

```

;-----
; MIC2-EJERCICIO-DACmaxmin (0605-2019)
;
; Utilizando un conversor Analogico-Digital de 8 bits
; y que esta conectado al puerto 1.
; A) Almacenar 200 muestras a maxima velocidad a partir de
; la posiscion de memoria 4000H de la memoria XDATA.
; B) Obtener el valor Maximo y el Minimo de las muestras.
;
; Archivo: A24-EJERCICIO-DACmaxmin--0605-completo
; Fecha de realizacion: 0605-2019
;-----
;
;          ORG      0000H      ; vector reset power-on
;          JMP      SETUP      ; Salta al inicio del programa
;
KMUEST EQU      50H      ; REGISTRO CONTADOR de MUESTRAS
SC      EQU      P3.7      ; SC= START CONVERT BIT
EOC      EQU      P3.6      ; EOC = END of CONVERT BIT
MAX      EQU      52H      ; Registro Valor maximo
MIN      EQU      54H      ; Registro Valor minimo
;                                     **
;          ORG      100H      ;
SETUP:  MOV      DPTR,#4000H  ; puntero stack-muestra
;          MOV      KMUEST,#200 ; numero de muestras
;          MOV      MAX,#0D      ; Valor miximo inicial = 0
;          MOV      MIN,#255D    ; Valor minimo inicial = 255
;                                     **
;          SET      SC          ; disparo al ADC para muestreo de DATO
;          CLR      SC          ; coloco en forma normal a la señal
ESPERA:  JNB      EOC,ESPERA    ; espero que ADC convierta
;          MOV      A,P1        ; lee muestra
;          MOVX     @DPTR,A      ; guarda muestra
;                                     **
;          CJNE     A,MAX,SMAX   ; compara a MAXIMO
;          JMP      LMIN        ; igual --> sigue
SMAX:    JC      NMAX          ; A < MAX
xMAX:    MOV      MAX,A        ; mayor --> nuevo MAX
;          JMP      LMIN        ; sigue
NMAX:    JMP      LMIN        ; menor --> sigue
;                                     **
;          CJNE     A,MIN,SMIN   ; compara a MINIMO
;          JMP      NEXT        ; igual --> sigue
SMIN:    JC      NMIN          ; A < MIN
xMIN:    JMP      NEXT        ; mayor --> sigue
NMIN:    MOV      MIN,A        ; menor --> nuevo MIN
;                                     **
;          INC      DPTR        ; actualiza puntero stack
;          DJNZ     KMUEST,REPTE ; decrementa contador (Jump Outbound)
FIN:     JMP      FIN          ; stop dinamico
REPTE:   JMP      LOOP
;                                     **
;          END                  ;Fin del Programa**

```