

# General 8bit \* 8bit = 8bit multiply

```
; General 8bit * 8bit = 8bit multiply
; by White Flame 20030207

; Multiplies "num1" by "num2" and returns result in .A
; Instead of using a bit counter, this routine early-exits when num2 reaches
zero, thus saving iterations.

; Input variables:
; num1 (multiplicand)
; num2 (multiplier), should be small for speed
; Signedness should not matter

; .X and .Y are preserved
; num1 and num2 get clobbered

lda #$00
beq enterLoop

doAdd:
  clc
  adc num1

loop:
  asl num1
enterLoop: ;For an accumulating multiply (.A = .A + num1*num2), set up num1
and num2, then enter here
  lsr num2
  bcs doAdd
  bne loop

end:

; 15 bytes
```

From:

<https://codebase64.org/> - **Codebase 64 wiki**

Permanent link:

[https://codebase64.org/doku.php?id=base:8bit\\_multiplication\\_8bit\\_product](https://codebase64.org/doku.php?id=base:8bit_multiplication_8bit_product)

Last update: **2017-10-26 06:59**

