

```
;PROGRAMA PARA EMULAR UN FLIP FLOP "TIPO SR"
;USANDO UN AVR ATTINY2313
```

Encabezado para ATtiny2313

Stack Pointer para ATtiny2313

```
;                0bxxxx_xxRS
LDI R16, 0b0000_0000 ;PORTB como entrada para R y S
OUT DDRB,R16
```

```

;
;
; Q-NEG
; Q
LDI R16, 0b0000_0011 ;PORTD como salida para Q Y Q-negada
OUT DDRD,R16

LEYENDO:
IN R17,PINB ;Leyendo PINB para detectar R y S
LDI R18,0b0000_0000 ;Compara si la entrada es R=0 y S=0
AND R18,R17 ;Máscara para primera condición
CP R18,R17
BREQ QQ1

LDI R18,0b0000_0001 ;Compara si la entrada ES R=0 y S=1
AND R18,R17 ;Máscara para segunda condición
CP R18,R17
BREQ QQ2

LDI R18,0b0000_0010 ;Compara si la entrada es R=1 y S=0
AND R18,R17 ;Máscara para tercera condición
CP R18,R17
BREQ QQ3

LDI R18,0b0000_0011 ;Éste es el estado prohibido
AND R18,R17 ;Compara si la entrada es R=1 y S=1
CP R18,R17 ;Máscara para cuarta condición
BREQ QQ4
RJMP LEYENDO

QQ1:
; 0bx|x|x|x_x|x|Q_neg|Q
LDI R19,0b0000_0010 ;La salida es 0bx|x|x|x|x|x|Q_neg|Q
OUT PORTD,R19
RJMP LEYENDO

QQ2:
; 0bx|x|x|x_x|x|Q_neg|Q
LDI R19,0b0000_0010
OUT PORTD,R19
RJMP LEYENDO

```

QQ3:

```

;          0bx|x|x|x_x|x|Q_neg|Q
LDI R19,0b0000_0001
OUT PORTD,R19
RJMP LEYENDO

```

;CONDICIÓN PROHIBIDA

QQ4:

```

;          0bx|x|x|x_x|x|Q_neg|Q
LDI R19,0b0000_0000
OUT PORTD,R19
RJMP LEYENDO ←

```

Este programa ha terminado aquí; sin embargo, si se desea se puede introducir un LED que indique cuando el **estado-prohibido** se ha activado, también se puede alterar el comportamiento de Q y \bar{Q}