

Líneas eliminadas al hacer los cambios

Líneas añadidas al hacer los cambios

```
MOV    IP, SP
STMDB  SPI,{R4-R11,LR}
MOV    FP, IP
LDMIA  FP,{R7-R9}
```

SUB SP, #4 ;Espacio para var posicion_valida

LDR R10,[R1] ;Guarda la variable longitud, se cambiará durante la función
de manera local en registro, y al acabar se guardará en
memoria

```
ADD    R2,R2,R7
MOV    R2,R2,LSL #24
MOV    R5,R2,ASR #24
ADD    R2,R3,R8
MOV    R2,R2,LSL #24
MOV    R6,R2,ASR #24
```

;ficha_valida

```
CMP    R5,#0
CMPGE  R6,#0
BLT    else
CMP    R5,#8
CMPLT  R6,#8
BGE    else
ADD    R2,R0,R5,LSL #3
LDRB   R4,[R2,R6]
CMP    R4,#CASILLA_VACIA
BEQ    else
```

MOV R2,#1

STR R2,[R13] ;Posicion valida = 1

MOV R3,#1 ;Posicion valida = 1

B while

else **MOV R2,#1**

STR R2,[R13] ;Posicion valida = 0

MOV R3,#0 ;Posicion valida = 0

MOV R4,#CASILLA_VACIA ;Casilla = casilla vacia

;Cuerpo del bucle

```
bucle  ADD    R2,R5,R7      ;FA = FA + SF
MOV    R2,R2,LSL #24      ;Conserva solo los 8 primeros bits
MOV    R5,R2,ASR #24      ;Expande el número con signo de 8 bits, a uno de 32
ADD    R2,R6,R8           ;CA = CA + SC
MOV    R2,R2,LSL #24      ;Conserva solo los 8 primeros bits
MOV    R6,R2,ASR #24      ;Expande el número con signo de 8 bits, a uno de 32
```

LDR R2,[R1]

ADD R2,R2,#1 ;Longitud = Longitud + 1;

STR R2,[R1] ;Actualiza longitud en memoria

ADD R10,R10,#1 ; Longitud = Longitud + 1;

;ficha_valida Se omite porque es igual a la anterior

;Condiciones del bucle

```
while LDR R2,[SP] ;Carga R2 = posicion_valida
      CMP R2,#1 ;Comprueba si la posición es válida
```

```
while CMP R3,#1 ;Comprueba si la posición es válida
      BNE if
      CMP R4,R9 ;Comprueba si la casilla es igual al color
      BNE bucle
```

```
if MOV R0,#NO_HAY_PATRON ;Presupone que no va a haber patrón
   CMP R4,R9 ;Comprueba si la casilla es igual al color
   BNE fin
   LDR R2,[R1]
   CMP R2,#0 ;Comprueba si la longitud es 0
   CMP R10,#0 ;Comprueba si la longitud es 0
   MOVGT R0,#PATRON_ENCONTRADO
```

```
fin ADD SP, SP,#4 ;Quita espacio reservado para pos. valida.
    STR R10,[R1] ;Guarda en memoria los cambios de longitud
    LDMIA SP!,{R4-R11,LR}
    MOV PC,LR

    END
```