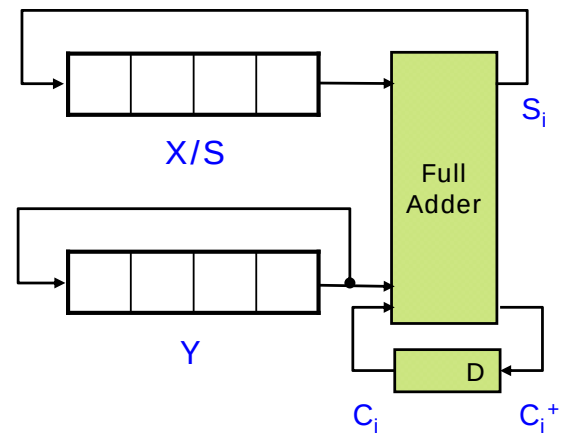
4

Serial Adder with Accumulator Operation

4-bit example

	X	Y	C_i	S_i	C_i^+
	$(x_3x_2x_1x_0)$	$(y_3y_2y_1y_0)$			
t_0	0101	0111	0	0	1
t_1	0010	1011	1	0	1
t_2	0001	1101	1	1	1
t_3	1000	1110	1	1	0
t_4	1100	0111	0	(1)	(0)

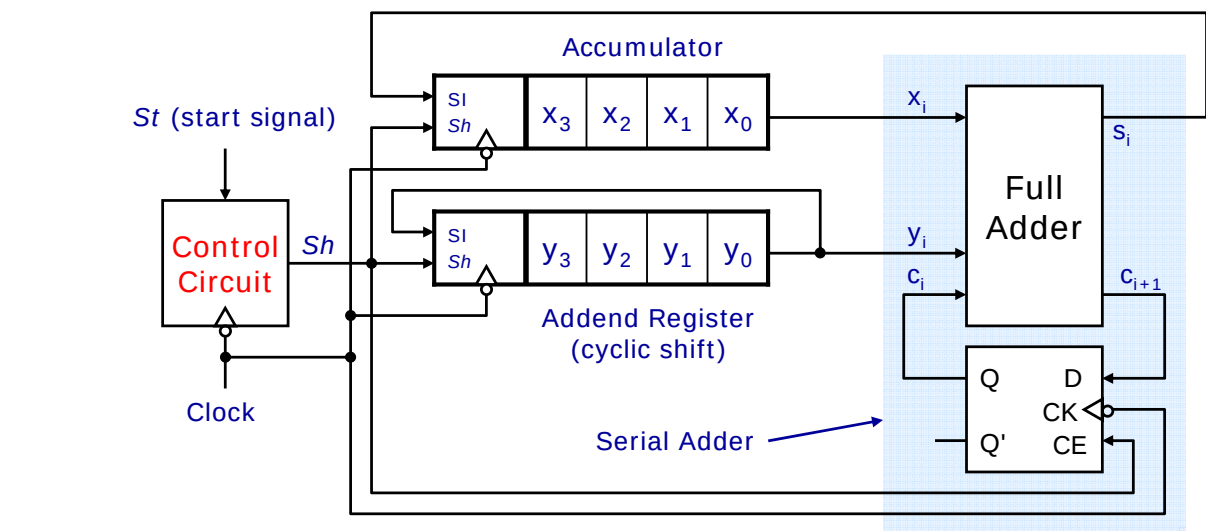
0	1	0	1
+	0	1	1
1	1	0	0



5

Serial Adder with Accumulator

Block diagram

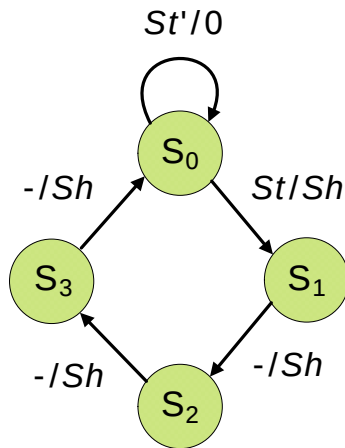


SI: serial input
 St: start signal
 Sh: shift signal } control signals

6

Serial Adder with Accumulator Control Circuit Design

State graph and state table



	Next State		Sh	
	St = 0	1	St = 0	1
S ₀	S ₀	S ₁	0	1
S ₁	S ₂	S ₂	1	1
S ₂	S ₃	S ₃	1	1
S ₃	S ₀	S ₀	1	1

Shift 4 times after St is activated

7

Serial Adder with Accumulator Control Circuit Design

Derivation of control circuit equations

Transition table

	AB	A+B	
		St = 0	1
S ₀	00	00	01
S ₁	01	10	10
S ₂	10	11	11
S ₃	11	00	00



State Assignment

AB	St	
	0	1
00	0	0
01	1	1
11	0	0
10	1	1

A⁺

$$D_A = A'B + AB' = A \oplus B$$

AB	St	
	0	1
00	0	1
01	0	0
11	0	0
10	1	1

B⁺

$$D_B = StB' + AB'$$

AB	St	
	0	1
00	0	1
01	1	1
11	1	1
10	1	1

Sh

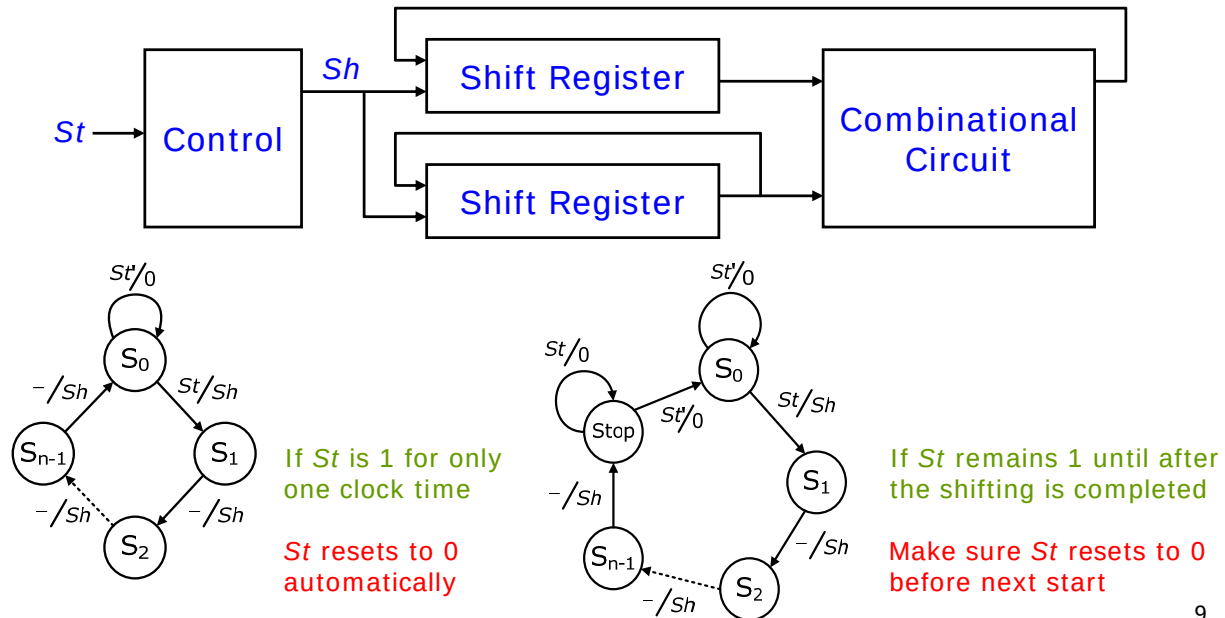
$$Sh = St + A + B$$

8

Serial Adder with Accumulator

Typical Serial Processing Unit

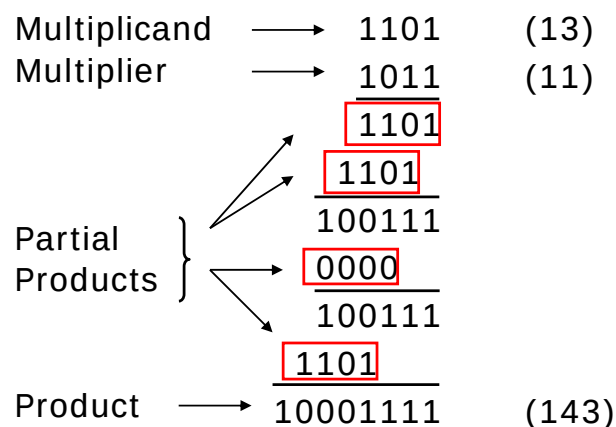
- Typical serial processing unit with n-bit shift registers



9

Design of a Parallel Multiplier

- Design a parallel adder for positive binary numbers
 - Require only shifting and adding
 - Add two binary numbers at a time

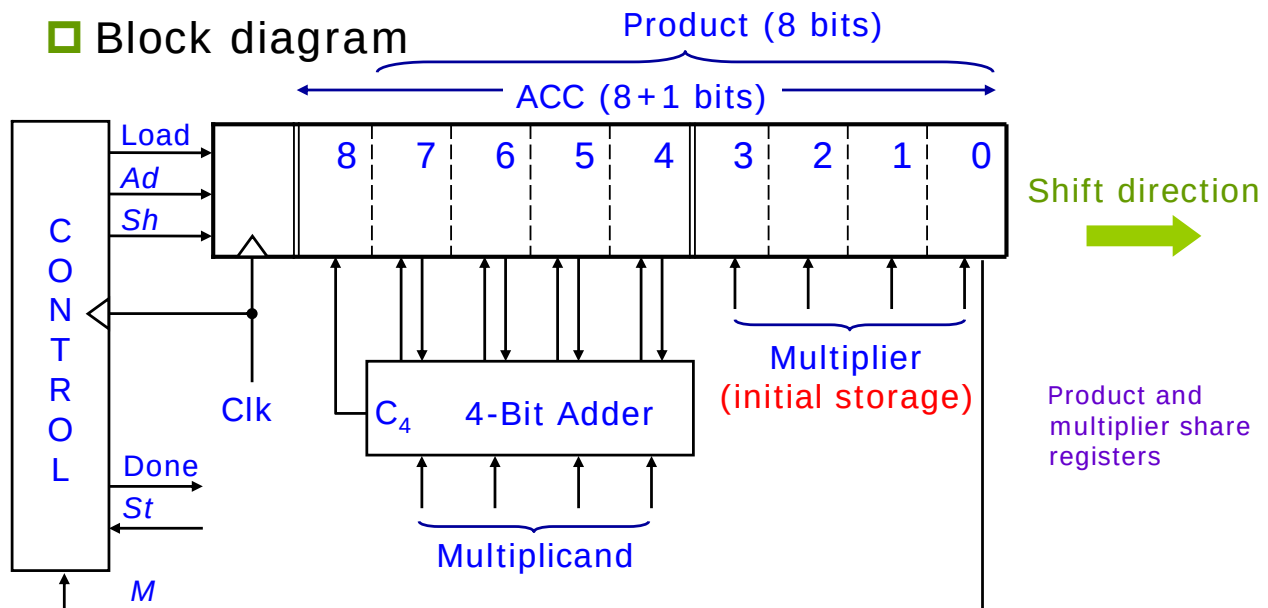


10

Design of a Parallel Multiplier

Parallel Binary Multiplier

Block diagram



Load: load multiplier to ACC[0:3] and clear ACC[4:8]
 Ad: add signal (to store adder outputs in ACC[4:7] and C₄ in ACC[8])
 Sh: shift the ACC contents one place to the right
 M: current multiplier bit

11

Design of a Parallel Multiplier

Parallel Binary Multiplier

Operation example

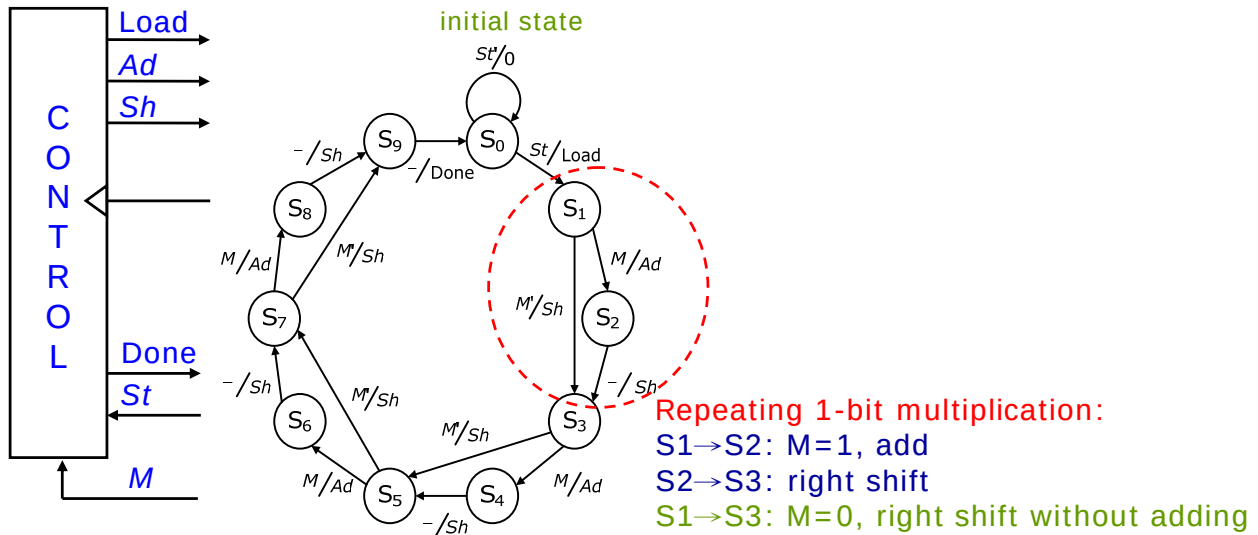
	{product, multiplier}
initial contents of product register	0 0 0 0 0 1 0 1 1 ← M (11)
(add multiplicand because M = 1)	1 1 0 1 (13)
after addition	0 1 1 0 1 1 0 1 1
after shift	0 0 1 1 0 1 1 0 1 1 ← M
(add multiplicand because M = 1)	1 1 0 1
after addition	1 0 0 1 1 1 1 0 1
after shift	0 1 0 0 1 1 1 1 0 1 ← M
(skip addition because M = 0)	
after shift	0 0 1 0 0 1 1 1 1 0 1 ← M
(add multiplicand because M = 1)	1 1 0 1
after addition	1 0 0 0 1 1 1 1 1
after shift (final answer)	0 1 0 0 0 1 1 1 1 (143)

Dividing line between product and multiplier

12

Design of a Parallel Multiplier Control Circuit Design

Method 1 (direct implementation)

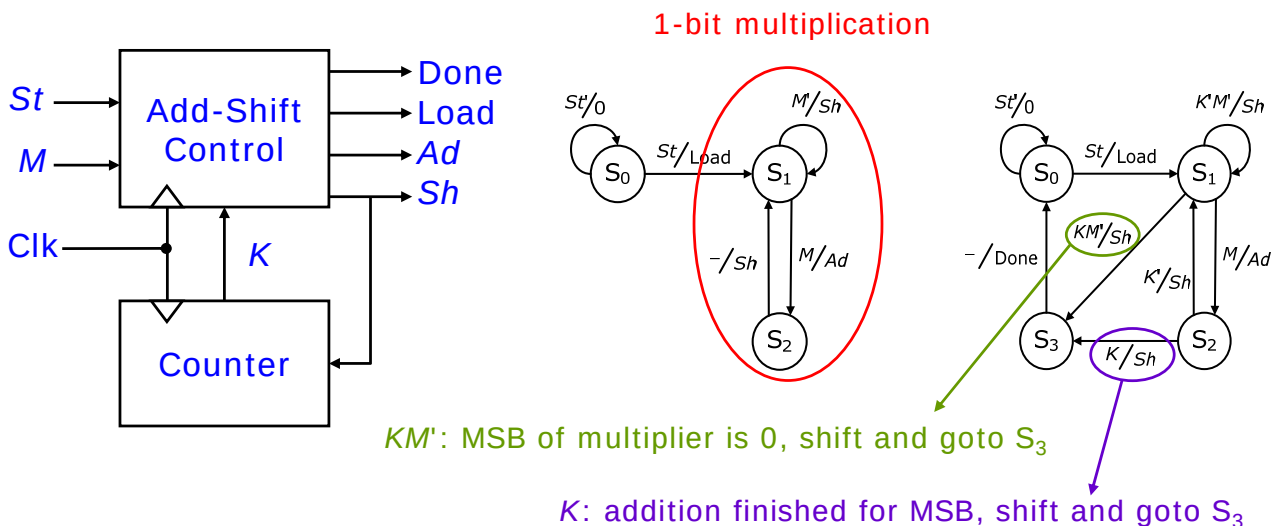


13

Design of a Parallel Multiplier Control Circuit Design

Method 2 (use counter, fewer states)

- Introduce signal K for counting completion



How many states in total (including add-shift control and counter)?

14

Design of a Parallel Multiplier Control Circuit Design

□ Method 2 (cont'd)

Operation example for 1101×1011

Time	State	Counter	Product Register	<i>St</i>	<i>M</i>	<i>K</i>	Load	<i>Ad</i>	<i>Sh</i>	Done
t_0	S_0	00	000000000	0	0	0	0	0	0	0
t_1	S_0	00	000000000	1	0	0	1	0	0	0
t_2	S_1	00	000001011	0	1	0	0	1	0	0
t_3	S_2	00	011011011	0	1	0	0	0	1	0
t_4	S_1	01	001101101	0	1	0	0	1	0	0
t_5	S_2	01	100111101	0	1	0	0	0	1	0
t_6	S_1	10	010011110	0	0	0	0	0	1	0
t_7	S_1	11	001001111	0	1	1	0	1	0	0
t_8	S_2	11	100011111	0	1	1	0	0	1	0
t_9	S_3	00	010001111	0	1	0	0	0	0	1