



# DIGITAL DESIGN & COMPUTER ORGANISATION

## Multiplication-1

---

**Sudarshan T S B., Ph.D.**  
Department of Computer Science  
& Engineering



# DIGITAL DESIGN & COMPUTER ORGANISATION

---

## Multiplication-1

Sudarshan T S B., Ph.D.

Department of Computer Science & Engineering

# MULTIPLICATION - 1

## Course Outline

---

- Digital Design
  - ▶ Combinational logic design
  - ▶ Sequential logic design
  - ★ Multiplication – 1
  
- Computer Organisation
  - ▶ Architecture (microprocessor instruction set)
  - ▶ Microarchitecture (microprocessor operation)

Concepts covered

- Binary Multiplication



## MULTIPLICATION - 1

### Array Multiplication

---

- Manual method of multiplication of 2 decimal numbers
- Similarly we can accomplish multiplication of 2 binary numbers



## MULTIPLICATION - 1

### Array Multiplication

---

- Manual method of multiplication of 2 decimal numbers
- Similarly we can accomplish multiplication of 2 binary numbers

$$\begin{array}{r} 13 \\ \times 11 \\ \hline 13 \\ 13 \\ \hline 143 \end{array}$$



## MULTIPLICATION - 1

### Array Multiplication

---

- Manual method of multiplication of 2 decimal numbers
- Similarly we can accomplish multiplication of 2 binary numbers

$$\begin{array}{r} 13 \\ \times 11 \\ \hline \end{array}$$

1 1 0 1

$$\begin{array}{r} 13 \\ 13 \\ \hline \end{array}$$

143

$$\begin{array}{r} \\ \\ \hline \end{array}$$



## MULTIPLICATION - 1

### Array Multiplication

---

- Manual method of multiplication of 2 decimal numbers
- Similarly we can accomplish multiplication of 2 binary numbers

$$\begin{array}{r} 13 \\ \times 11 \\ \hline \end{array}$$

$$\begin{array}{r} 1101 \\ \times 1011 \\ \hline \end{array}$$



## MULTIPLICATION - 1

### Array Multiplication

---



- Manual method of multiplication of 2 decimal numbers
- Similarly we can accomplish multiplication of 2 binary numbers

13 X 11 -----  13 13 ----- 143 -----	<p>1 1 0 1      (13) Multiplicand, M</p> <p>X 1 0 1 1</p>
--	---

## MULTIPLICATION - 1

### Array Multiplication

---



- Manual method of multiplication of 2 decimal numbers
- Similarly we can accomplish multiplication of 2 binary numbers

13
X 11
-----
13
13
-----
143
-----

1 1 0 1	(13) Multiplicand, M
X 1 0 1 1	(11) Multiplier, Q

## MULTIPLICATION - 1

### Array Multiplication

---



- Manual method of multiplication of 2 decimal numbers
- Similarly we can accomplish multiplication of 2 binary numbers

13 X 11 -----  13 13 ----- 143 -----	1 1 0 1 X 1 0 1 1 -----	<p>(13) Multiplicand, M (11) Multiplier, Q</p>
--	-------------------------------	--

## MULTIPLICATION - 1

### Array Multiplication

---



- Manual method of multiplication of 2 decimal numbers
- Similarly we can accomplish multiplication of 2 binary numbers

13 X 11 -----  13 13 ----- 143 -----	1 1 0 1 X 1 0 1 1 -----  1 1 0 1	<p>(13) Multiplicand, M (11) Multiplier, Q</p>
--	--	--

# MULTIPLICATION - 1

## Array Multiplication

---



- Manual method of multiplication of 2 decimal numbers
- Similarly we can accomplish multiplication of 2 binary numbers

13 X 11 -----  13 13 -----  143 -----	1 1 0 1 X 1 0 1 1 -----  1 1 0 1 1 1 0 1 -----	<p>(13) Multiplicand, M (11) Multiplier, Q</p>
--	--	--

## MULTIPLICATION - 1

### Array Multiplication

---



- Manual method of multiplication of 2 decimal numbers
- Similarly we can accomplish multiplication of 2 binary numbers

13 X 11 -----  13 13 ----- 143	$\begin{array}{r} 1101 \\ \times 1011 \\ \hline \end{array}$  $\begin{array}{r} 1101 \\ 1101 \\ \hline 0000 \end{array}$	<p>(13) Multiplicand, M (11) Multiplier, Q</p>
---	--	--

# MULTIPLICATION - 1

## Array Multiplication



- Manual method of multiplication of 2 decimal numbers
- Similarly we can accomplish multiplication of 2 binary numbers

13 X 11 -----  13 13 ----- 143	$\begin{array}{r} 1101 \\ \times 1011 \\ \hline \end{array}$  $\begin{array}{r} 1101 \\ 1101 \\ 0000 \\ 1101 \\ \hline \end{array}$	<p>(13) Multiplicand, M (11) Multiplier, Q</p>
---	---	--

## MULTIPLICATION - 1

### Array Multiplication

---



- Manual method of multiplication of 2 decimal numbers
- Similarly we can accomplish multiplication of 2 binary numbers

13 X 11 -----  13 13 ----- 143	$\begin{array}{r} 1101 \\ \times 1011 \\ \hline \end{array}$  $\begin{array}{r} 1101 \\ 1101 \\ 0000 \\ \hline 1101 \end{array}$	<p>(13) Multiplicand, M (11) Multiplier, Q</p>
---	--	--

## MULTIPLICATION - 1

### Array Multiplication

---



- Manual method of multiplication of 2 decimal numbers
- Similarly we can accomplish multiplication of 2 binary numbers

13 X 11 -----  13 13 ----- 143 -----	1 1 0 1 X 1 0 1 1 -----  1 1 0 1 1 1 0 1 0 0 0 0 1 1 0 1 -----  1 0 0 0 1 1 1 1	<p>(13) Multiplicand, M</p> <p>(11) Multiplier, Q</p>
--	---	---

## MULTIPLICATION - 1

### Array Multiplication

---



- Manual method of multiplication of 2 decimal numbers
- Similarly we can accomplish multiplication of 2 binary numbers

13 X 11 -----	1 1 0 1 X 1 0 1 1 -----	<p>(13) Multiplicand, M (11) Multiplier, Q</p> <p>13 13 ----- 0 0 0 1 1 0 1 ----- 1 0 0 0 1 1 1 1</p> <p>(143) Product, P</p>
---------------------	-------------------------------	---

# MULTIPLICATION - 1

## Array Multiplication

- Manual method of multiplication of 2 decimal numbers
- Similarly we can accomplish multiplication of 2 binary numbers

- This algorithm applies to unsigned numbers or positive signed numbers.
- The product of two n-digit numbers will result in  $2n$  digit result.

13	1 1 0 1	(13) Multiplicand, M
X 11	X 1 0 1 1	(11) Multiplier, Q
-----	-----	
13	1 1 0 1	
13	1 1 0 1	
-----	0 0 0 0	
143	1 1 0 1	
-----	1 0 0 0 1 1 1 1	(143) Product, P

# MULTIPLICATION - 1

## Array Multiplication

- Manual method of multiplication of 2 decimal numbers
- Similarly we can accomplish multiplication of 2 binary numbers

- This algorithm applies to unsigned numbers or positive signed numbers.
- The product of two n-digit numbers will result in  $2n$  digit result.

13	1 1 0 1	(13) Multiplicand, M
X 11	X 1 0 1 1	(11) Multiplier, Q
-----	-----	
13	1 1 0 1	
13	1 1 0 1	
-----	0 0 0 0	
143	1 1 0 1	
-----	1 0 0 0 1 1 1 1	(143) Product, P

- Multiplication of binary numbers can be represented using AND function
- The product can be obtained by adding partial product in each stage
- Multiplication of binary numbers can be achieved using an array of combinational elements

# MULTIPLICATION - 1

## Array Multiplication

1	1	0	1	(13) Multiplicand, M
X	1	0	1	1 (11) Multiplier, Q
-----				

$m_3$	$m_2$	$m_1$	$m_0$
-------	-------	-------	-------

# MULTIPLICATION - 1

## Array Multiplication

1 1 0 1 (13) Multiplicand, M  
X 1 0 1 1 (11) Multiplier, Q

0 0 0 0 Initial Partial Product, PP<sub>0</sub>

m<sub>3</sub> m<sub>2</sub> m<sub>1</sub> m<sub>0</sub>

# MULTIPLICATION - 1

## Array Multiplication

1 1 0 1 (13) Multiplicand, M  
X 1 0 1 1 (11) Multiplier, Q

0 0 0 0 Initial Partial Product, PP<sub>0</sub>

PP<sub>0</sub>

0 m<sub>3</sub> 0 m<sub>2</sub> 0 m<sub>1</sub> 0 m<sub>0</sub>

# MULTIPLICATION - 1

## Array Multiplication

1 1 0 1 (13) Multiplicand, M  
X 1 0 1 1 (11) Multiplier, Q

-----  
0 0 0 0 Initial Partial  
1 1 0 1 Product, PP<sub>0</sub>

PP<sub>0</sub>

0 m<sub>3</sub> 0 m<sub>2</sub> 0 m<sub>1</sub> 0 m<sub>0</sub>

# MULTIPLICATION - 1

## Array Multiplication

1 1 0 1 (13) Multiplicand, M  
X 1 0 1 1 (11) Multiplier, Q

-----  
0 0 0 0 Initial Partial  
1 1 0 1 Product, PP<sub>0</sub>

PP<sub>0</sub>

0 m<sub>3</sub> 0 m<sub>2</sub> 0 m<sub>1</sub> 0 m<sub>0</sub>

# MULTIPLICATION - 1

## Array Multiplication

1 1 0 1 (13) Multiplicand, M  
X 1 0 1 1 (11) Multiplier, Q

0 0 0 0 Initial Partial  
1 1 0 1 Product,  $PP_0$

1 1 0 1 Partial Product,  $PP_1$

$PP_0$

0 m<sub>3</sub> 0 m<sub>2</sub> 0 m<sub>1</sub> 0 m<sub>0</sub>

# MULTIPLICATION - 1

## Array Multiplication

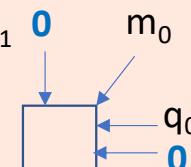
$$\begin{array}{r} 1101 \\ \times 1011 \\ \hline \end{array} \quad \begin{array}{l} (13) \text{ Multiplicand, } M \\ (11) \text{ Multiplier, } Q \end{array}$$

$$\begin{array}{r} 0000 \\ 1101 \\ \hline \end{array} \quad \begin{array}{l} \text{Initial Partial} \\ \text{Product, } PP_0 \end{array}$$

$$1101 \quad \text{Partial Product, } PP_1$$

PP<sub>0</sub>

0 m<sub>3</sub> 0 m<sub>2</sub> 0 m<sub>1</sub> 0 m<sub>0</sub>



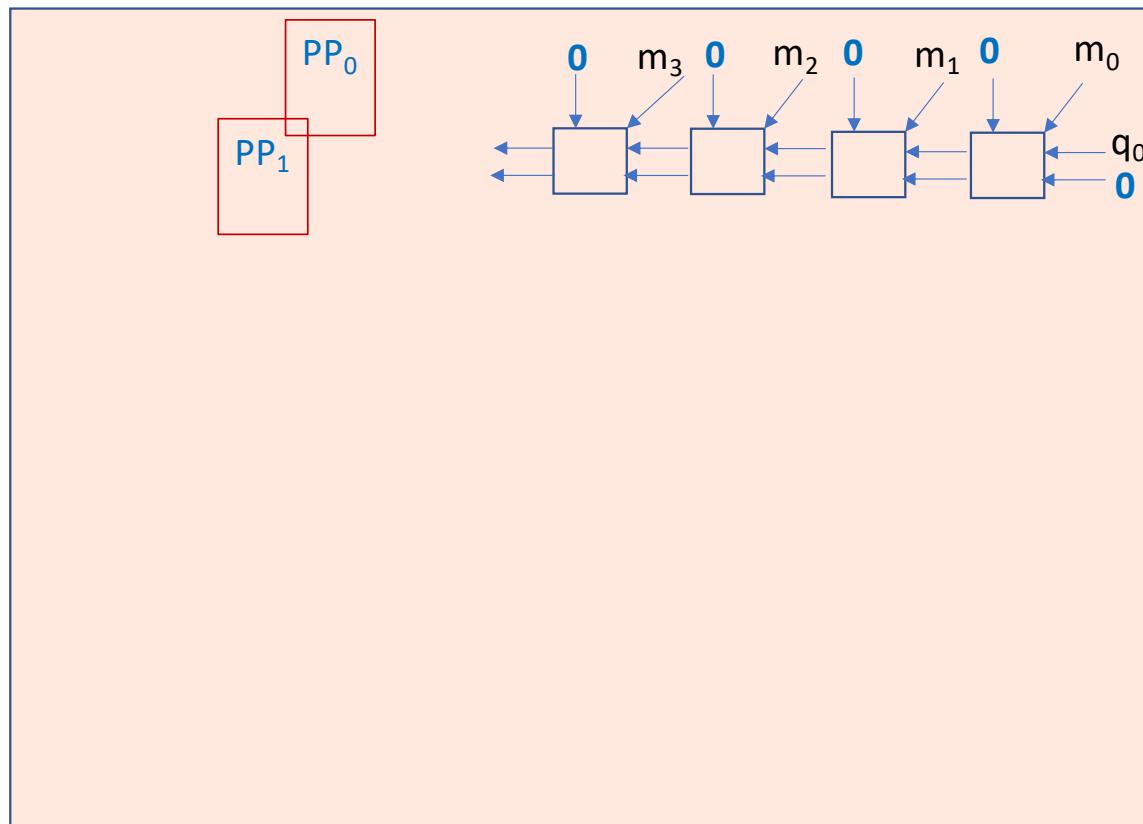
# MULTIPLICATION - 1

## Array Multiplication

$$\begin{array}{r} 1101 \\ \times 1011 \\ \hline \end{array} \quad \begin{array}{l} (13) \text{ Multiplicand, } M \\ (11) \text{ Multiplier, } Q \end{array}$$

-----  
0 0 0 0      Initial Partial  
1 1 0 1      Product,  $PP_0$

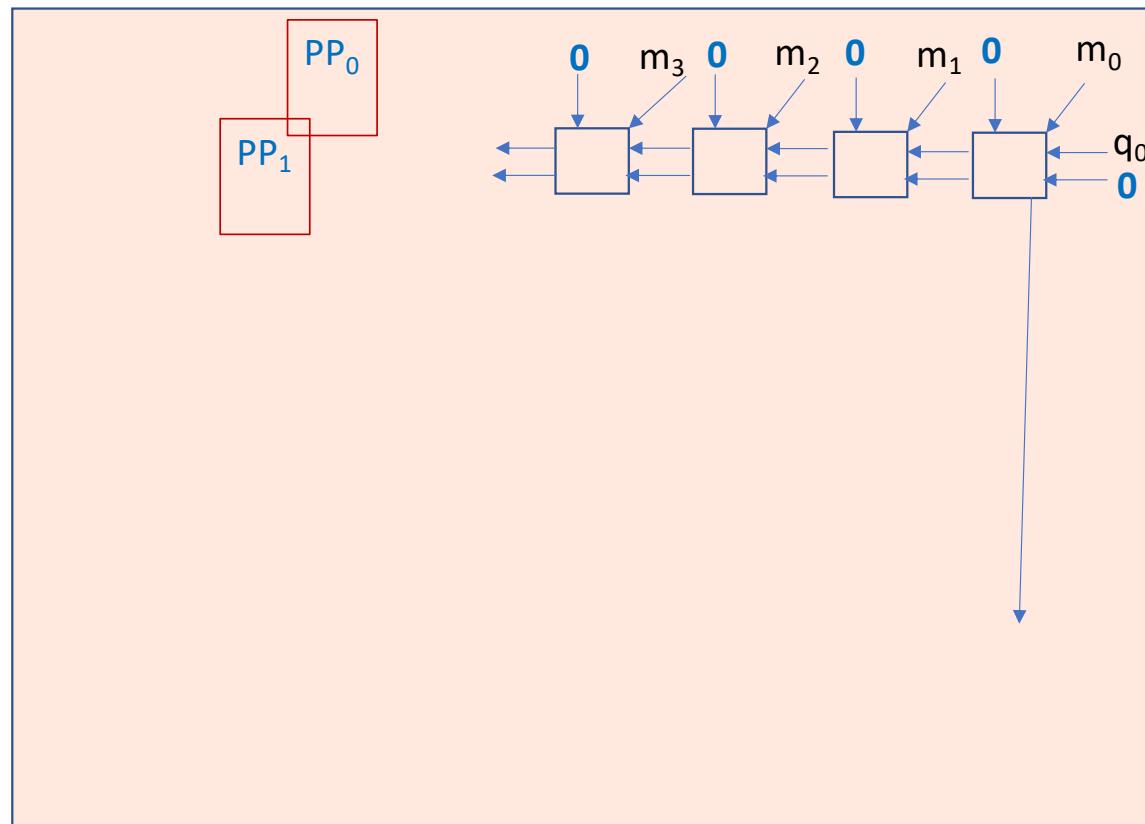
-----  
1 1 0 1      Partial Product,  $PP_1$



# MULTIPLICATION - 1

## Array Multiplication

1 1 0 1	(13) Multiplicand, M
X 1 0 1 1	(11) Multiplier, Q
-----	
0 0 0 0	Initial Partial Product, PP <sub>0</sub>
1 1 0 1	
-----	
1 1 0 1	Partial Product, PP <sub>1</sub>



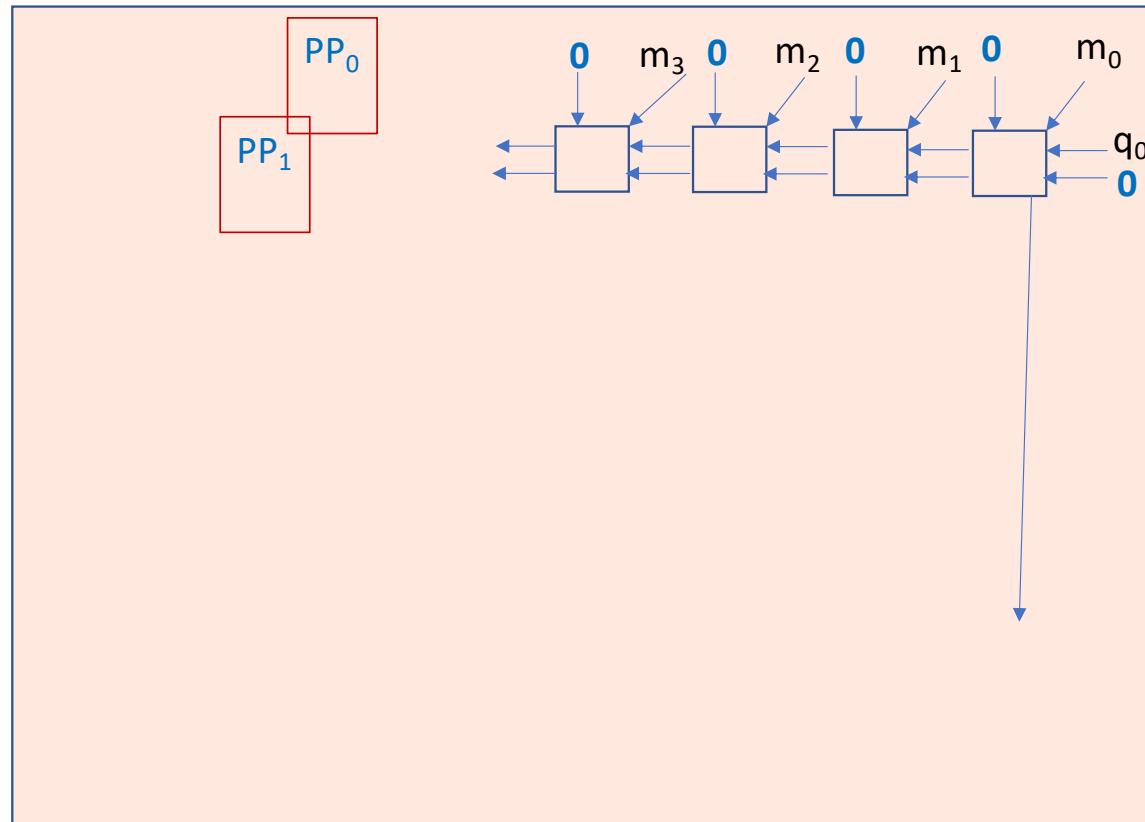
# MULTIPLICATION - 1

## Array Multiplication

$$\begin{array}{r} 1101 \\ \times 1011 \\ \hline \end{array} \quad \begin{array}{l} (13) \text{ Multiplicand, } M \\ (11) \text{ Multiplier, } Q \end{array}$$

-----  
0 0 0 0      Initial Partial  
1 1 0 1      Product,  $PP_0$

-----  
1 1 0 1      Partial Product,  $PP_1$   
1 1 0 1



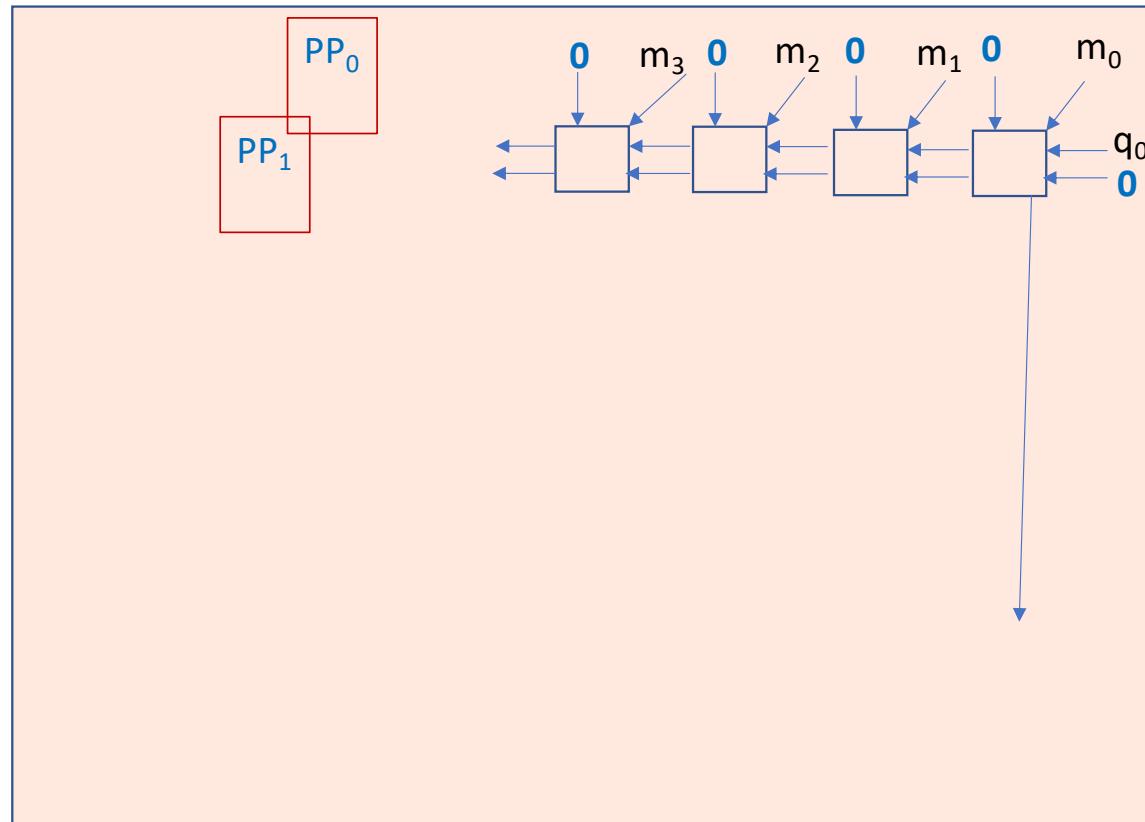
# MULTIPLICATION - 1

## Array Multiplication

$$\begin{array}{r} 1101 \\ \times 1011 \\ \hline \end{array} \quad \begin{array}{l} (13) \text{ Multiplicand, } M \\ (11) \text{ Multiplier, } Q \end{array}$$

$$\begin{array}{r} 0000 \\ 1101 \\ \hline \end{array} \quad \begin{array}{l} \text{Initial Partial} \\ \text{Product, } PP_0 \end{array}$$

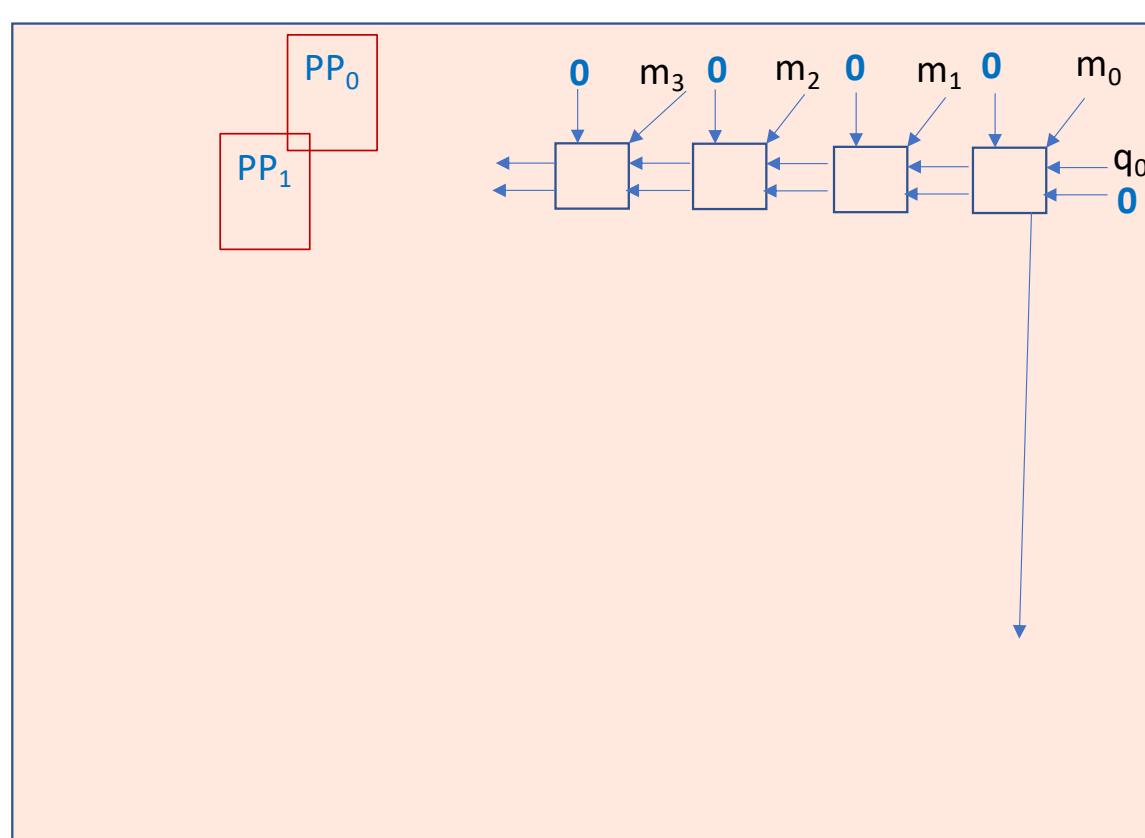
$$\begin{array}{r} 1101 \\ 1101 \\ \hline \end{array} \quad \begin{array}{l} \text{Partial Product, } PP_1 \end{array}$$



# MULTIPLICATION - 1

## Array Multiplication

1 1 0 1	(13) Multiplicand, M
X 1 0 1 1	(11) Multiplier, Q
<hr/>	
0 0 0 0	Initial Partial Product, PP <sub>0</sub>
<hr/>	
1 1 0 1	Partial Product, PP <sub>1</sub>
1 1 0 1	
<hr/>	
1 0 0 1 1 1	



# MULTIPLICATION - 1

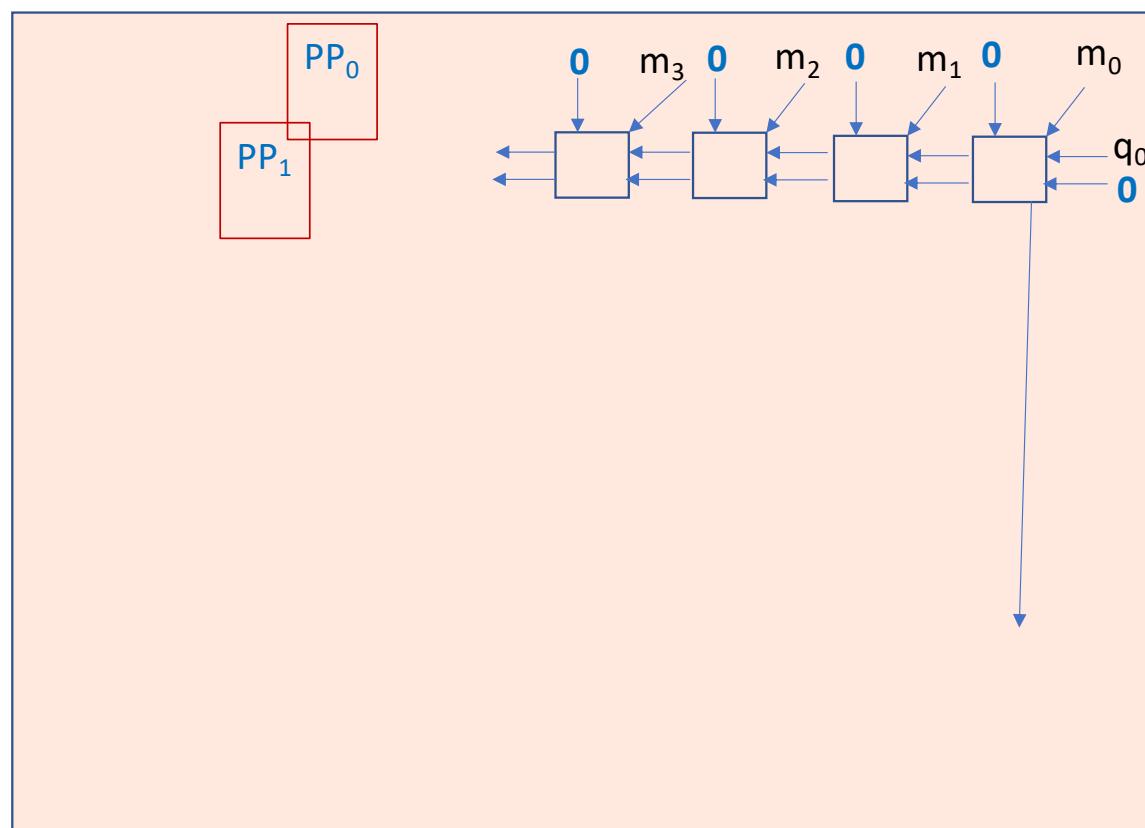
## Array Multiplication

$$\begin{array}{r} 1101 \\ \times 1011 \\ \hline \end{array} \quad \begin{array}{l} (13) \text{ Multiplicand, } M \\ (11) \text{ Multiplier, } Q \end{array}$$

$$\begin{array}{r} 0000 \\ 1101 \\ \hline \end{array} \quad \begin{array}{l} \text{Initial Partial} \\ \text{Product, } PP_0 \end{array}$$

$$\begin{array}{r} 1101 \\ 1101 \\ \hline \end{array} \quad \begin{array}{l} \text{Partial Product, } PP_1 \end{array}$$

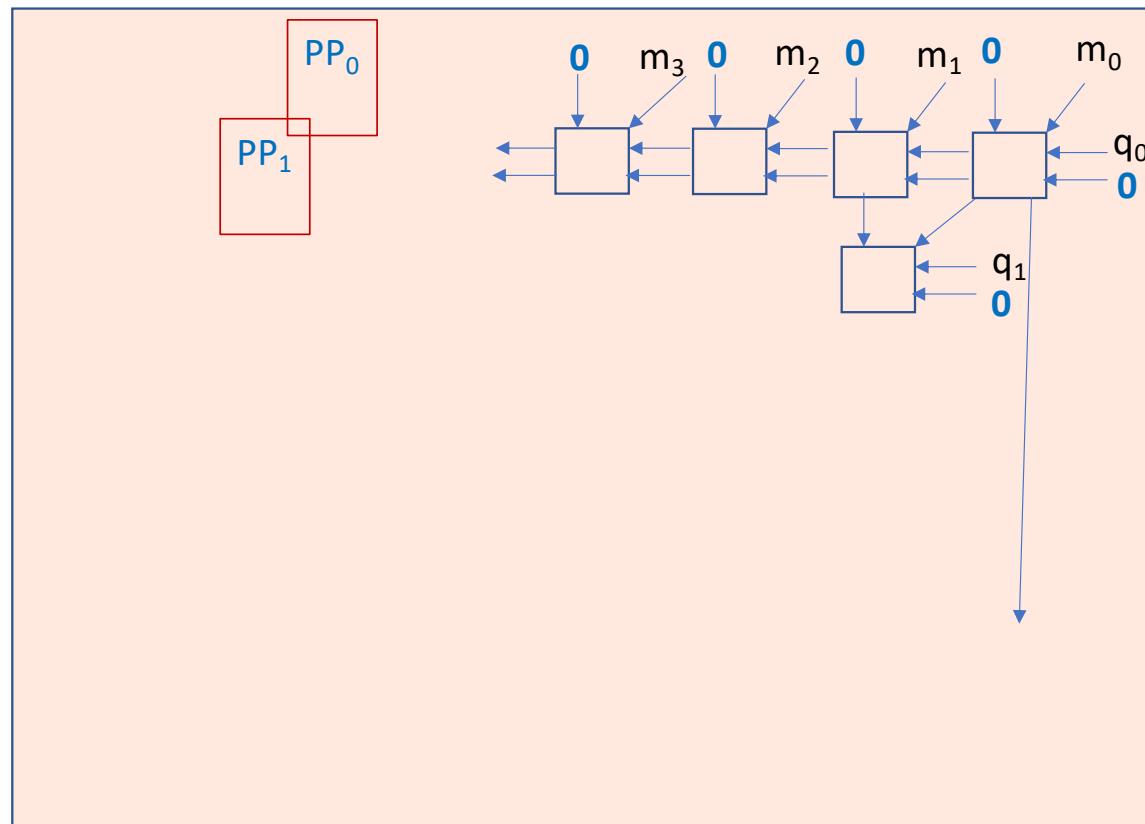
$$\begin{array}{r} 100111 \\ \hline \end{array} \quad \begin{array}{l} \text{Partial Product, } PP_2 \end{array}$$



# MULTIPLICATION - 1

## Array Multiplication

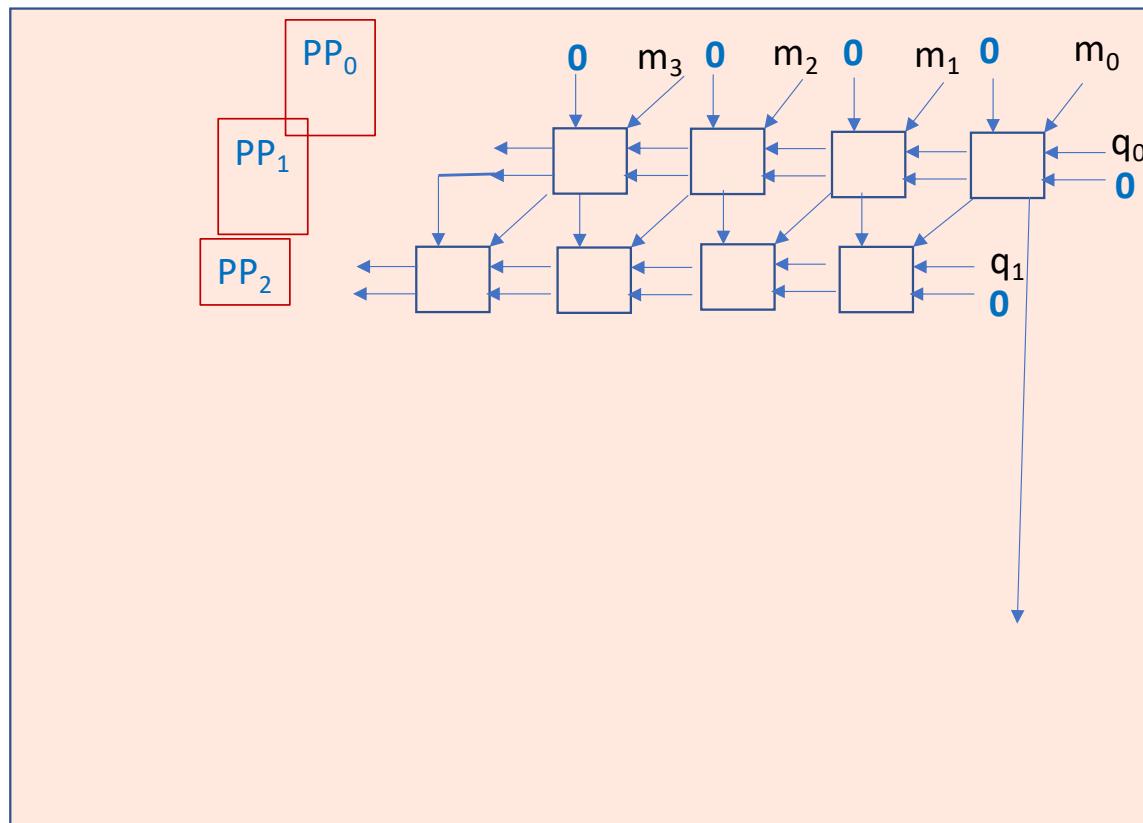
1 1 0 1	(13) Multiplicand, M
X 1 0 1 1	(11) Multiplier, Q
<hr/>	
0 0 0 0	Initial Partial Product, PP <sub>0</sub>
<hr/>	
1 1 0 1	Partial Product, PP <sub>1</sub>
1 1 0 1	
<hr/>	
1 0 0 1 1 1	Partial Product, PP <sub>2</sub>



# MULTIPLICATION - 1

## Array Multiplication

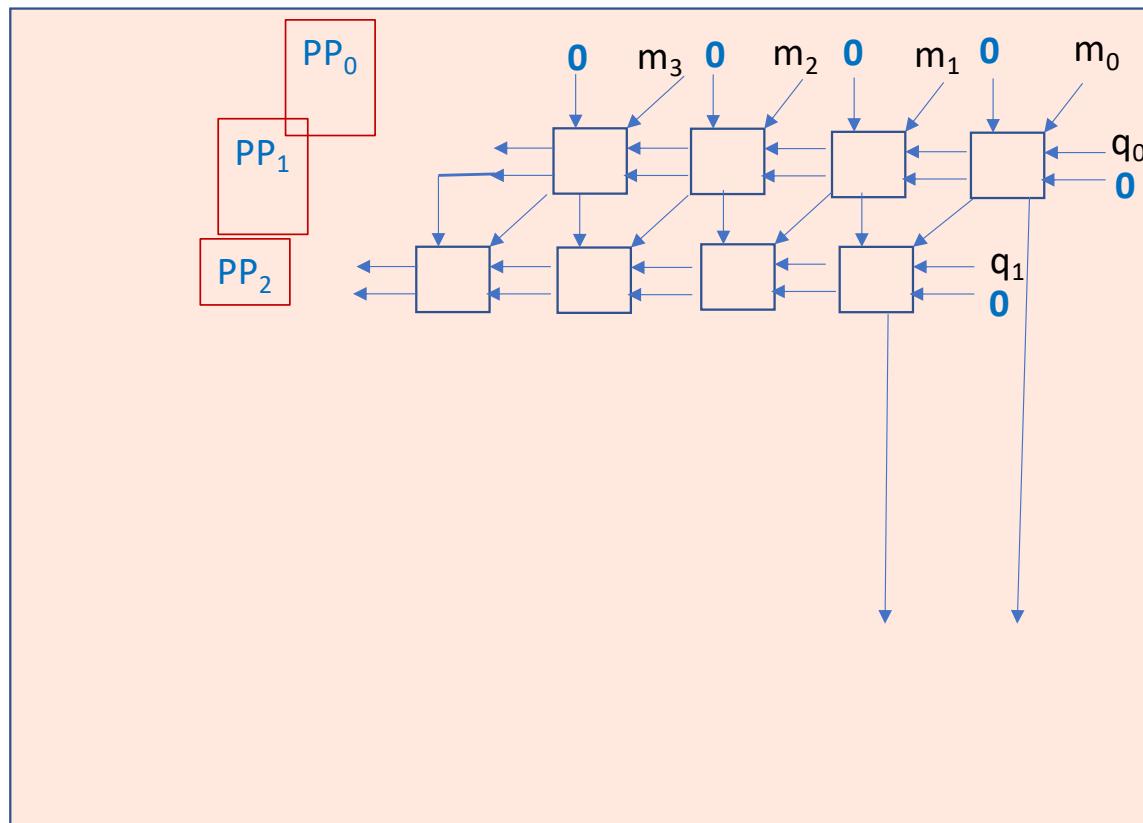
1 1 0 1	(13) Multiplicand, M
X 1 0 1 1	(11) Multiplier, Q
<hr/>	
0 0 0 0	Initial Partial Product, PP <sub>0</sub>
1 1 0 1	Partial Product, PP <sub>1</sub>
<hr/>	
1 1 0 1	Partial Product, PP <sub>2</sub>
1 1 0 1	
<hr/>	



# MULTIPLICATION - 1

## Array Multiplication

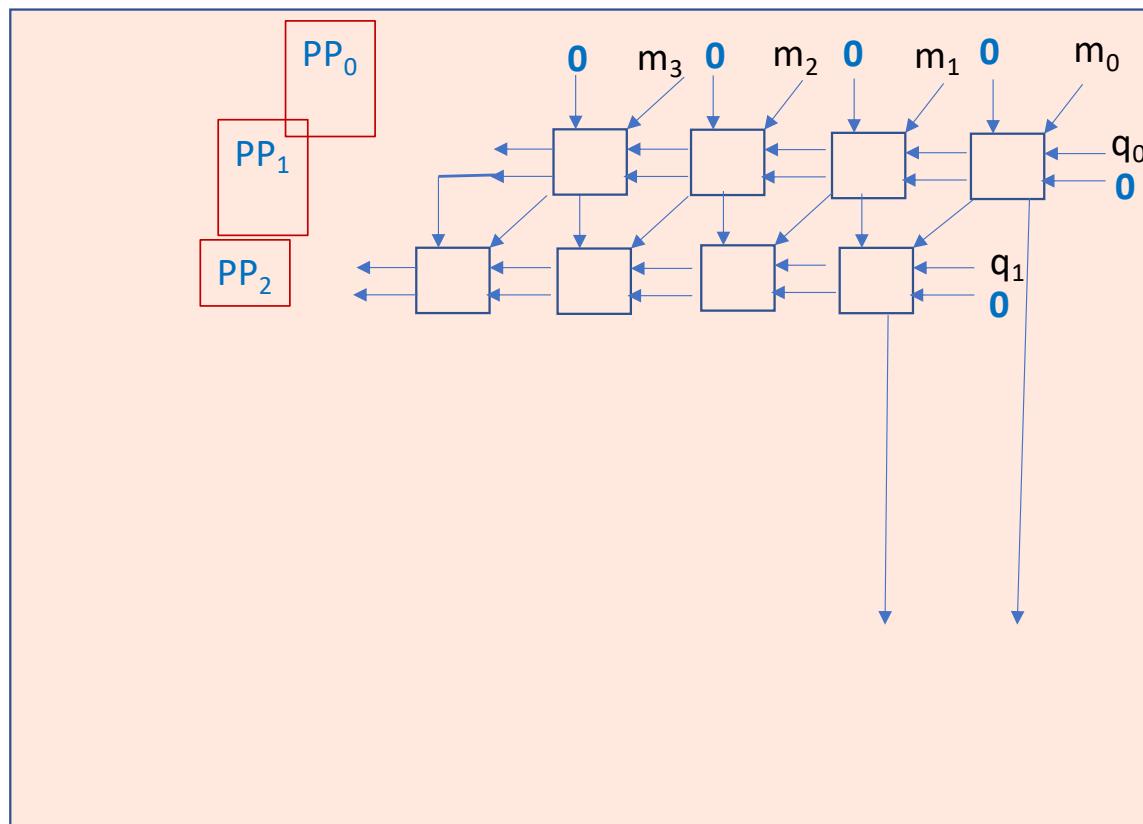
1 1 0 1	(13) Multiplicand, M
X 1 0 1 1	(11) Multiplier, Q
-----	
0 0 0 0	Initial Partial Product, PP <sub>0</sub>
1 1 0 1	
-----	
1 1 0 1	Partial Product, PP <sub>1</sub>
1 1 0 1	
-----	
1 0 0 1 1 1	Partial Product, PP <sub>2</sub>



# MULTIPLICATION - 1

## Array Multiplication

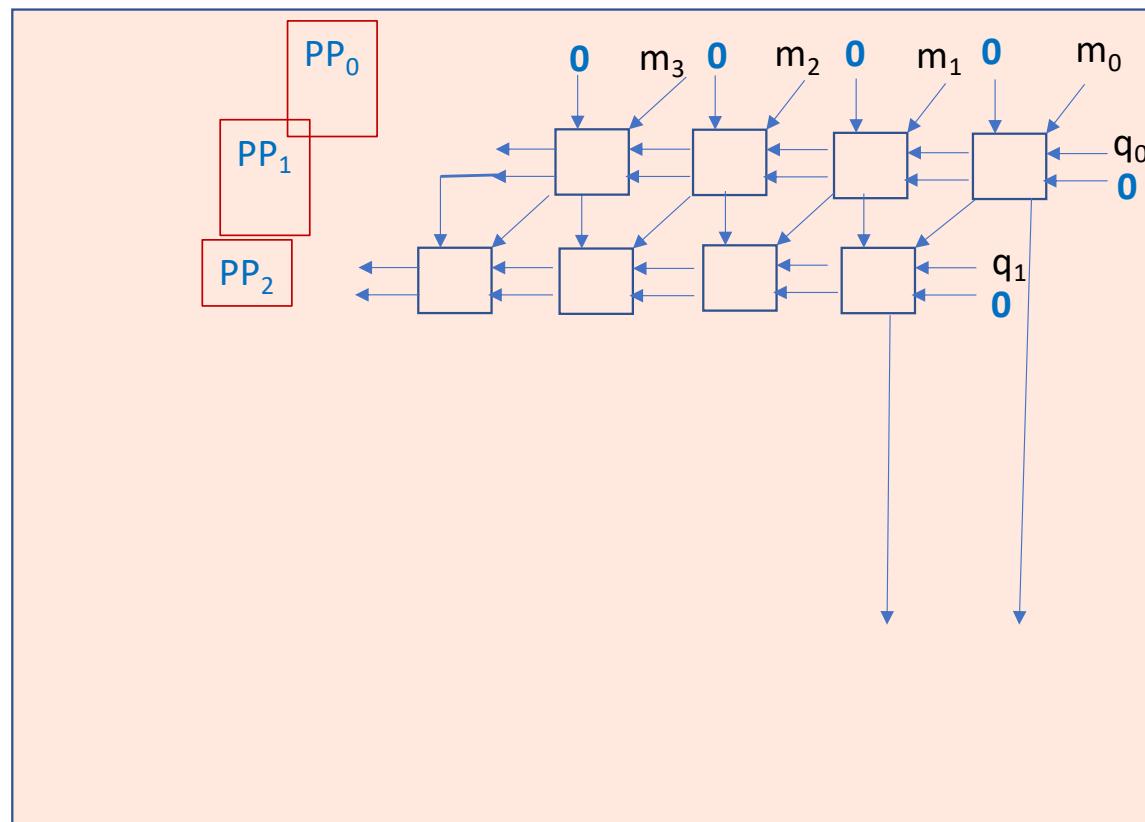
1 1 0 1	(13) Multiplicand, M
X 1 0 1 1	(11) Multiplier, Q
-----	
0 0 0 0	Initial Partial Product, PP <sub>0</sub>
1 1 0 1	
-----	
1 1 0 1	Partial Product, PP <sub>1</sub>
1 1 0 1	
-----	
1 0 0 1 1 1	Partial Product, PP <sub>2</sub>
0 0 0 0	



# MULTIPLICATION - 1

## Array Multiplication

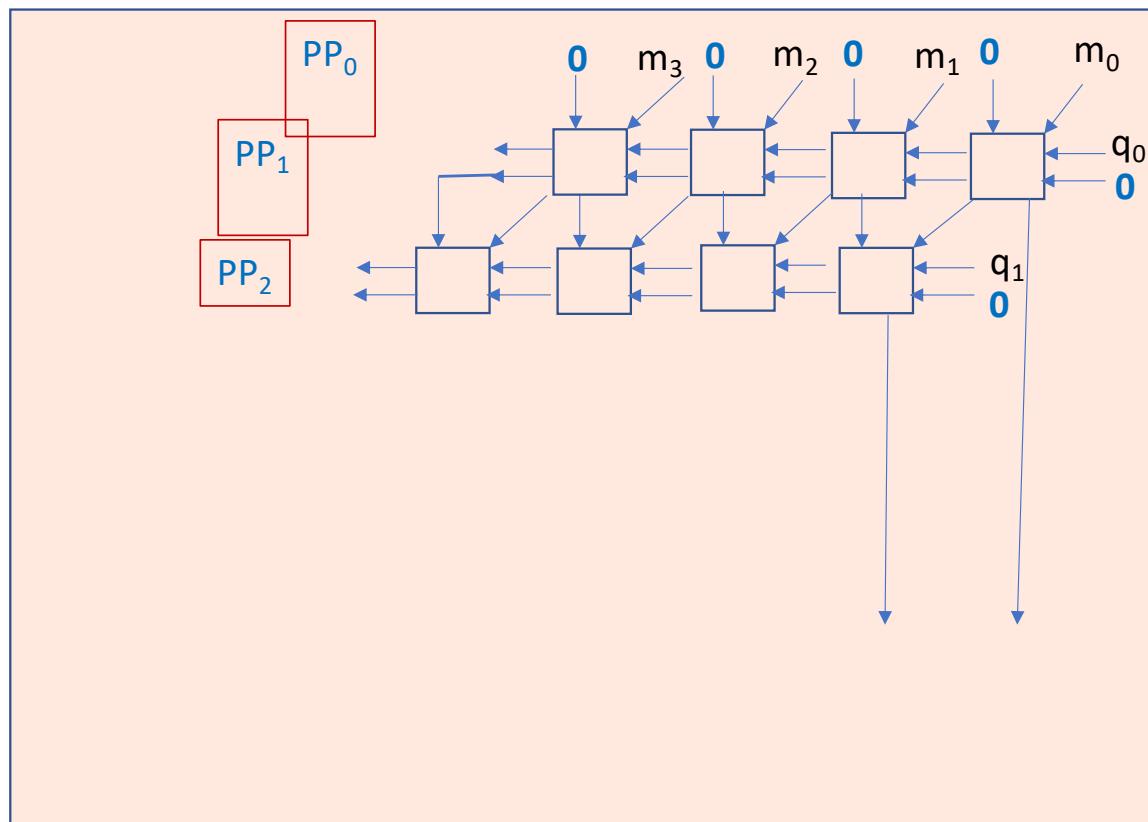
1 1 0 1	(13) Multiplicand, M
X 1 0 1 1	(11) Multiplier, Q
<hr/>	
0 0 0 0	Initial Partial Product, PP <sub>0</sub>
1 1 0 1	
<hr/>	
1 1 0 1	Partial Product, PP <sub>1</sub>
1 1 0 1	
<hr/>	
1 0 0 1 1 1	Partial Product, PP <sub>2</sub>
0 0 0 0	
<hr/>	



# MULTIPLICATION - 1

## Array Multiplication

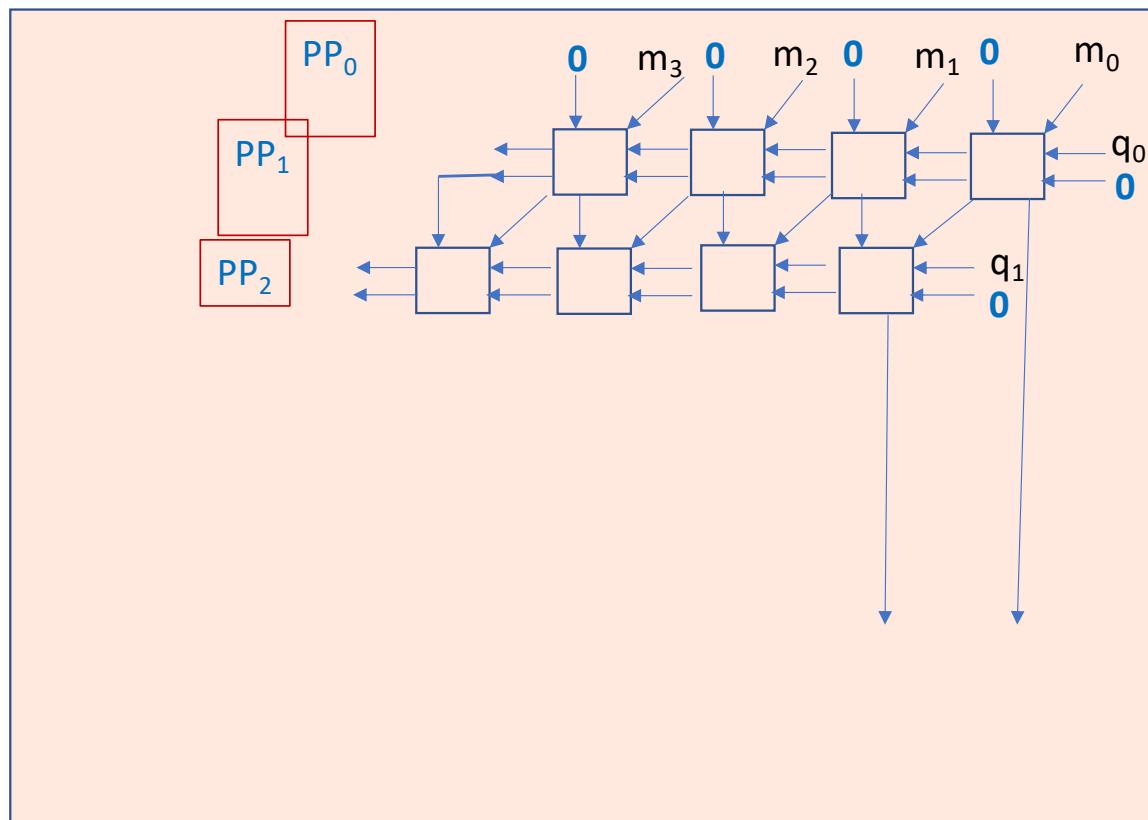
1 1 0 1	(13) Multiplicand, M
X 1 0 1 1	(11) Multiplier, Q
<hr/>	
0 0 0 0	Initial Partial Product, PP <sub>0</sub>
1 1 0 1	
<hr/>	
1 1 0 1	Partial Product, PP <sub>1</sub>
1 1 0 1	
<hr/>	
1 0 0 1 1 1	Partial Product, PP <sub>2</sub>
0 0 0 0	
<hr/>	
1 0 0 1 1 1	



# MULTIPLICATION - 1

## Array Multiplication

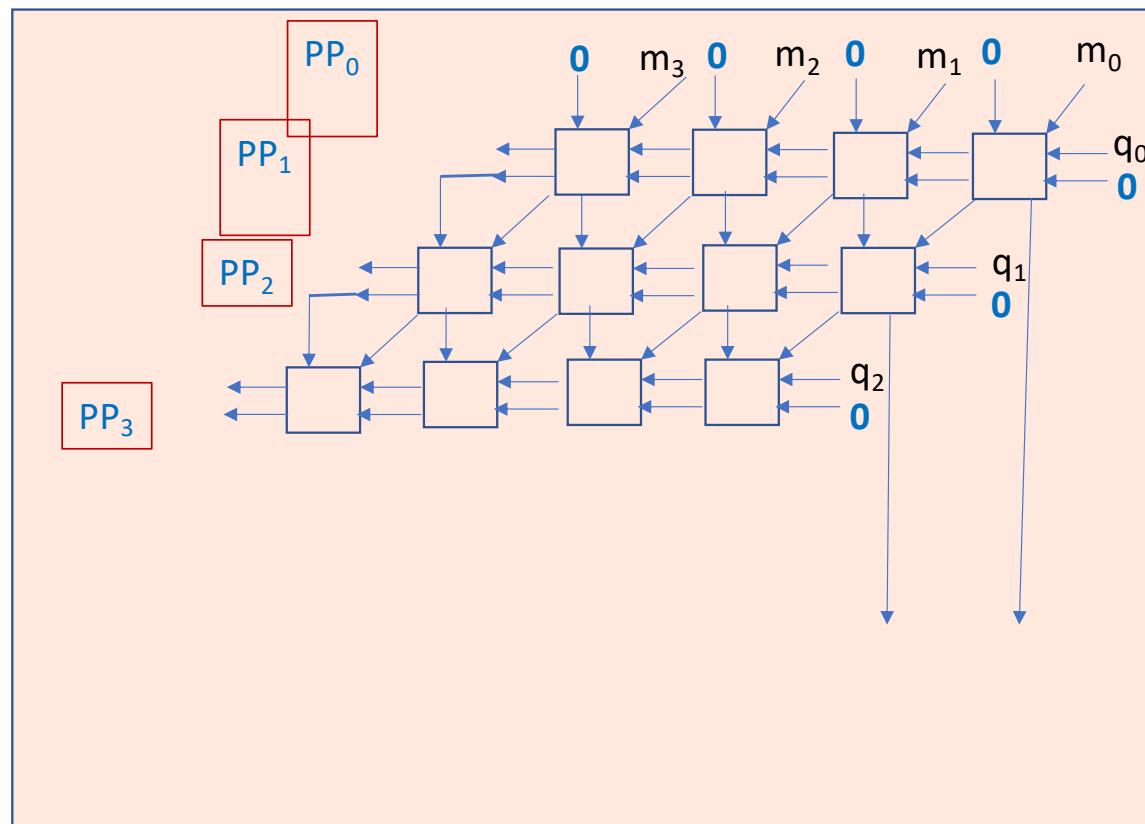
1 1 0 1	(13) Multiplicand, M
X 1 0 1 1	(11) Multiplier, Q
<hr/>	
0 0 0 0	Initial Partial Product, PP <sub>0</sub>
1 1 0 1	
<hr/>	
1 1 0 1	Partial Product, PP <sub>1</sub>
1 1 0 1	
<hr/>	
1 0 0 1 1 1	Partial Product, PP <sub>2</sub>
0 0 0 0	
<hr/>	
1 0 0 1 1 1	Partial Product, PP <sub>3</sub>



# MULTIPLICATION - 1

## Array Multiplication

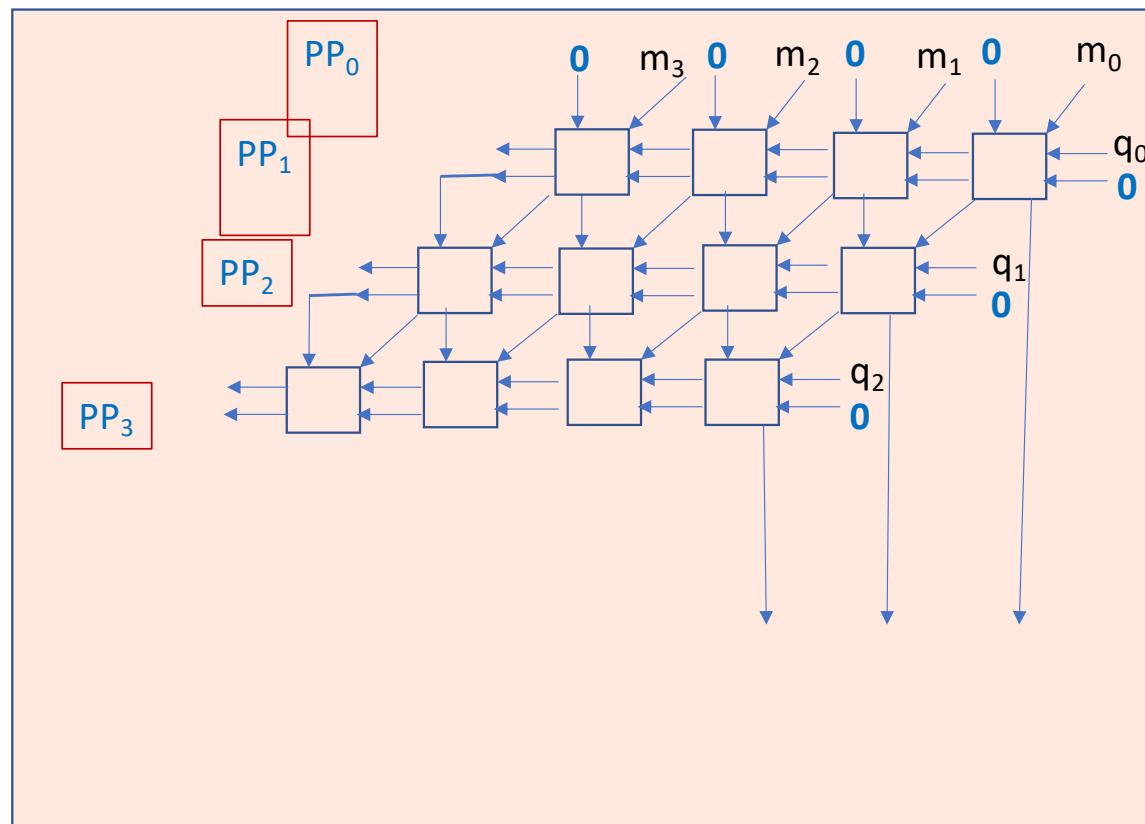
1 1 0 1	(13) Multiplicand, M
X 1 0 1 1	(11) Multiplier, Q
-----	
0 0 0 0	Initial Partial Product, PP <sub>0</sub>
1 1 0 1	Partial Product, PP <sub>1</sub>
-----	
1 1 0 1	Partial Product, PP <sub>2</sub>
1 1 0 1	
-----	
1 0 0 1 1 1	Partial Product, PP <sub>3</sub>
0 0 0 0	
-----	
1 0 0 1 1 1	Partial Product, PP <sub>3</sub>



# MULTIPLICATION - 1

## Array Multiplication

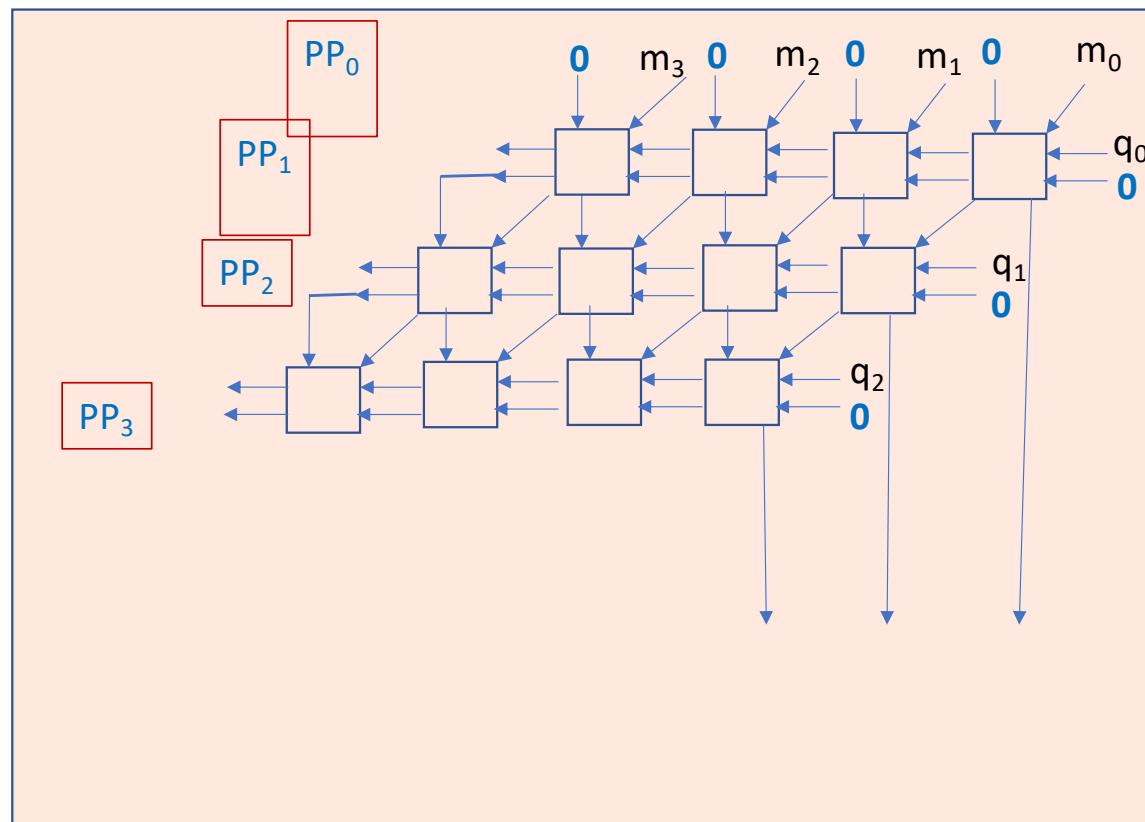
1 1 0 1	(13) Multiplicand, M
X 1 0 1 1	(11) Multiplier, Q
-----	
0 0 0 0	Initial Partial Product, PP <sub>0</sub>
1 1 0 1	Partial Product, PP <sub>1</sub>
-----	
1 1 0 1	Partial Product, PP <sub>2</sub>
1 1 0 1	
-----	
1 0 0 1 1 1	Partial Product, PP <sub>3</sub>
0 0 0 0	
-----	
1 0 0 1 1 1	Partial Product, PP <sub>3</sub>



# MULTIPLICATION - 1

## Array Multiplication

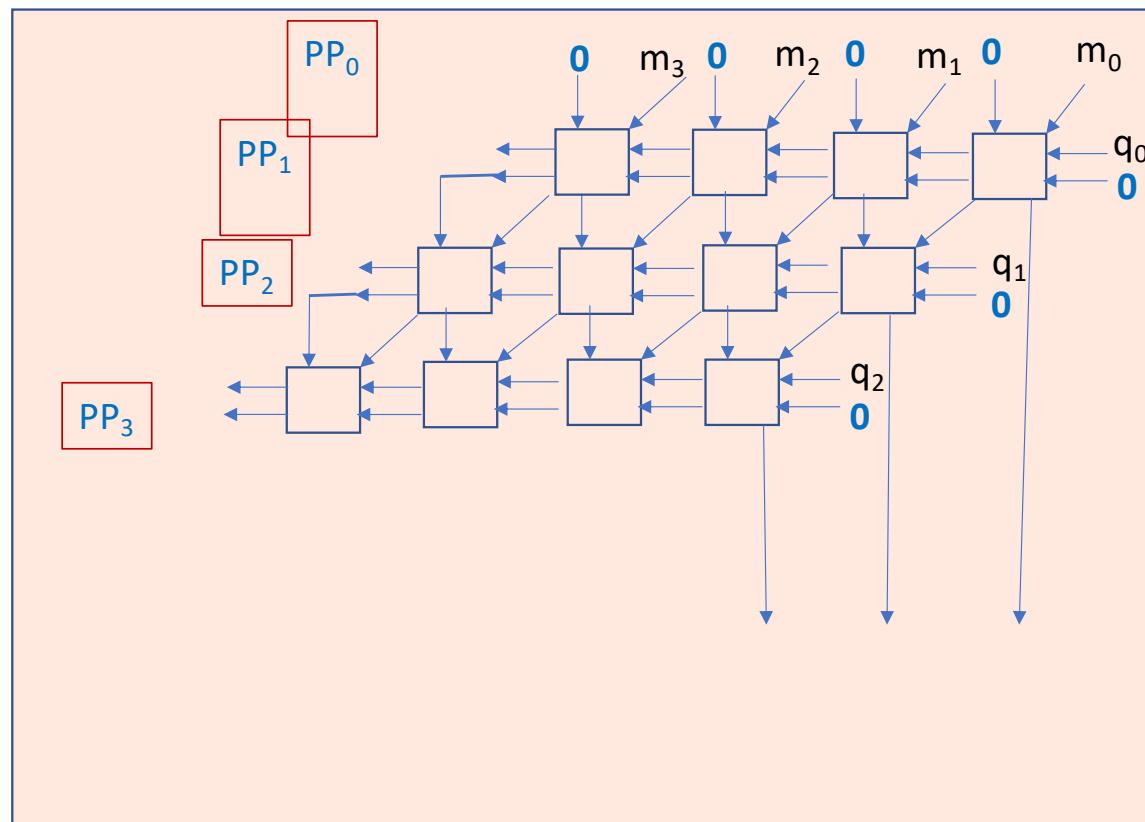
1 1 0 1	(13) Multiplicand, M
X 1 0 1 1	(11) Multiplier, Q
-----	
0 0 0 0	Initial Partial Product, PP <sub>0</sub>
1 1 0 1	
-----	
1 1 0 1	Partial Product, PP <sub>1</sub>
1 1 0 1	
-----	
1 0 0 1 1 1	Partial Product, PP <sub>2</sub>
0 0 0 0	
-----	
1 0 0 1 1 1	Partial Product, PP <sub>3</sub>
1 1 0 1	



# MULTIPLICATION - 1

## Array Multiplication

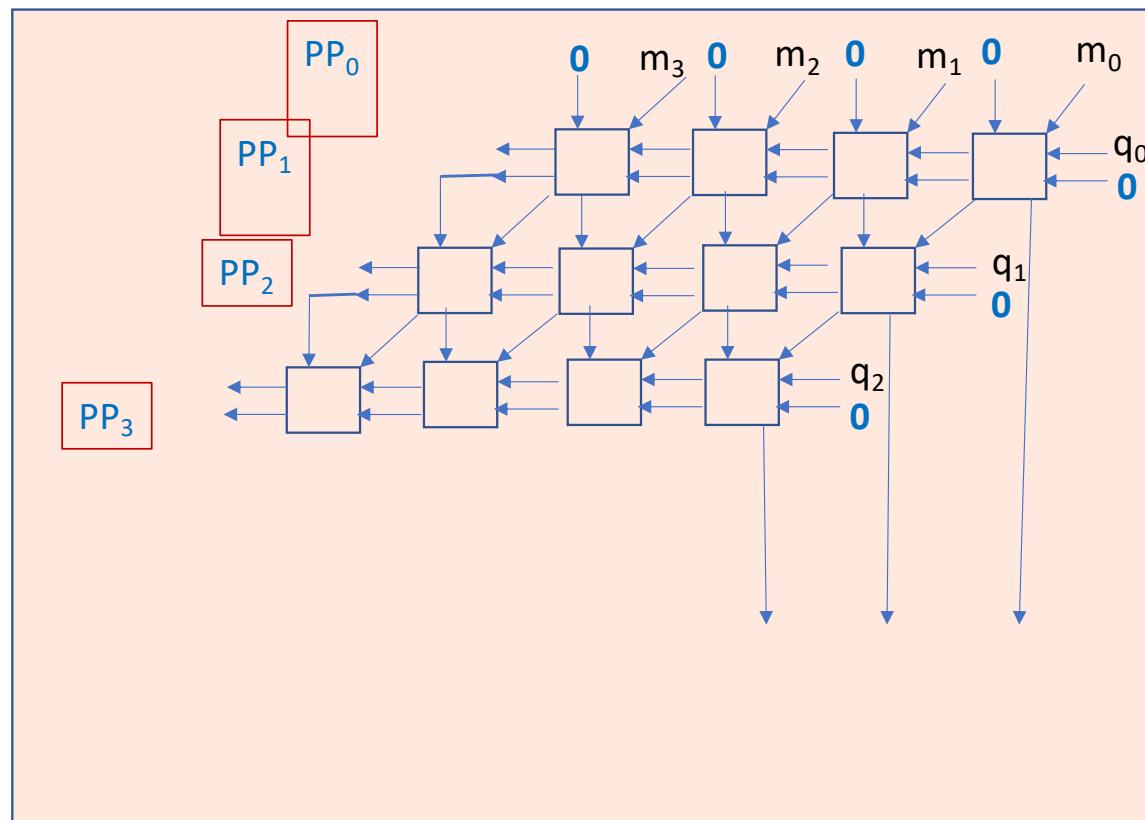
1 1 0 1	(13) Multiplicand, M
X 1 0 1 1	(11) Multiplier, Q
-----	
0 0 0 0	Initial Partial Product, PP <sub>0</sub>
1 1 0 1	
-----	
1 1 0 1	Partial Product, PP <sub>1</sub>
1 1 0 1	
-----	
1 0 0 1 1 1	Partial Product, PP <sub>2</sub>
0 0 0 0	
-----	
1 0 0 1 1 1	Partial Product, PP <sub>3</sub>
1 1 0 1	
-----	



# MULTIPLICATION - 1

## Array Multiplication

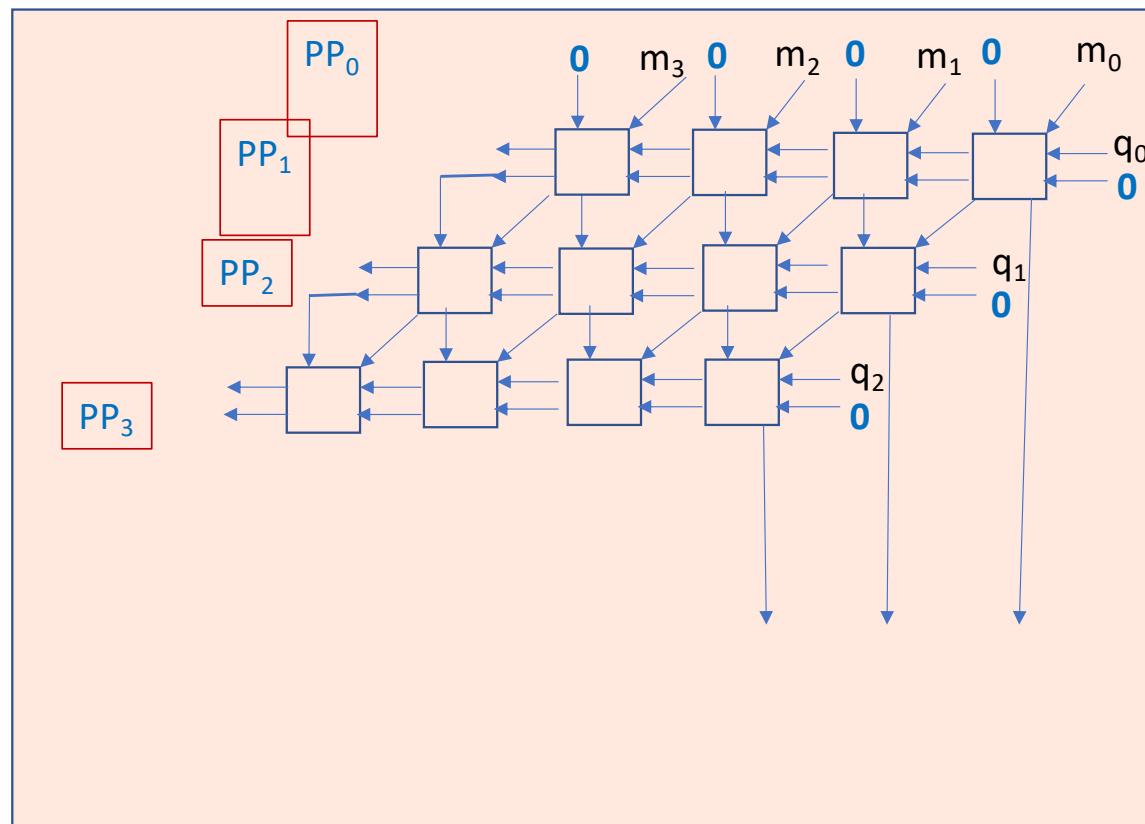
1 1 0 1	(13) Multiplicand, M
X 1 0 1 1	(11) Multiplier, Q
-----	
0 0 0 0	Initial Partial Product, PP <sub>0</sub>
1 1 0 1	Partial Product, PP <sub>1</sub>
-----	
1 1 0 1	Partial Product, PP <sub>2</sub>
1 1 0 1	
-----	
1 0 0 1 1 1	Partial Product, PP <sub>3</sub>
0 0 0 0	
-----	
1 0 0 1 1 1	Partial Product, PP <sub>3</sub>
1 1 0 1	
-----	
1 0 0 0 1 1 1 1	



# MULTIPLICATION - 1

## Array Multiplication

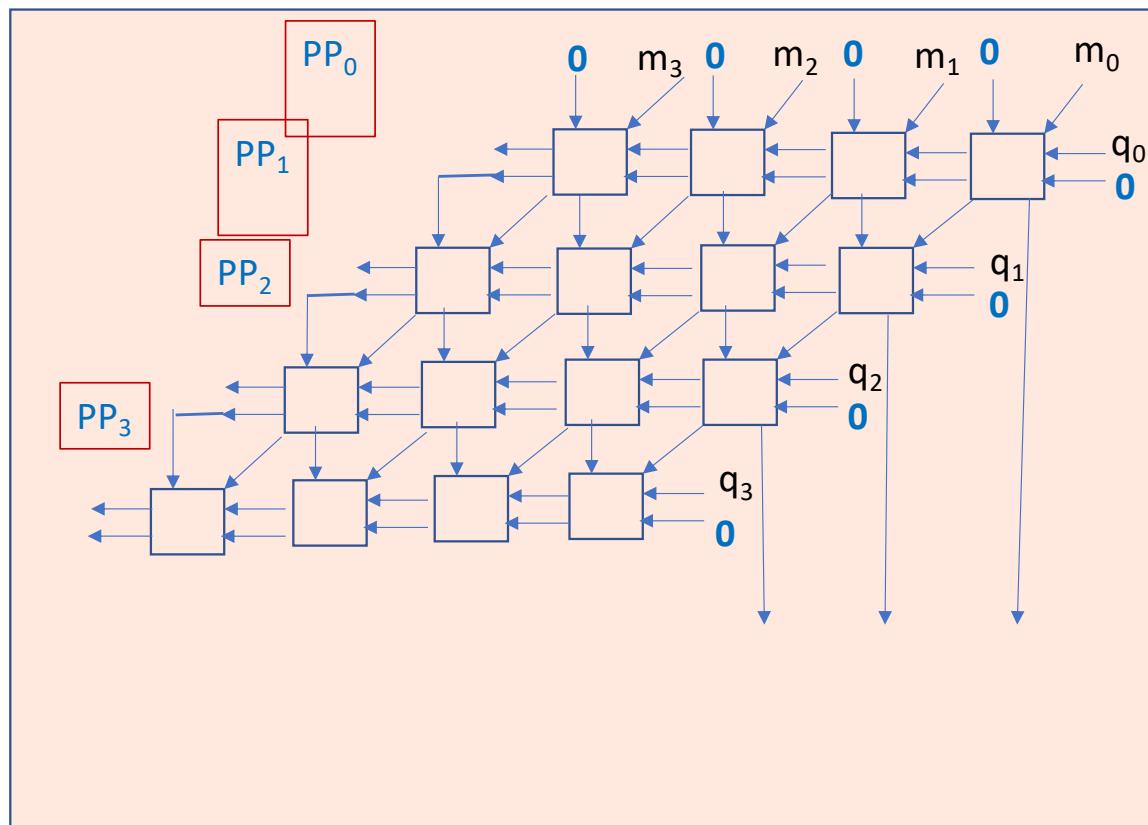
1 1 0 1	(13) Multiplicand, M
X 1 0 1 1	(11) Multiplier, Q
-----	
0 0 0 0	Initial Partial Product, PP <sub>0</sub>
1 1 0 1	
-----	
1 1 0 1	Partial Product, PP <sub>1</sub>
1 1 0 1	
-----	
1 0 0 1 1 1	Partial Product, PP <sub>2</sub>
0 0 0 0	
-----	
1 0 0 1 1 1	Partial Product, PP <sub>3</sub>
1 1 0 1	
-----	
1 0 0 0 1 1 1 1	(143) Product, P



# MULTIPLICATION - 1

## Array Multiplication

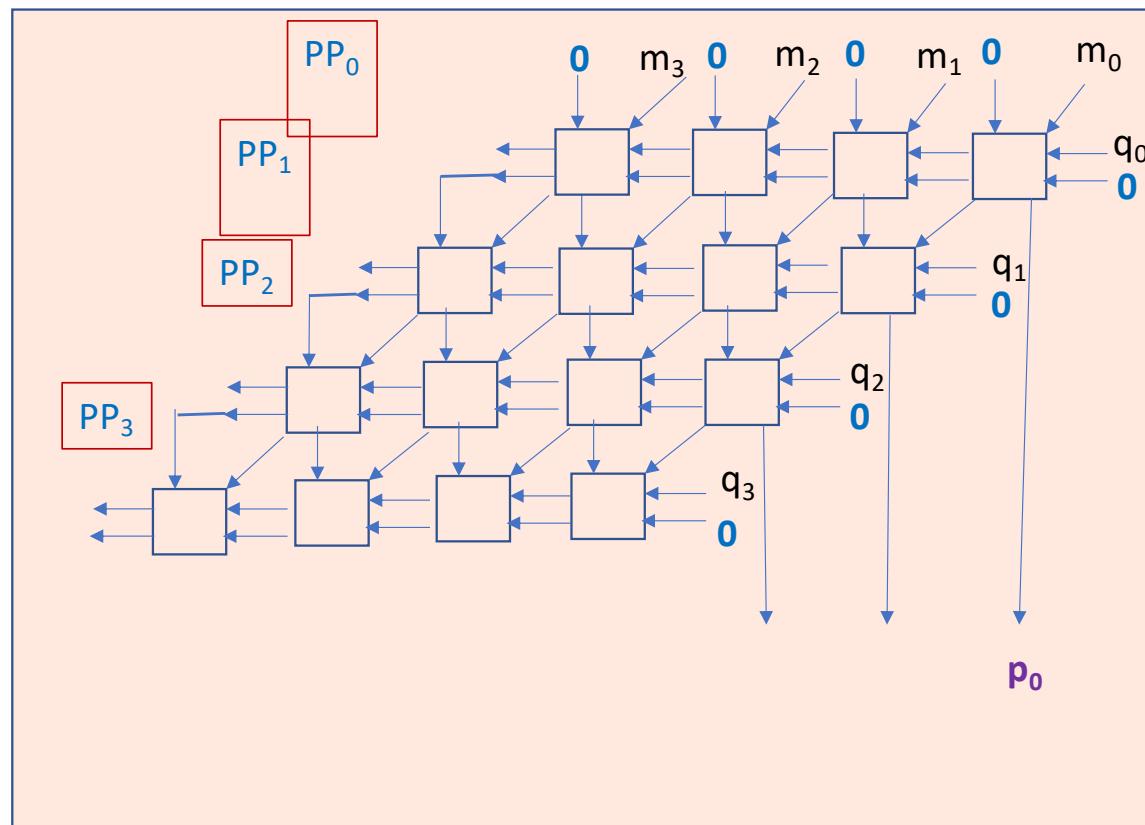
1 1 0 1	(13) Multiplicand, M
X 1 0 1 1	(11) Multiplier, Q
-----	
0 0 0 0	Initial Partial Product, PP <sub>0</sub>
1 1 0 1	Partial Product, PP <sub>1</sub>
-----	
1 1 0 1	Partial Product, PP <sub>2</sub>
1 1 0 1	
-----	
1 0 0 1 1 1	Partial Product, PP <sub>3</sub>
0 0 0 0	
-----	
1 0 0 1 1 1	Partial Product, PP <sub>3</sub>
1 1 0 1	
-----	
1 0 0 0 1 1 1 1	(143) Product, P



# MULTIPLICATION - 1

## Array Multiplication

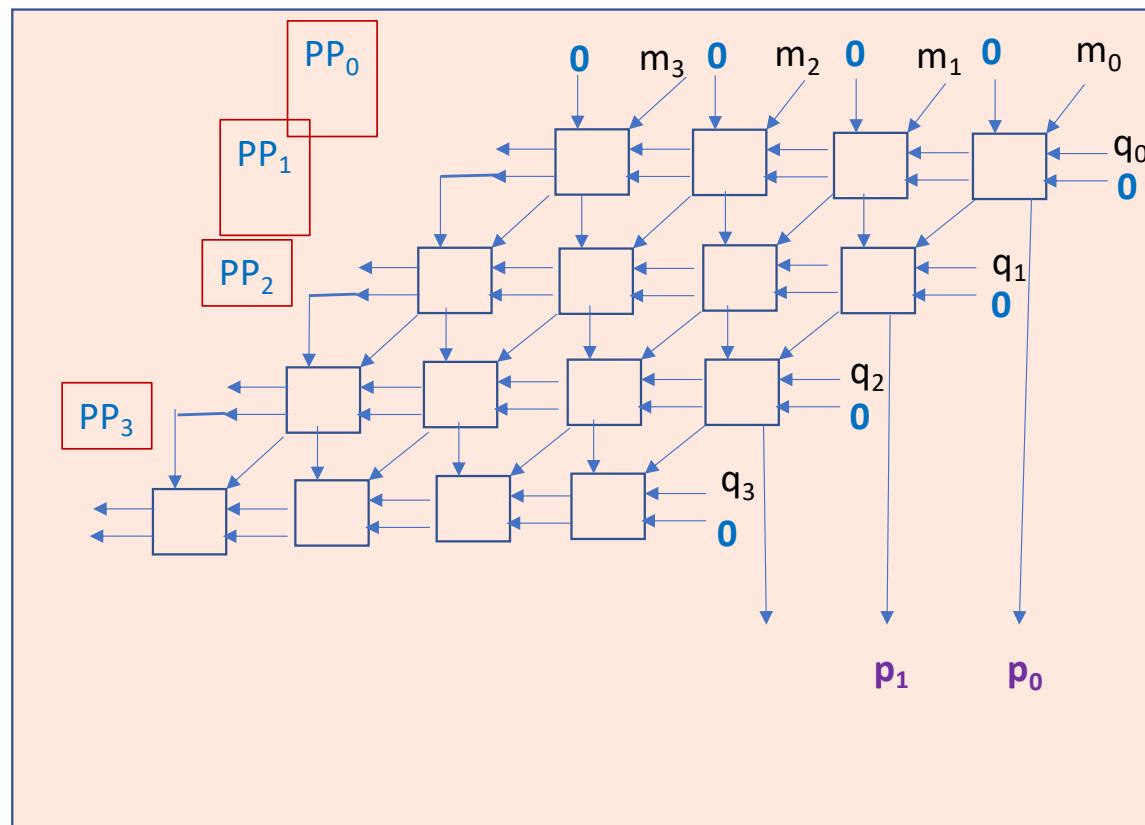
1 1 0 1	(13) Multiplicand, M
X 1 0 1 1	(11) Multiplier, Q
-----	
0 0 0 0	Initial Partial Product, PP <sub>0</sub>
1 1 0 1	Partial Product, PP <sub>1</sub>
-----	
1 1 0 1	Partial Product, PP <sub>2</sub>
1 1 0 1	
-----	
1 0 0 1 1 1	Partial Product, PP <sub>3</sub>
0 0 0 0	
-----	
1 0 0 1 1 1	Partial Product, PP <sub>3</sub>
1 1 0 1	
-----	
1 0 0 0 1 1 1 1	(143) Product, P



# MULTIPLICATION - 1

## Array Multiplication

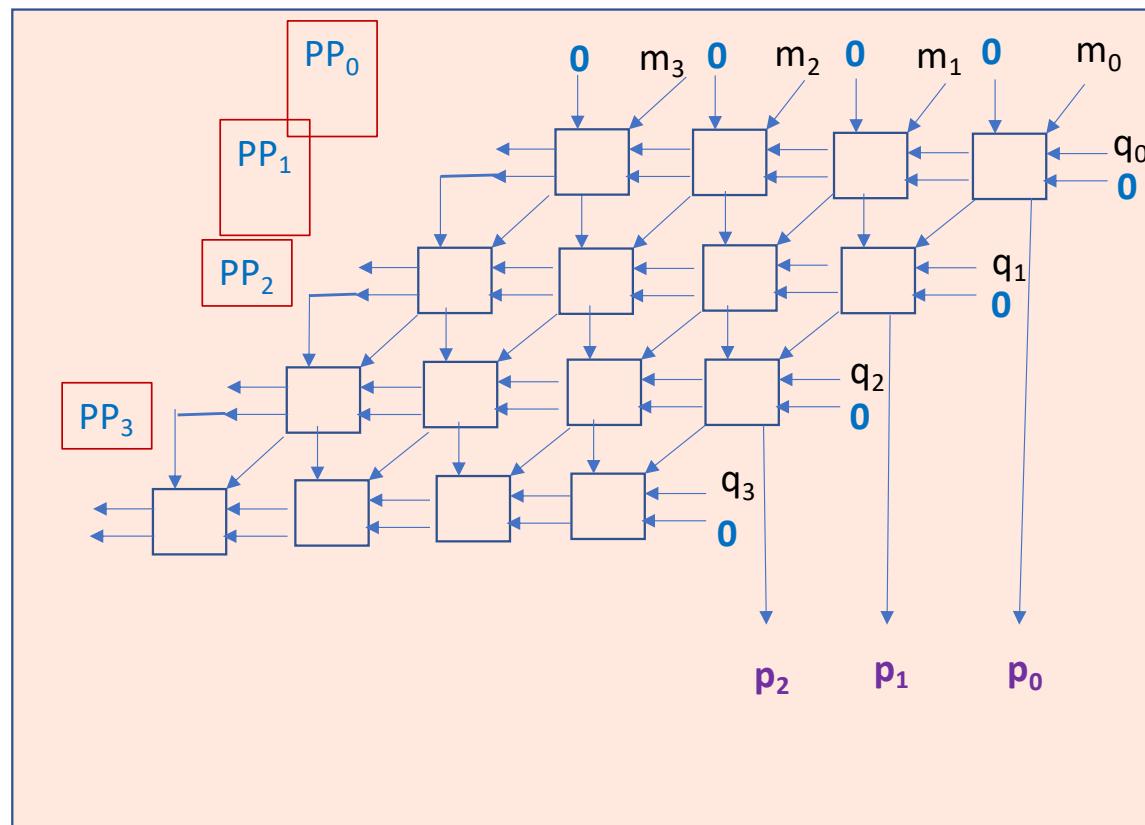
1 1 0 1	(13) Multiplicand, M
X 1 0 1 1	(11) Multiplier, Q
-----	
0 0 0 0	Initial Partial Product, PP <sub>0</sub>
1 1 0 1	Partial Product, PP <sub>1</sub>
-----	
1 1 0 1	Partial Product, PP <sub>2</sub>
1 1 0 1	
-----	
1 0 0 1 1 1	Partial Product, PP <sub>3</sub>
0 0 0 0	
-----	
1 0 0 1 1 1	Partial Product, PP <sub>3</sub>
1 1 0 1	
-----	
1 0 0 0 1 1 1 1	(143) Product, P



# MULTIPLICATION - 1

## Array Multiplication

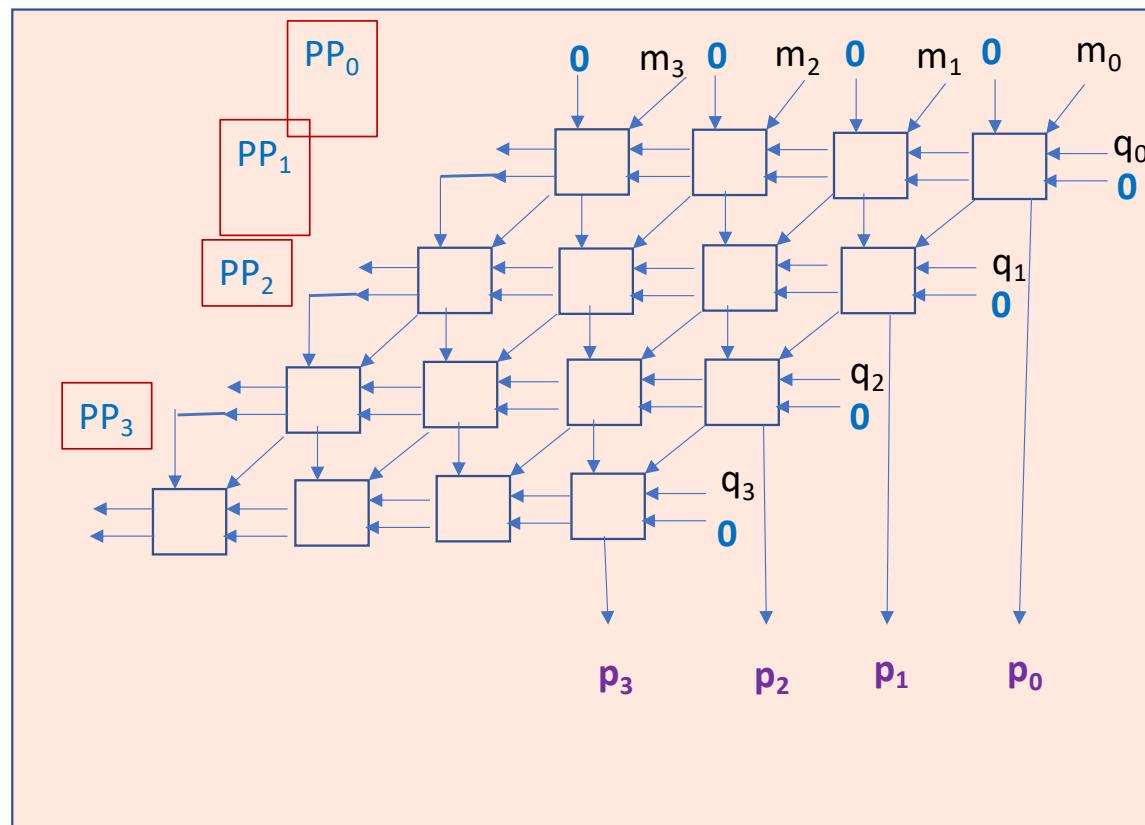
1 1 0 1	(13) Multiplicand, M
X 1 0 1 1	(11) Multiplier, Q
-----	
0 0 0 0	Initial Partial Product, PP <sub>0</sub>
1 1 0 1	Partial Product, PP <sub>1</sub>
-----	
1 1 0 1	Partial Product, PP <sub>2</sub>
1 1 0 1	
-----	
1 0 0 1 1 1	Partial Product, PP <sub>3</sub>
0 0 0 0	
-----	
1 0 0 1 1 1	Partial Product, PP <sub>3</sub>
1 1 0 1	
-----	
1 0 0 0 1 1 1 1	(143) Product, P



# MULTIPLICATION - 1

## Array Multiplication

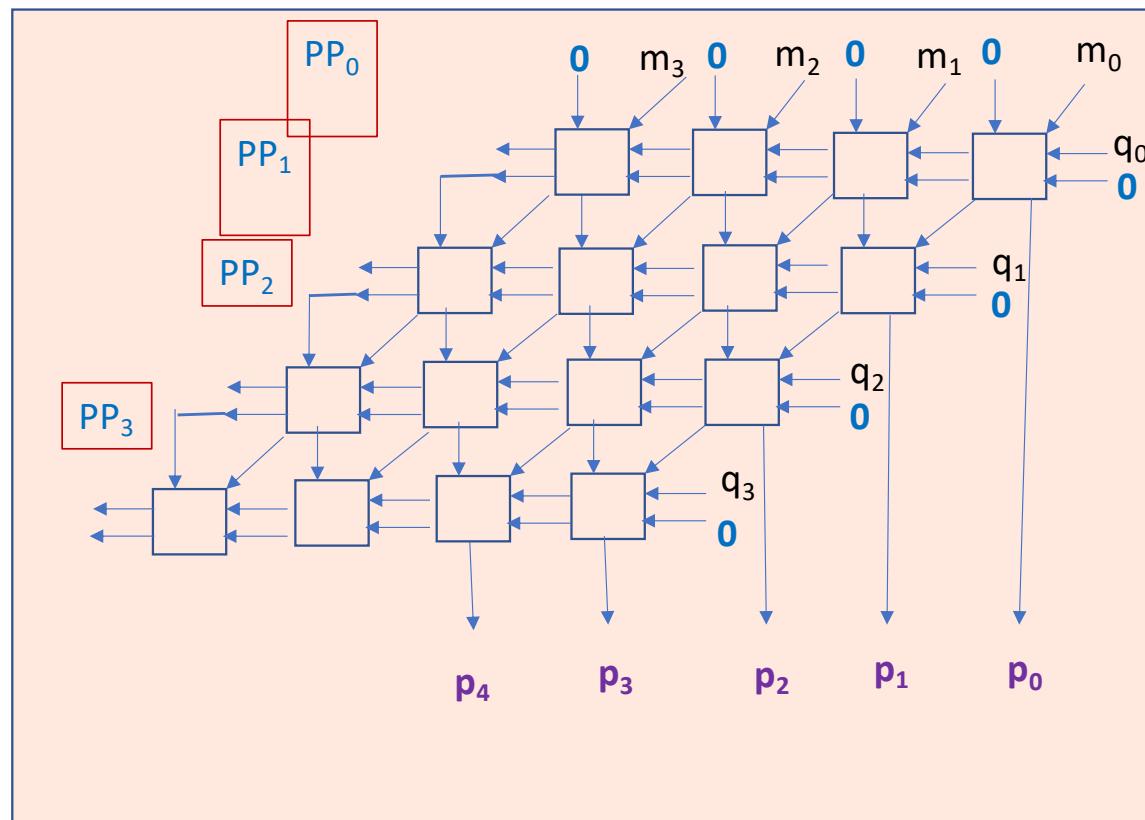
1 1 0 1	(13) Multiplicand, M
X 1 0 1 1	(11) Multiplier, Q
-----	
0 0 0 0	Initial Partial Product, PP <sub>0</sub>
1 1 0 1	Partial Product, PP <sub>1</sub>
-----	
1 1 0 1	Partial Product, PP <sub>2</sub>
1 1 0 1	
-----	
1 0 0 1 1 1	Partial Product, PP <sub>3</sub>
0 0 0 0	
-----	
1 0 0 1 1 1	Partial Product, PP <sub>3</sub>
1 1 0 1	
-----	
1 0 0 0 1 1 1 1	(143) Product, P



# MULTIPLICATION - 1

## Array Multiplication

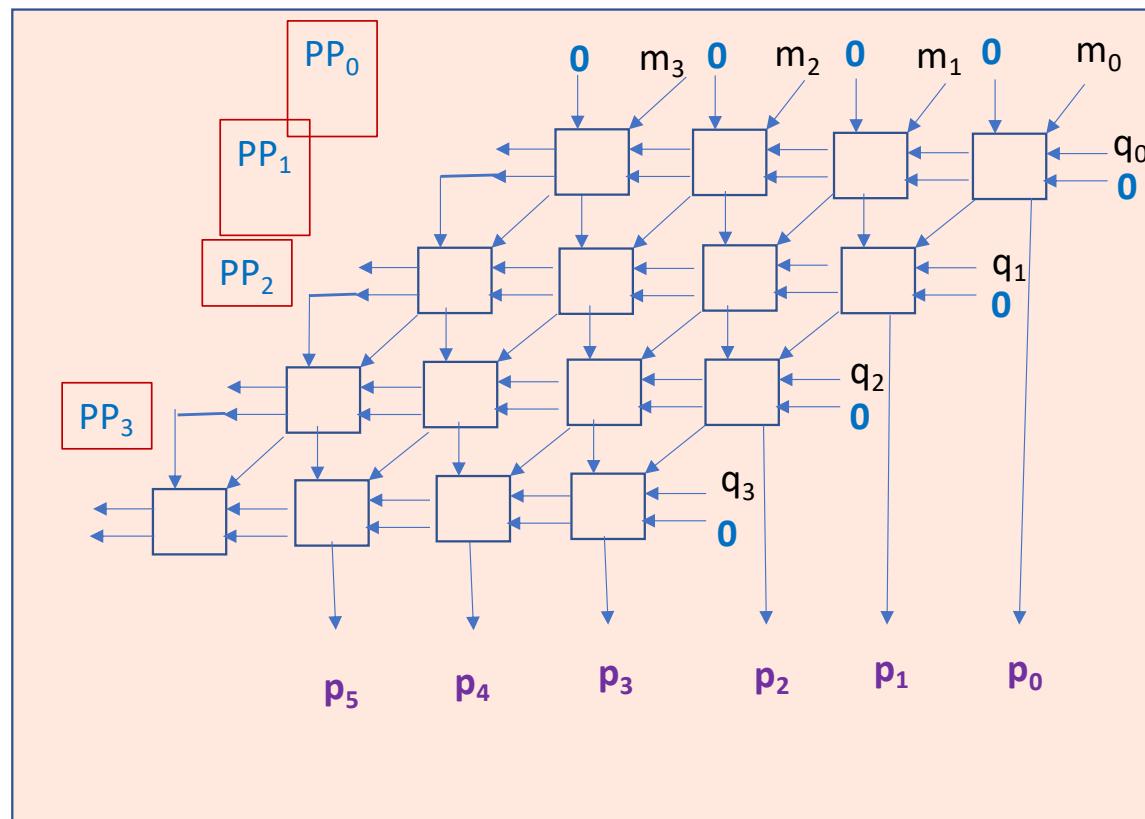
1 1 0 1	(13) Multiplicand, M
X 1 0 1 1	(11) Multiplier, Q
-----	
0 0 0 0	Initial Partial Product, PP <sub>0</sub>
1 1 0 1	Partial Product, PP <sub>1</sub>
-----	
1 1 0 1	Partial Product, PP <sub>2</sub>
1 1 0 1	
-----	
1 0 0 1 1 1	Partial Product, PP <sub>3</sub>
0 0 0 0	
-----	
1 0 0 1 1 1	Partial Product, PP <sub>3</sub>
1 1 0 1	
-----	
1 0 0 0 1 1 1 1	(143) Product, P



# MULTIPLICATION - 1

## Array Multiplication

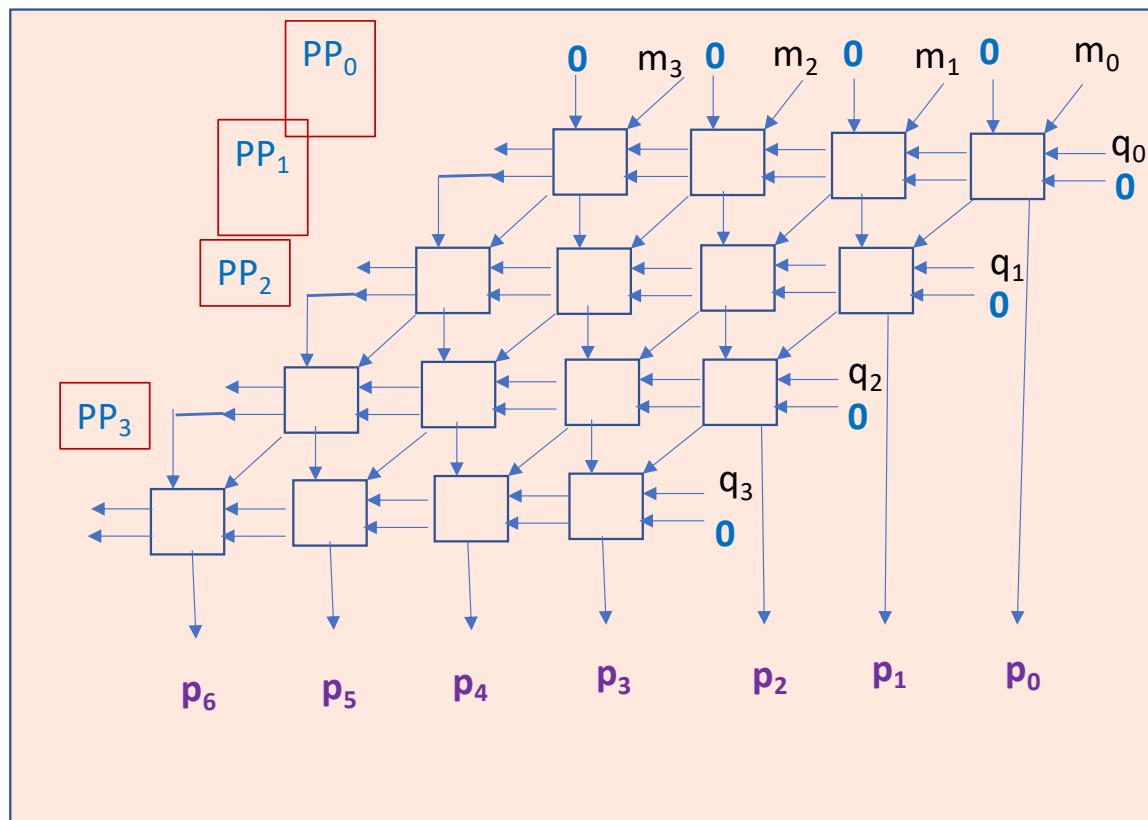
1 1 0 1	(13) Multiplicand, M
X 1 0 1 1	(11) Multiplier, Q
-----	
0 0 0 0	Initial Partial Product, PP <sub>0</sub>
1 1 0 1	Partial Product, PP <sub>1</sub>
-----	
1 1 0 1	Partial Product, PP <sub>2</sub>
1 1 0 1	
-----	
1 0 0 1 1 1	Partial Product, PP <sub>3</sub>
0 0 0 0	
-----	
1 0 0 1 1 1	Partial Product, PP <sub>3</sub>
1 1 0 1	
-----	
1 0 0 0 1 1 1 1	(143) Product, P



# MULTIPLICATION - 1

## Array Multiplication

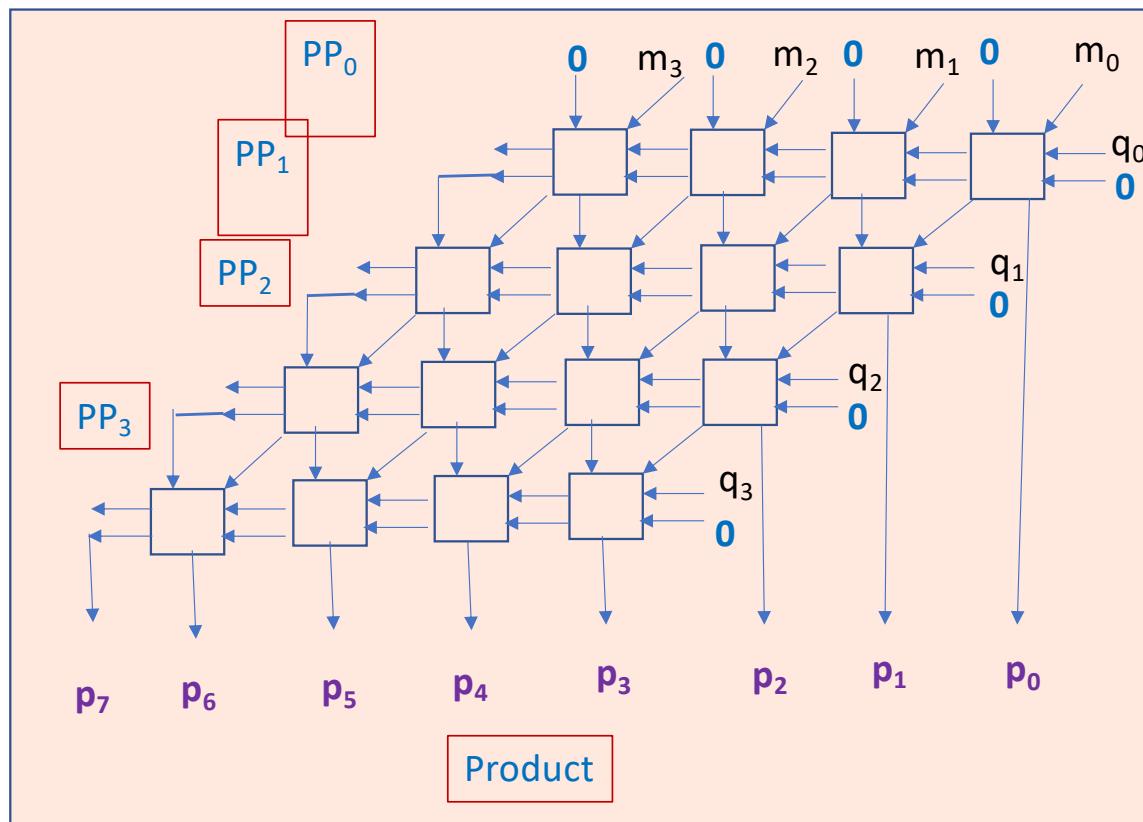
	1 1 0 1	(13) Multiplicand, M
X	1 0 1 1	(11) Multiplier, Q
-----		
	0 0 0 0	Initial Partial Product, PP <sub>0</sub>
	1 1 0 1	
-----		
	1 1 0 1	Partial Product, PP <sub>1</sub>
	1 1 0 1	
-----		
	1 0 0 1 1 1	Partial Product, PP <sub>2</sub>
	0 0 0 0	
-----		
	1 0 0 1 1 1	Partial Product, PP <sub>3</sub>
	1 1 0 1	
-----		
	1 0 0 0 1 1 1 1	(143) Product, P



# MULTIPLICATION - 1

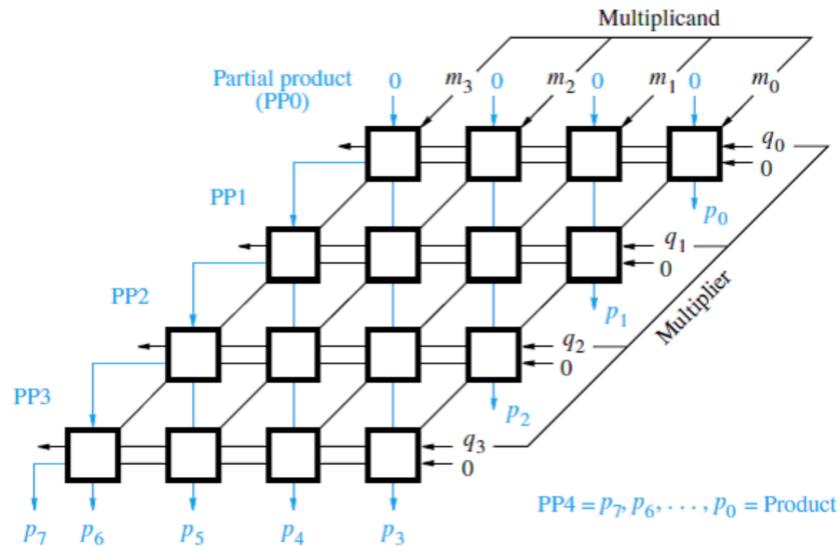
## Array Multiplication

	1 1 0 1	(13) Multiplicand, M
X	1 0 1 1	(11) Multiplier, Q
-----		
	0 0 0 0	Initial Partial Product, PP <sub>0</sub>
	1 1 0 1	
-----		
	1 1 0 1	Partial Product, PP <sub>1</sub>
	1 1 0 1	
-----		
	1 0 0 1 1 1	Partial Product, PP <sub>2</sub>
	0 0 0 0	
-----		
	1 0 0 1 1 1	Partial Product, PP <sub>3</sub>
	1 1 0 1	
-----		
	1 0 0 0 1 1 1 1	(143) Product, P



# MULTIPLICATION - 1

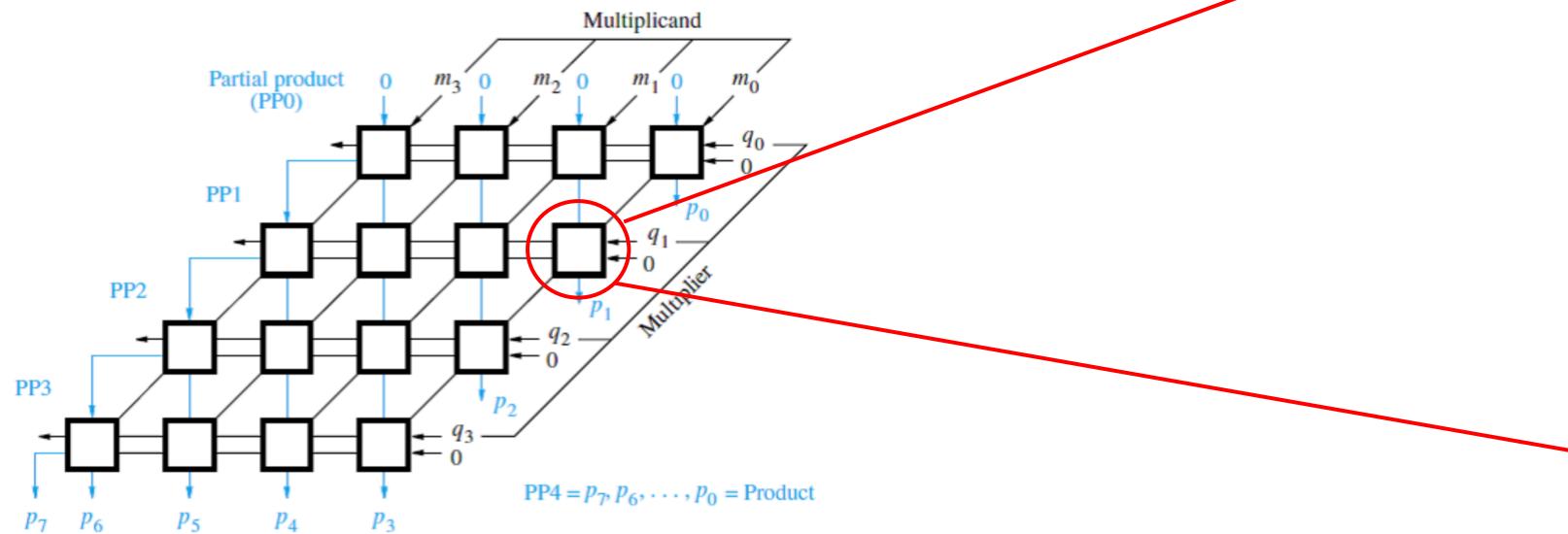
## Array Multiplication



- AND gate in each cell performs  $m_j \cdot q_i$  product
- A Full Adder adds this with the previous partial product bit  $PP_i$  along with previous bit Carry-in and generates  $PP_{(i+1)}$  and Carry-out.
- $m_j$  is shifted to downward cell whereas  $q_i$  is connected to all the cells in a row.

# MULTIPLICATION - 1

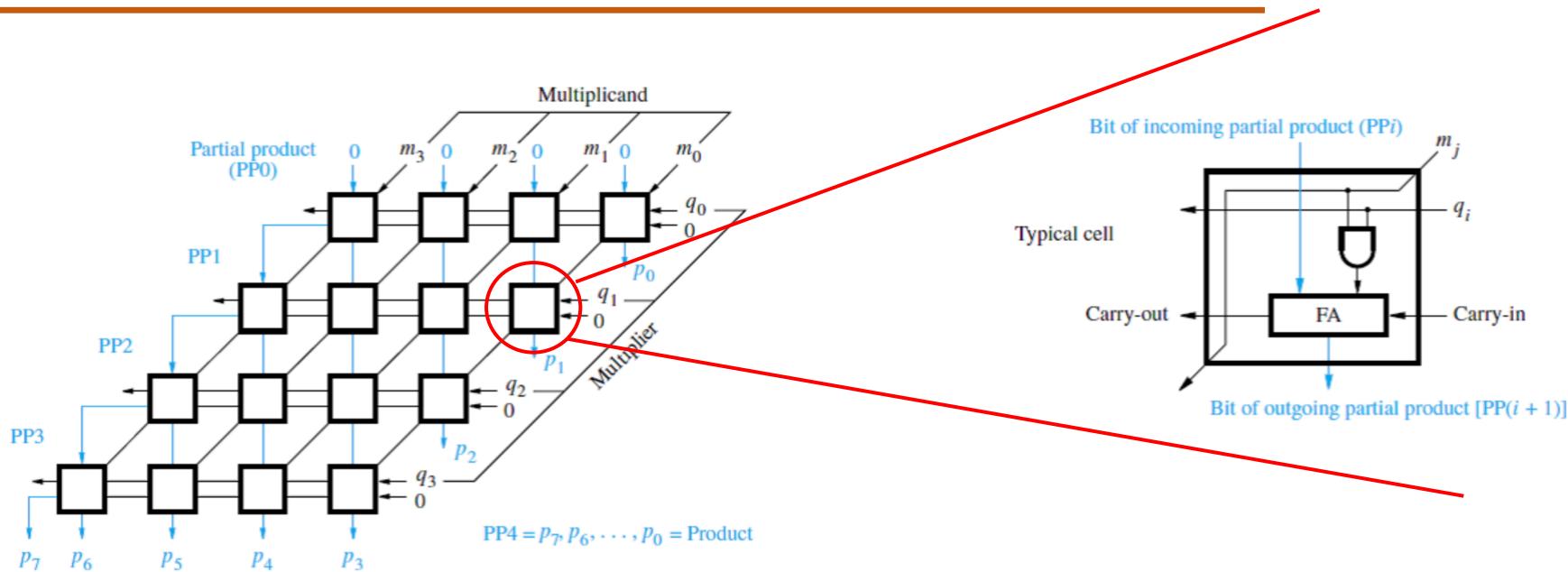
## Array Multiplication



- AND gate in each cell performs  $m_j \cdot q_i$  product
- A Full Adder adds this with the previous partial product bit  $PP_i$  along with previous bit Carry-in and generates  $PP_{(i+1)}$  and Carry-out.
- $m_j$  is shifted to downward cell whereas  $q_i$  is connected to all the cells in a row.

# MULTIPLICATION - 1

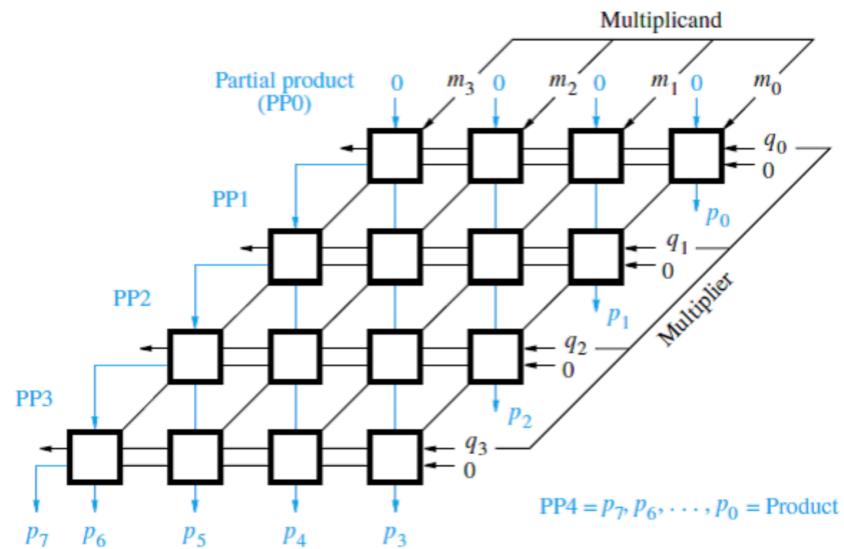
## Array Multiplication



- AND gate in each cell performs  $m_j \cdot q_i$  product
- A Full Adder adds this with the previous partial product bit  $PP_i$  along with previous bit Carry-in and generates  $PP_{(i+1)}$  and Carry-out.
- $m_j$  is shifted to downward cell whereas  $q_i$  is connected to all the cells in a row.

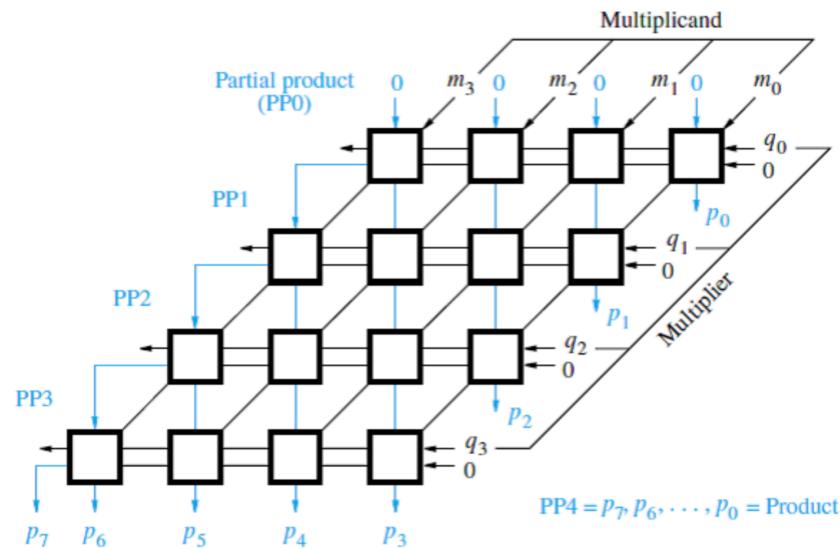
## MULTIPLICATION - 1

Think about it



# MULTIPLICATION - 1

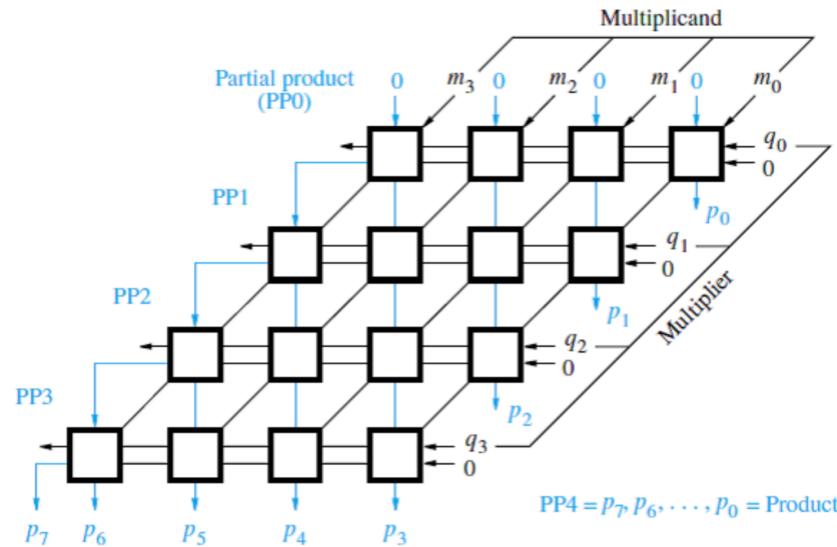
Think about it



- What is the worst case propagation delay?

# MULTIPLICATION - 1

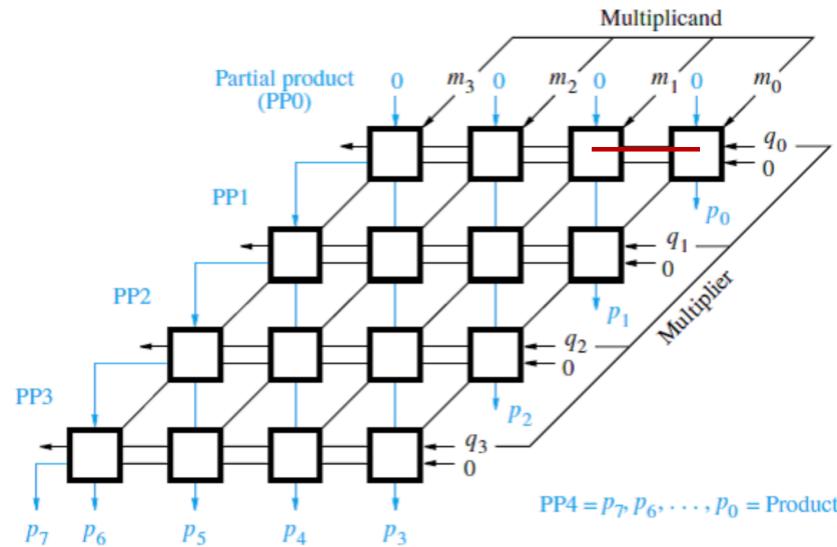
Think about it



- What is the worst case propagation delay?
- Assume FA has 2 gate delays and all AND gates has 1 gate delay. Also, the first row can have only AND gates.

# MULTIPLICATION - 1

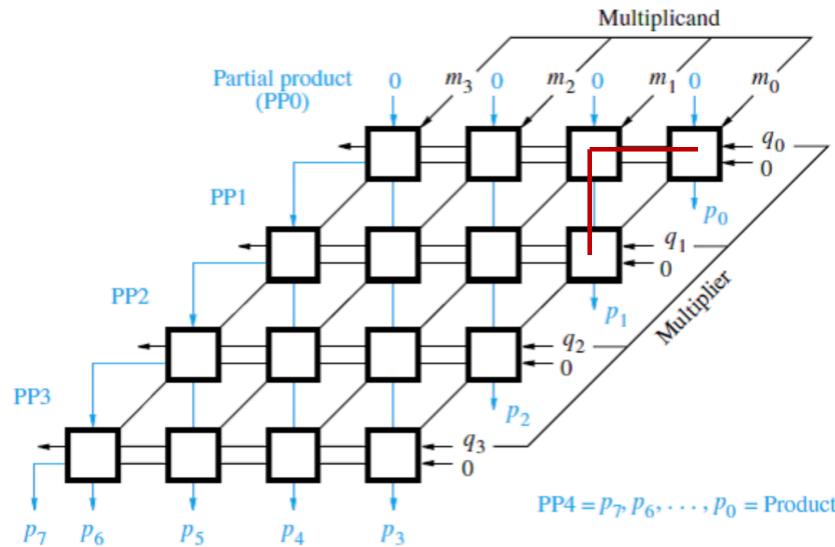
Think about it



- What is the worst case propagation delay?
- Assume FA has 2 gate delays and all AND gates has 1 gate delay. Also, the first row can have only AND gates.

# MULTIPLICATION - 1

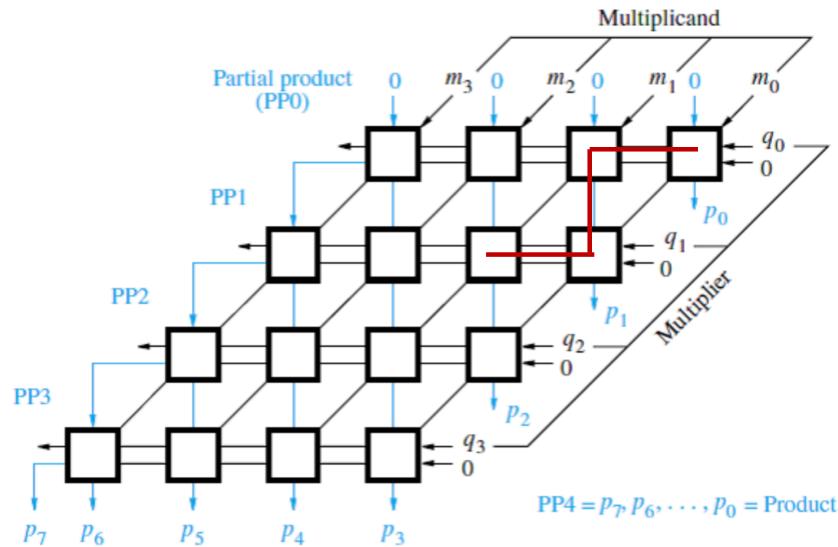
Think about it



- What is the worst case propagation delay?
- Assume FA has 2 gate delays and all AND gates has 1 gate delay. Also, the first row can have only AND gates.

# MULTIPLICATION - 1

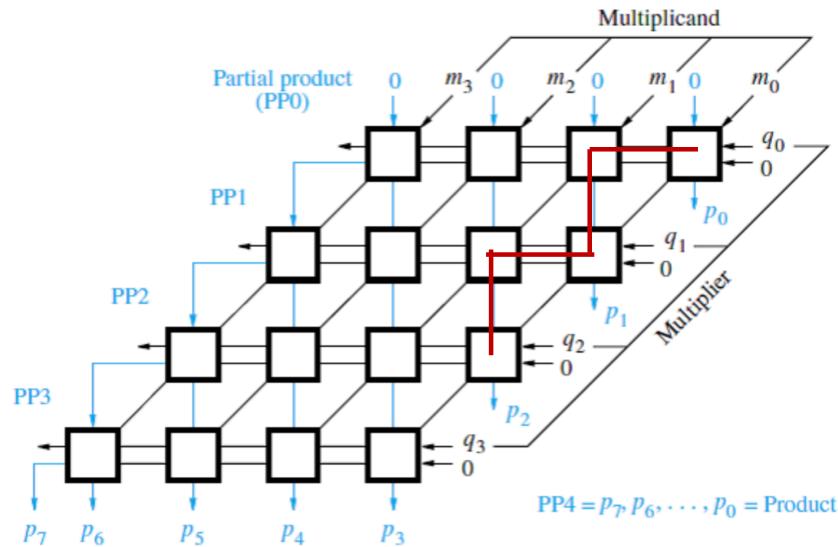
Think about it



- What is the worst case propagation delay?
- Assume FA has 2 gate delays and all AND gates has 1 gate delay. Also, the first row can have only AND gates.

# MULTIPLICATION - 1

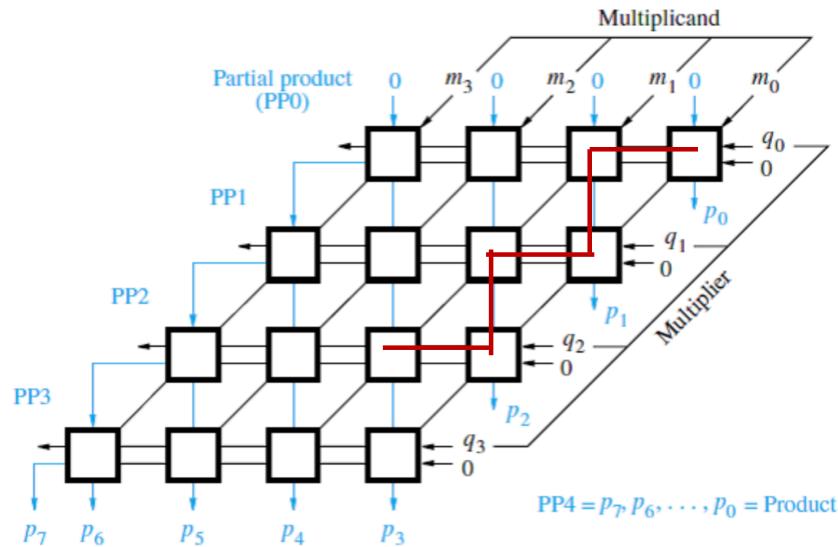
Think about it



- What is the worst case propagation delay?
- Assume FA has 2 gate delays and all AND gates has 1 gate delay. Also, the first row can have only AND gates.

# MULTIPLICATION - 1

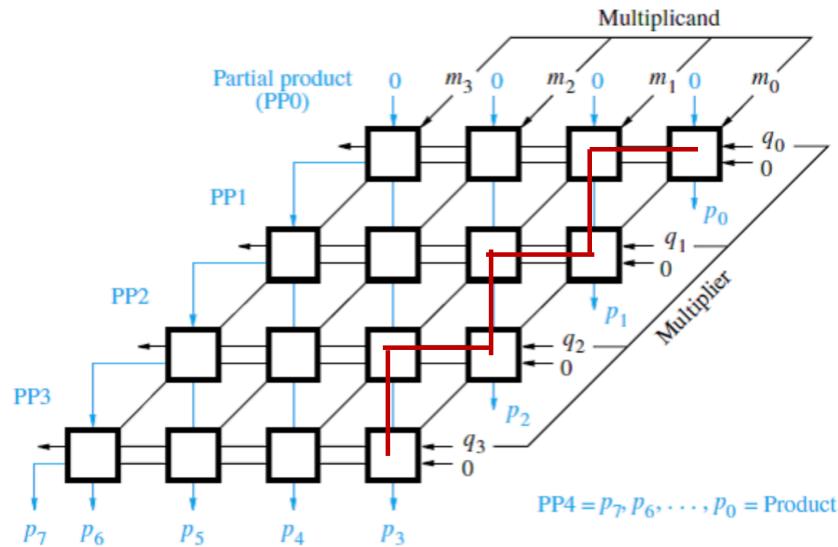
Think about it



- What is the worst case propagation delay?
- Assume FA has 2 gate delays and all AND gates has 1 gate delay. Also, the first row can have only AND gates.

# MULTIPLICATION - 1

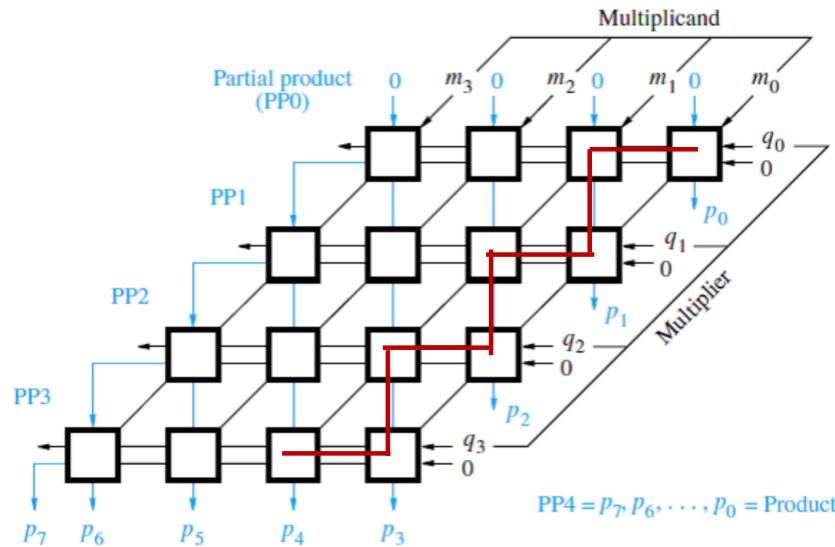
Think about it



- What is the worst case propagation delay?
- Assume FA has 2 gate delays and all AND gates has 1 gate delay. Also, the first row can have only AND gates.

# MULTIPLICATION - 1

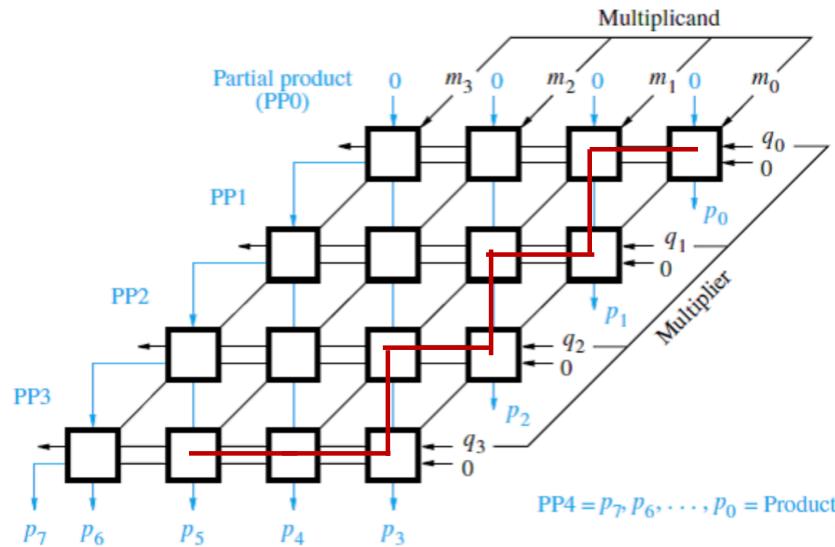
Think about it



- What is the worst case propagation delay?
- Assume FA has 2 gate delays and all AND gates has 1 gate delay. Also, the first row can have only AND gates.

# MULTIPLICATION - 1

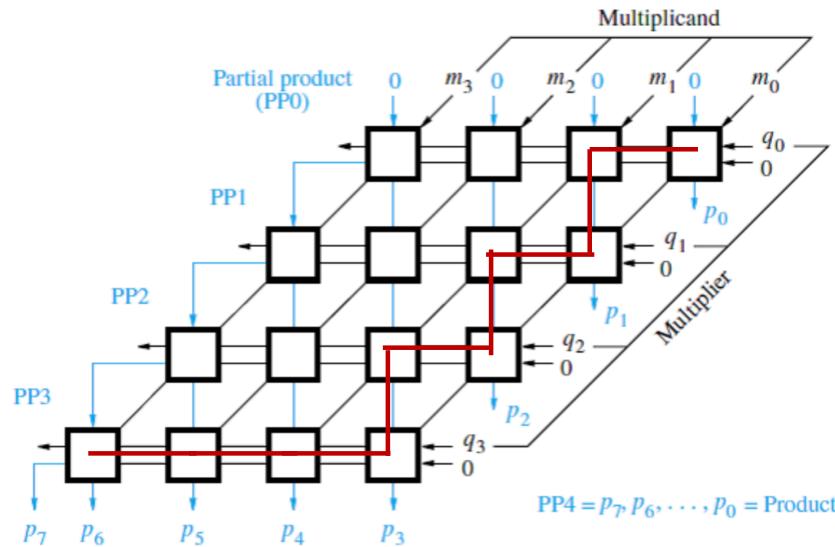
Think about it



- What is the worst case propagation delay?
- Assume FA has 2 gate delays and all AND gates has 1 gate delay. Also, the first row can have only AND gates.

# MULTIPLICATION - 1

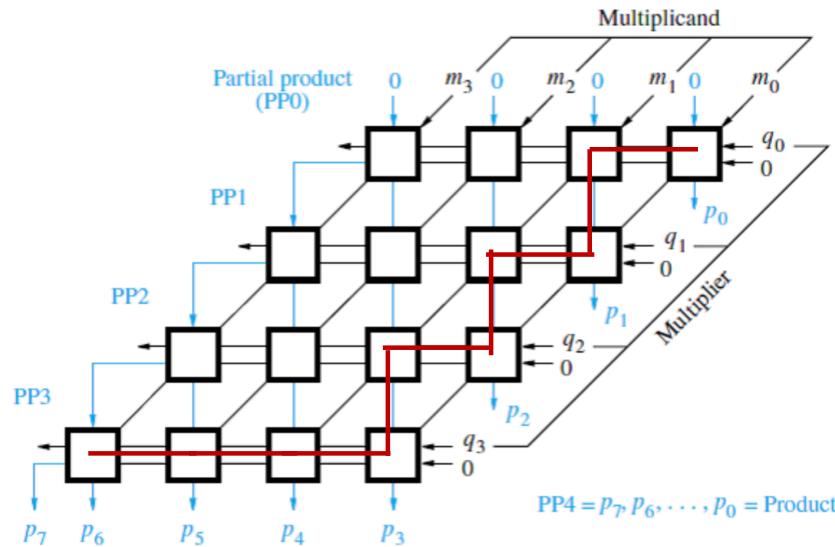
Think about it



- What is the worst case propagation delay?
- Assume FA has 2 gate delays and all AND gates has 1 gate delay. Also, the first row can have only AND gates.

# MULTIPLICATION - 1

Think about it



- What is the worst case propagation delay?
- Assume FA has 2 gate delays and all AND gates has 1 gate delay. Also, the first row can have only AND gates.
- How many gates are required for  $m \times n$  array multiplication?



**THANK YOU**

---

**Sudarshan T S B. Ph.D.,**  
Department of Computer Science & Engineering  
**sudarshan@pes.edu**  
**+91 80 6666 3333 Extn 215**