

5^{AVR}



Cuerpo del programa

Subrutina_1:

LDI R16,\$FF

LDI R16,\$04

.

.

.

RET

"Dos-puntos" al final de la etiqueta

Usar guion bajo para separación

Puede terminar en RET

;el cursor regresa una línea después a donde fue
;invocado

Subrutina_2:

LDI R16,\$FF

LDI R16,\$04

.

.

.

RJMP saltar_a_otra_etiqueta

Puede terminar en RJMP

LDI R16, 0xFF ; (usando "0x")

También:

LDI R16, \$FF ; (usando el símbolo de "\$")

En notación binaria:

LDI R16, 0b11111111 ; (todos los bits juntos)

También:

LDI R16, 0b1111_1111 ; (usando un separador de guion bajo "_")

En notación decimal:

LDI R16, 255 ; (el valor decimal va desde el 0 hasta el 255)

Los AVR trabajan con *lógica positiva*, es decir, para configurar un bit como salida el bit debe valer "1", y para entrada el bit debe valer "0"

En notación octal:

```
LDI R16, 0377 ; (al valor octal se le debe agregar un 0
               ; al principio de la cifra)
```

Posteriormente se envía el valor del registro (en este caso el R16) al registro DDRA, de esta forma:

```
LDI R16, 0b11111111 ; (todos los bits juntos)
OUT DDRA, R16        ; (aquí está configurado
                     ; para el puerto A)
```

Recuerde

Recuerde este cuadro-resumen para las diferentes nomenclaturas que puede manejar el AVR:

Resumen de nomenclaturas:

```
LDI R16, 0xFF      ; HEXADECIMAL 1
LDI R16, $FF       ; HEXADECIMAL 2
LDI R16, 255       ; DECIMAL 1
LDI R16, 8          ; DECIMAL 2
LDI R16, 0377       ; OCTAL
LDI R16, 0b11111111 ; BINARIO 1
LDI R16, 0b1111_1111 ; BINARIO 2
LDI R16, -128       ; NÚMEROS SIGNADOS
LDI R16, 'A'        ; CARACTERES ASCII
```

Si deseamos configurar unos pines como entrada y como salida, digamos:

Bit 7=1	Bit 6=0	Bit 5=0	Bit 4=1	Bit 3=0	Bit 2=1	Bit 1=0	Bit 0=1
Salida	Entrada	Entrada	Salida	Entrada	Salida	Entrada	Salida

Escribiremos así:

```
LDI R16, 0b1001_0101
OUT DDRA, R16
```

Para configurar el registro de salida (PORTx) para sacar voltaje en un puerto (en este caso es el puerto A), se usa la siguiente sintaxis :

```
LDI R16, 0b1001_0101 ; Nótese que usamos el mismo valor del
                     ; DDRA anterior
```

```
OUT PORTA, R16           ;0b1001_0101) para sacar voltaje por
                          ;Bit7, Bit4, Bit2, Bit0
```

```
LDI R16, 0b1001_0101    ;si ésta es la configuración deseada
                          ;entrada/salida
OUT DDRA,R16             ;se deberá usar una máscara
                          ;"complemento"
```

```
IN R17,PINA              ;el registro PINx para leer el
                          ;voltaje presente en el "puerto A"
LDI R18, 0b0110_1010    ;Valor cargado a R18 que contiene
                          ;el complemento de la palabra
                          ;0b1001_0101
AND R18,R17              ;con la instrucción AND el valor de
                          ;"R18-AND-R17" será el que sólo
                          ;deseamos leer, que en este caso
                          ;es el valor de los bits de entrada.
                          ;De esta forma separamos el valor
                          ;de los bits de entrada del valor
                          ;de los bits de salida.
```