

```

;-----
;Programa tomado en el examen 21/06/17
;Estacionamiento con semáforo (Verde, Rojo) y Barrera, Sensores en Entrada y Salida.
;
;
;       Registro IE
; EA | -- | -- | ES | ET1 | EX1 | ET0 | EX0 |
;  0  |  0  |  0  |  0  |  0  |  1  |  0  |  1  |
;
;
;       Registro IP
; -- | -- | PT2 | PS | PT1 | PX1 | PT0 | PX0 |
;  0  |  0  |  0  |  0  |  0  |  0  |  0  |  0  |
;
;
;       Registro TCON
; TF1 | TR1 | TF0 | TR0 | IE1 | IT1 | IE0 | IT0 |
;  0  |  0  |  0  |  0  |  0  |  1  |  0  |  1  |
;
;Archivo: examenSinTimer.s03
;Fecha de realización: 22/06/2017
;Autor: Sebastian Caccavallo - electgpl.blogspot.com.ar
;Modificación:
;-----
;-----
;
;       ORG      0000H      ;Dirección del origen del programa
;       JMP      SETUP     ;Salta al Inicio del programa
;
;
;       ORG      0003H      ;Dirección de interrupción externa INT0
;       JMP      EXT_INT0   ;Salta a subrutina de interrupción
;
;
;       ORG      0013H      ;Dirección de interrupción externa INT1
;       JMP      EXT_INT1   ;Salta a subrutina de interrupción
;
;
;CONT_A      EQU      50H      ;Dirección de la variable CONT_A
;N_AUTO      EQU      51H      ;Dirección de la constante N_AUTO
;
;
;SETUP:      MOV      TCON,#05H ;Configura IT1=1 y IT0=1
;            MOV      IE,#05H  ;Configura EX1=1 y EX0=1
;            MOV      IP,#00H  ;Des-habilita prioridad de interrupción
;            SETB     EA       ;Habilita interrupción global
;            MOV      CONT_A,#00H ;Inicializa contador CONT_A con 0
;            MOV      N_AUTO,#05H ;Se setea máximo de autos a 5 (puede variar)
;            SETB     P1.0     ;Semáforo Luz Verde TRUE
;            CLR      P1.1     ;Semáforo Luz Roja FALSE
;            SETB     P1.2     ;Barrera TRUE (Alta)
;
;
;LOOP:       MOV      A,CONT_A  ;Cargamos Acumulador con CONT_A
;            CJNE     A,N_AUTO,NOTEQ ;Comparamos CONT_A (cantidad actual) con
;N_AUTO (Máximo)
;            SJMP     IGUAL     ;Si son iguales salta a IGUAL
;NOTEQ:      JC       MENOR     ;Si Carry=1 Salta a MENOR
;            SJMP     MAYOR     ;Si Carry=0 Salta a MAYOR
;IGUAL:      CALL     COMPLETO  ;Si IGUAL, llama a subrutina COMPLETO
;            SJMP     LOOP     ;Repite el ciclo con LOOP
;MAYOR:      CALL     COMPLETO  ;Si MAYOR, llama a subrutina COMPLETO
;            SJMP     LOOP     ;Repite el ciclo con LOOP
;MENOR:      CALL     LIBRE     ;Si MENOR, llama a subrutina LIBRE
;            SJMP     LOOP     ;Repite el ciclo con LOOP
;
;
;COMPLETO:   CLR      P1.0     ;Semáforo Luz Verde FALSE
;            SETB     P1.1     ;Semáforo Luz Roja TRUE
;            CLR      P1.2     ;Barrera FALSE (Baja)
;            RET             ;Retorno de Subrutina
;
;

```

```

LIBRE:      SETB    P1.0          ;Semáforo Luz Verde TRUE
            CLR     P1.1          ;Semáforo Luz Roja FALSE
            SETB    P1.2          ;Barrera TRUE (Alta)
            RET                     ;Retorno de Subrutina
;
EXT_INT0:   MOV     A,CONT_A      ;Intercambio de variables
            INC     A              ;Incrementa A
            MOV     CONT_A,A      ;Carga CONT_A con valor incrementado
            RETI                   ;Retorno de interrupción
;
EXT_INT1:   MOV     A,CONT_A      ;Intercambio de variables
            DEC     A              ;Incrementa A
            JC      CERO          ;Validación de que CONT_A no sea menor que
cero
            MOV     CONT_A,A      ;Carga CONT_A con valor incrementado
            RETI                   ;Retorno de interrupción
CERO:       MOV     CONT_A,#00H   ;Mueve cero a CONT_A
            RETI                   ;Retorno de interrupción
;
            END                   ;Fin del programa

```