

```
.INCLUDE "M8515DEF.INC"
```

```
.CSEG
.ORG 0
```

```
INICIO:
```

```
IN R0,SPL
IN R1,SPH
```

Este bloque es muy útil para almacenar el estado del Stack Pointer y así evitar errores

```
;Guardar el valor del "SP" BAJO en
;R0=SPL, que ;aquí es $00
;Guardar el valor del "SP" ALTO en
;R1=SPH, que ;aquí es $00
```

```
LDI R16,LOW(RAMEND) ;En esta sección se configura el STACK
;POINTER
OUT SPL,R16 ;Los valores que toman los registros
;SPL y SPH del STACK
LDI R16,HIGH(RAMEND) ;POINTER dependen del modelo de AVR
OUT SPH,R16
```

```

IN R0,SPL           ;Guardar el valor del "SP" BAJO en
                    ;R0=SPL, que aquí es $5F
IN R1,SPH           ;Guardar el valor del "SP" ALTO en
                    ;R1=SPH, que aquí es $02

LDI R16,$00         ;En esta sección el valor del stack
                    ;pointer no se modifica

OUT DDRA,R16
LDI R16,0B1010_0000
OUT DDRD,R16

IN R0,SPL           ;Por eso el valor del SPL aquí sigue
                    ;siendo $5F
IN R1,SPH           ;Por eso el valor del SPH aquí sigue
                    ;siendo $02

LEYENDO_OTRA_VEZ:

RCALL LEER_TECLADO  ;Aquí al ejecutar el "RCALL" cambió
                    ;el valor de ;SPL a ;$5D
                    ;y el valor de SPH a $00

FIN:RJMP FIN

LEER_TECLADO:

IN R0,SPL           ;Aquí el valor del SPL sigue siendo
                    ;$5D
IN R1,SPH           ;Aquí el valor del SPH sigue siendo
                    ;$00

LDI R20,200         ;Cargar R20=200

PUSH R20            ;Aquí se modificó el valor del STACK

IN R0,SPL           ;Así que el nuevo valor del SPH es
                    ;$5C
IN R1,SPH           ;Así que el nuevo valor del SPH es
                    ;$00

POP R20             ;Aquí se modificó el valor del STACK

```

```

IN R0,SPL           ;Así que el nuevo valor del SPH es
                    ;$5D
IN R1,SPH           ;Así que el nuevo valor del SPH es $00

RJMP LEYENDO_OTRA_VEZ ;Aquí se modificó el valor del STACK

```

```

.INCLUDE "M8515DEF.INC"

```

```

.DEF ENTRADA_TECLADO    =R20
.DEF GUARDAR_NUM        =R21

```

```

.CSEG
.ORG 0

```

```

RJMP INICIO                                ;DIR $000 <--- EL CURSOR SALTA
                                           ;AQUÍ EN LUGAR DE
                                           ;
                                           "ESPERANDO_TECLA"
RJMP LEER_TECLADO                          ;DIR $001
                                           ↑

INICIO:
LDI R16,LOW(RAMEND)
OUT SPL,R16
LDI R16,HIGH(RAMEND)
OUT SPH,R16

LDI R16,$00
OUT DDRA,R16
LDI R16,0B1010_0000
OUT DDRD,R16

LDI R16,1<<INT0
OUT GICR,R16

LDI R16,1<<ISC01|1<<ISC00
OUT MCUCR,R16

SEI

ESPERANDO_TECLA:                         ;<---- AQUÍ DEBIERA REGRESAR
                                           ;EL CURSOR
                                           ↑
RJMP ESPERANDO_TECLA

LEER_TECLADO:
MOV GUARDAR_NUM,ENTRADA_TECLADO

PUSH GUARDAR_NUM
RETI                                     ;<----- ESTE "RETI" ENVÍA A "RJMP
                                           ;INICIO" EN LUGAR DE "ESPERANDO_TECLA"

.
.
.

```

```

.
.
.
SEI

```

```

ESPERANDO_TECLA:

```

```

RJMP ESPERANDO_TECLA

```

```

LEER_TECLADO:

```

```

MOV GUARDAR_NUM, ENTRADA_TECLADO

```

```

SEI

```

```

RJMP ESPERANDO_TECLA

```

```

FIN: RJMP FIN

```

```

.
.
.
SEI

```

```

ESPERANDO_TECLA:           ;<----EL CURSOR DEBE REGRESAR AQUÍ
                           ;DESPUES DEL RETI

```

```

RJMP ESPERANDO_TECLA

```

```

LEER_TECLADO:

```

```

MOV GUARDAR_NUM, ENTRADA_TECLADO

```

```

IN R0, SPL                 ;Guardar el STACK POINTER BAJO

```

```
IN R1,SPH                                ;Guardar el STACK POINTER ALTO

PUSH GUARDAR_NUM                         ;Este PUSH modifica el valor del STACK
                                         ;POINTER

.
.
;CUALQUIER_OTRA_COSA
.
.
OUT SPL,R0                               ;<----Esta línea restablece el valor del
                                         ;STACK POINTER BAJO
OUT SPH,R1                               ;<----Esta línea restablece el valor del
                                         ;STACK POINTER ALTO

RETI

RJMP OTRO_PROCESO

FIN: RJMP FIN
```

Encabezado para ATmega8515

INICIO:

Stack Pointer para ATmega8515

```
LDI R16,$00
OUT DDRA,R16
LDI R16,0b1010_0000
OUT DDRD,R16
```

LEYENDO_OTRA_VEZ:

```
RCALL LEER_TECLADO      ;<---- EN ESTA LÍNEA SE GENERA EL
                        ;ERROR "STACK OVERFLOW"
```

FIN:RJMP FIN

LEER_TECLADO:

LDI R20,200

```
RJMP LEYENDO_OTRA_VEZ ;<---- TERMINA CON "RJMP" EN LUGAR DE
                        ;"RET"
```

```
.INCLUDE "M8515DEF.INC"  
.CSEG  
.ORG 0
```

INICIO:

```
LDI R16,LOW(RAMEND)  
OUT SPL,R16  
LDI R16,HIGH(RAMEND)  
OUT SPH,R16
```

```
LDI R16,$00  
OUT DDRA,R16  
LDI R16,0b1010_0000  
OUT DDRD,R16
```

LEYENDO_OTRA_VEZ:

```
RCALL LEER_TECLADO
```

```
FIN:RJMP FIN
```

LEER_TECLADO:

```
LDI R20,200
```

```
RET
```

Del "RET" se regresa a "FIN"