A0 A1 A2 A3 A4 A5 A6 A7

B0 B1 B2 B3 B4 B5 B6 B7

Π

A0B7 A1B7 A2B7 A3B7 A4B7 A5B7 A6B7 A7B7

A0B6 A1B6 A2B6 A3B6 A4B6 A5B6 A6B6 A7B6

A0B5 A1B5 A2B5 A3B5 A4B5 A5B5 A6B5 A7B5

A0B4 A1B4 A2B4 A3B4 A4B4 A5B4 A6B4 A7B4

A0B3 A1B3 A2B3 A3B3 A4B3 A5B3 A6B3 A7B7

A0B2 A1B2 A2B2 A3B2 A4B2 A5B2 A6B2 A7B2

A0B1 A1B1 A2B1 A3B1 A4B1 A5B1 A6B1 A7B7

A0B0 A1B0 A2B0 A3B0 A4B0 A5B0 A6B0 A7B0

VAI 1 VAI 1 VAI 1 VAI 1 VAI 1 VAI 1 VAI 1 VAI 1 VAI 1 VAI 1 VAI 1 VAI 1 VAI 1 VAI 1

Σ

Y0 Y1 Y2 Y3 Y4 Y5 Y6 Y7 Y8 Y9 Y10 Y11 Y12 Y13 Y14 Y15

Por serem 14 saídas, devemos usar um display hexadecimal como saída para mostrar os valores numéricos da multiplicação.

0x0 = 0 0+0 = 0 ; VAI 1 = 0

0x1 = 0 0+1 = 1 ; VAI 1 = 0 VAI 1 pode ser implementado por uma porta AND

1x0 = 0 1+0 = 1 ; VAI 1 = 0 Já a soma é feita por uma XOR

1x1 = 1 1+1 = 0 ; VAI 1 = 1

É fácil a observação sobre a melhor porta lógica a ser usada para a obtenção dos valores AiBj; 0 ≤ i ≤ 7; 0 ≤ j ≤ 7 .

Devemos observar que para a obtenção das saídas Yn (0 ≤ n ≤ 15) é necessário que façamos o somatório entre as respectivas saídas AiBj e o elemento “VAI 1” obtido em cada soma anterior - devemos construir um somador 2bits e implementá-lo no projeto do multiplicador.