

```
LEYENDO:
IN R17,PINA
```

```
LEYENDO_HUNDREDS:
MOV R0,R17
SUBI R17,100
BRCS LEYENDO_DECENAS
INC R18
BRCC LEYENDO_HUNDREDS
```

Con la instrucción BRCC LEYENDO\_HUNDREDS, el cursor regresará a LEYENDO\_HUNDREDS hasta que la bandera de Carry se active. R18 almacena el número de vueltas, ergo, el número de centenas. Cuando el Carry se active, el cursor saltará a la subrutina de las decenas LEYENDO\_DECENAS. Y así sucesivamente hasta terminar con las unidades. Esta subrutina puede ser usada para contar el número de cifras deseadas (centenas, unidad de millar, etc.)

### Subrutina:

```
;CON ESTA SUBROUTINA EL DISPLAY DESPLIEGA EL VALOR LEÍDO
;PROVENIENTE DEL CONVERTIDOR ADC0804.
```

Encabezado para ATmega8515

Stack Pointer para ATmega8515

```
LDI R16,$00
OUT DDRA,R16
```

```
LDI R16,$FF
OUT DDRC,R16
```

```
LDI R16,$FF
OUT DDRE,R16
```

```
;SE USÓ "PUERTO C y E" PARA CONTROL DEL DISPLAY LCD
```

```
;E    = PE0...31-6
```

```
;R/W = PE1...30-5
```

```
;RS   = PE2...29-4
```

Tabla resumen de las confi- guraciones del LCD usado en este libro
--------------------------------------------------------------------------

```
;RS= 0...CONTROL
```

```
;RS= 1...DATOS
```

```
;INICIALIZAR EL LCD
```

```
;LDI R16,$0C ;SIN_UNDERLINE_SIN_BLINK
```

```
;LDI R16,$10 ;RECORRE_IZQUIERDA_CURSOR
```

```
;LDI R16,$14 ;RECORRE_DERECHA_CURSOR
```

```
;LDI R16,$18 ;RECORRE_IZQUIERDA_DISPLAY
```

```
;LDI R16,$1C ;RECORRE_DERECHA_DISPLAY
```

```
;LDI R16,$80 ;ESCRIBE_LINEA_UNO
```

```
;LDI R16,$C0 ;ESCRIBE_LINEA_DOS
```

```
;LDI R16,$01 ;CLEAR_DISPLAY
```

```
;LDI R16,$02 ;HOME
```

```
;LDI R16,$0F ;DISPLAY_CONTROL_ON
```

```
;LDI R16,$08 ;DISPLAY_CONTROL_OFF
```

```
;LDI R16,$30 ;UNA_LINEA..OPCIONAL
```

```
;LDI R16,$38 ;DOS_LINEAS..OPCIONAL
```

```
;LDI R16,$3C ;8-Bits/2 líneas/5X10 dot/
```

INICIA:

```
;INICIALIZA DISPLAY LCD Y HABILITA LÍNEA UNO
```

```
RCALL DISPLAY_CONTROL_ON
```

```
RCALL CLEAR_DISPLAY
```

```
RCALL ESCRIBIR_EN_LINEA_UNO
```

```
RCALL HOME
```

```
;LÍNEA DE ARRIBA DEL LCD
```

```
RCALL L
```

```
RCALL E
```

```
RCALL C
```

```
RCALL T
```

```
RCALL U
```

```
RCALL R
```

```
RCALL A
```

RCALL ESCRIBIR\_EN\_LINEA\_DOS

LEYENDO:

IN R17,PINA ;Se obtiene el dato del CONVERTIDOR

;CONVERTIDOR ADC A DISPLAY LCD

;\*\*\*\*\*

CONVIERTE\_DATO\_A\_LCD:

;ESTA SUBROUTINA DETECTA LA ACTIVACION DEL FLAG CARRY "C"

LDI R18,0 ;CENTENAS

LDI R19,0 ;DECENAS

LDI R20,0 ;UNIDADES

LEYENDO\_HUNDREDS:

MOV R0,R17

SUBI R17,100

BRCS LEYENDO\_DECENAS

INC R18 ;CENTENAS

BRCC LEYENDO\_HUNDREDS

LEYENDO\_DECENAS:

MOV R17,R0

SALTO:

MOV R0,R17

SUBI R17,10

BRCS LEYENDO\_UNIDADES

INC R19 ;DECENAS

BRCC SALTO

LEYENDO\_UNIDADES:

MOV R17,R0

SALTO\_2:

MOV R0,R17

SUBI R17,1

BRCS SACANDO\_ASCII

INC R20 ;UNIDADES

BRCC SALTO\_2

RJMP CONVIERTE\_DATO\_A\_LCD

Con este bloque extraemos el número de centenas, decenas y unidades de un número de 8-bits (con más etapas se puede manejar a 16, 32-bits, o más)

SACANDO\_ASCII:

;EN ESTA SECCIÓN SE ORDENAN LOS DATOS A SER DESPLEGADOS  
;EN EL DISPLAY.

```
LDI R21,$30
ADD R18,R21
ADD R19,R21
ADD R20,R21
```

A cada centena, decena y unidad, se le debe sumar \$30 para convertirlo a ASCII. Revise la Tabla de ASCII

```
OUT PORTC,R18 ;Saca CENTENAS al LCD
```

```
LDI R16,$05
OUT PORTE,R16
RCALL DELAY
LDI R16,$00
OUT PORTE,R16
```

```
OUT PORTC,R19 ;Saca DECENAS al LCD
```

```
LDI R16,$05
OUT PORTE,R16
RCALL DELAY
LDI R16,$00
OUT PORTE,R16
```

```
OUT PORTC,R20 ;Saca UNIDADES al LCD
```

```
LDI R16,$05
OUT PORTE,R16
RCALL DELAY
LDI R16,$00
OUT PORTE,R16
```

```
RCALL RECORRE_IZQUIERDA_CURSOR
RCALL RECORRE_IZQUIERDA_CURSOR
RCALL RECORRE_IZQUIERDA_CURSOR
```

```
RCALL DELAY
RCALL DELAY
RCALL DELAY
RCALL DELAY
RCALL DELAY
RCALL DELAY
```

Necesitamos recorrer el cursor del LCD tres lugares a la izquierda para sobrescribir el nuevo dato

```
RJMP LEYENDO
```

```
.INCLUDE "ARCHIVO_CONFIGURACION_Y_ASCII_DEL_LCD"
```

**Programa:**

```
.INCLUDE "TN13DEF.INC"  
  
.CSEG  
.ORG 0000
```

```

LDI R16,LOW(RAMEND)
OUT SPL,R16

LDI R16,$01
OUT DDRB,R16           ;PB0 para LED que parpadea

;**** REGISTRO DE SELECCION DE CANAL Y REFERENCIA ***

LDI R16,0<<REFS0|0<<ADLAR|1<<MUX1|1<<MUX0           ;Selecciona
                                                    ;el canal
OUT ADMUX,R16                                           ;referencia a
                                                    ;VCC (5V)
                                                    ;ADC3

LDI R16,1<<ADEN|1<<ADSC|1<<ADATE|0<<ADPS2|0<<ADPS1|0<<ADPS0
OUT ADCSRA,R16

LDI R16,0<<ACME|0<<ADTS2|0<<ADTS1|0<<ADTS0
OUT ADCSRB,R16

CICLO:
LDI R17,$01
OUT PORTB,R17
RCALL DELAY

LDI R17,$00
OUT PORTB,R17
RCALL DELAY
RJMP CICLO

DELAY:
LDI R31,$01

IN R30,ADCL
IN R29,ADCH

LDI R27,$00

```

```

DELAY1:
DEC R31
CP R31,R27
BRNE DELAY1

DEC R30
CP R30,R27
BRNE DELAY1

DEC R29
CP R29,R27
BRNE DELAY1
RET

```

Esta subrutina es de prueba, por lo que habría que hacer una relación del voltaje de entrada debido al potenciómetro conectado al canal 3 y la frecuencia generada en el LED. La subrutina funciona bien, pero sea paciente en calibrar el potenciómetro. En esta sintaxis el LED va a parpadear de muy lento a muy rápido