

Lab 3

Basic

▼ Answer

```
1 List p = 18f4520
2 #include<p18f4520.inc>
3 CONFIG OSC = INTIO67
4 CONFIG WDT = OFF
5 org 0x00
6
7     CLRF TRISA
8     MOVLW b'01011111' ;
9     MOVWF TRISA
10
11     BCF STATUS, C
12     RLCF TRISA, F
13
14     BCF STATUS, C
15     BTFSC TRISA, 7
16     BSF STATUS, C
17     RRCF TRISA, F
18
19 Over:
20     end
```

Advanced

▼ Answer


```
1 List p = 18f4520
2 #include<p18f4520.inc>
3 CONFIG OSC = INTIO67
4 CONFIG WDT = OFF
5 org 0x00
6
7     CLRF 0x020
8     CLRF 0x021
9     CLRF 0x022
10    CLRF 0x023
11
12    MOVLW 0x12
13    MOVWF 0x000 ; a1
14
15    MOVLW 0xCB
16    MOVWF 0x001 ; a0
17
18    MOVLW 0x09
19    MOVWF 0x010 ; b1
20
21    MOVLW 0x35
22    MOVWF 0x011 ; b0
23
24    Multiply:
25        ; Block 1
26        MOVF 0x011, W ; W = 0x011 (b0)
27        MULWF 0x001 ; a0 * b0
28
29        MOVF PRODL, W
30        ADDWF 0x023, F ; Save the low bits of a0 * b0
31        MOVF PRODH, W
32        ADDWF 0x022, F ; Save the high bits of a0 * b0
33
34        ; Block 2
35        MOVF 0x011, W ; W = 0x011 (b0)
36        MULWF 0x000 ; a1 * b0
37
38        MOVF PRODL, W
39        ADDWF 0x022, F ; Save the low bits of a1 * b0
40        BTFSC STATUS, C
41        INCF 0x021
42        MOVF PRODH, W
43        ADDWF 0x021, F ; Save the high bits of a1 * b0
44        BTFSC STATUS, C
45        INCF 0x020
46
47        ; Block 3
48        MOVF 0x010, W ; W = 0x010 (b1)
49        MULWF 0x001 ; a0 * b1
50
51        MOVF PRODL, W
52        ADDWF 0x022, F ; Save the low bits of a0 * b1
53        BTFSC STATUS, C
```

```
54      INCF 0x021
55      BTFSC STATUS, C
56      INCF 0x020
57      MOVF PRODH, W
58      ADDWF 0x021, F ; Save the high bits of a0 * b1
59      BTFSC STATUS, C
60      INCF 0x020
61
62      ; Block 4
63      MOVF 0x010, W ; W = 0x010 (b1)
64      MULWF 0x000 ; a1 * b1
65
66      MOVF PRODL, W
67      ADDWF 0x021, F ; Save the low bits of a1 * b1
68      BTFSC STATUS, C
69      INCF 0x020
70      MOVF PRODH, W
71      ADDWF 0x020, F ; Save the high bits of a0 * b1
72
73      Over:
74      end
```

Bonus

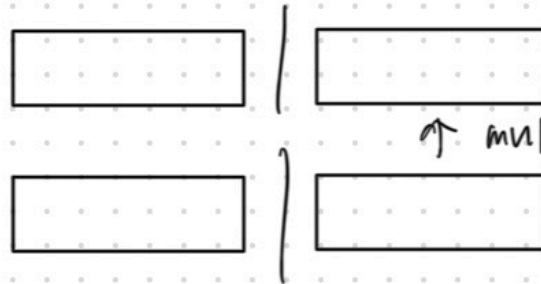
▼ Answer


```
1 List p = 18f4520
2 #include<p18f4520.inc>
3 CONFIG OSC = INTIO67
4 CONFIG WDT = OFF
5 org 0x00
6
7     CLRF 0x002
8     CLRF 0x003
9     MOVLW 0xFF
10    MOVWF 0x000 ; Save the high bits
11    MOVLW 0xF1
12    MOVWF 0x001 ; save the low bits
13
14    MOVLW 0x00
15    CPFSGT 0x000 ; If 0x000 == 0 goto lower
16    GOTO Lower
17    MOVLW 0x08
18    MOVWF 0x002
19    MOVF 0x000, W
20    MOVWF 0x004
21    GOTO Divide_2
22
23 Lower:
24     MOVF 0x001, W
25     MOVWF 0x004
26     GOTO Divide_2
27
28 Divide_2:
29     RRCF 0x004
30     BTFSC STATUS, C
31     GOTO Rounding
32     BCF STATUS, C
33     MOVLW 0x00
34     CPFSGT 0x004 ; If 0x004 == 0 goto Over
35     GOTO Over
36     INCF 0x002
37     GOTO Divide_2
38
39 Rounding:
40     INCF 0x003
41     BCF STATUS, C
42     MOVLW 0x00
43     CPFSGT 0x004 ; If 0x004 == 0 goto Over
44     GOTO Over
45     INCF 0x002
46     GOTO Divide_2
47
48 Over:
49     MOVLW 0x00
50     CPFSEQ 0x001 ; IF 0x001 > 0, 0x003++
51     INCF 0x003
52
53     MOVLW 0x03
```

```

54 CPFSLT 0x003 ; If 0x003 > 2 ,.do rounding
55 INCF 0x002
56
57 CLRF 0x03
58 end

```



	00	01	02	03
000	a_1	a_0	x	x
010	b_1	b_0	x	x
020	Carry + $a_1 b_1 H$	Carry + $a_1 b_0 H$ + $a_0 b_1 H$ + $a_1 b_1 L$	$a_0 b_0 H$ + $a_1 b_0 L$ + $a_0 b_1 L$	$a_0 b_0 L$