

```

;-----
;
;-----
;Programa que recibe datos por UART mediante interrupción y lo guarda en una
tabla
;Se requiere de un cristal de 11.059MHz para el baudrate de 9600bps. con 8051 @
8952.
;Utilización de UART en Modo 1 y Timer 1 en Modo 2.
;Archivo: RecibeUART40h.s03
;Fecha de realización: 13/06/2017
;Autor: Sebastian Caccavallo - electgpl.blogspot.com.ar
;Modificación:
;-----
;-----
                ORG     0000H                ;Dirección del Reset
                JMP     SETUP                ;Salta al Inicio del programa
;
                ORG     0023H                ;Dirección de interrupción
                JMP     UARTINT              ;Salta a la interrupción
;
SETUP:          MOV     SCON, #50h           ;UART en MODO 1 (8BIT), REN=1
                ORL     TMOD, #20h           ;TIMER 1 en MODO 2
                MOV     TH1, #0FDh          ;9600bps @ 11.059MHz
                MOV     TL1, #0FDh          ;9600bps @ 11.059MHz
                SETB    ES                   ;Habilitación de Interrupción Serial
                SETB    EA                   ;Habilitación de Interrupciones Globales
                SETB    TR1                  ;Activa TIMER 1
                MOV     DPTR, #40H           ;Cargo 40h en DPTR (Origen de la tabla)
;
LOOP:           CPL     P1.0                 ;Toggle LED en P1.0
                ACALL   DELAY                ;Rutina de Delay de 1s
                SJMP    LOOP                 ;Loop principal
;
UARTINT:        JNB     RI, UARTINT           ;Espera el dato Serie
                MOV     A, SBUF              ;Mueve el dato del buffer al acumulador
                CLR     RI                   ;Borra el flag de recepción
                MOVX    A, @DPTR             ;Cargo el valor apuntado en el acumulador
                INC     DPTR                 ;Incrementamos DPTR
                RETI                          ;Retorno de Interrupción
;
DELAY:          MOV     R7, #6D              ;[2]Cargamos R7 con 6
WAIT1:          MOV     R6, #185D            ;[2]Cargamos R6 con 185
WAIT2:          MOV     R5, #200D            ;[2]Cargamos R5 con 200
WAIT3:          DJNZ    R5, WAIT3             ;[2]Decrementamos R5 y Salta a WAIT3
                DJNZ    R6, WAIT2             ;[2]Decrementamos R6 y Salta a WAIT2
                DJNZ    R7, WAIT1             ;[2]Decrementamos R7 y Salta a WAIT1
                RET                          ;[2]Retorno de Subrutina
;
                END                          ;Fin del programa

```