

```

                                code1
;,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
;  Programme d'émission sur la liaison série
;,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
#include <sfr51.inc>

    org 0000h
    ljmp gerer
    org 0030h
gerer:  ; initialisation du TIMER1
        mov TMOD,#20h          ; mode auto rechargement - 8 bits
        mov TH1,#F3h          ; vitesse de transmission 2400
        mov TL1,#F3h          ; vitesse de transmission
        clr TF1
        setb TR1              ; lancement timer
        ; initialisation de la liaison série
        clr TI
        setb REN
        setb SM1              ; configuration de la ls est : 8 message, 1 stop bit
        ; initialisation de DPTR
        mov DPTR,#tab
        ; boucle tant que(1) faire
TQ:     mov A,#00h              ; index à 0
        ; tant que(index<4) faire
TQ1:    mov B,A                 ; sauvegarde de l'index
        movc A,@A+DPTR         ; car_e <- tab(i)
        lcall emettre          ; appel à la fonction emettre
        mov A,B                 ; restauration de l'index
        inc A
        cjne A,#04h,TQ1
boucle: ljmp boucle
;,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
;  fonction d'émission
;  entree :  car_e  A  char
;            TI flag de la liaison série
;  sortie :  SBUF buffer de la liaison série
emettre:  mov SBUF,A           ; SBUF <- car_e
attente:  jnb TI,attente      ; tant que (TI=0) faire attendre
        clr TF1
attente1: jnb TF1,attente1     ; tant que(TF1=0) faire attendre
        clr TI
        ret                   ; retour à la fonction d'appel

; implantation du tableau
tab:      db 45h ; code ascii de E
          db 53h ; code ascii de S
          db 54h ; code ascii de T
          db 20h ; code ascii de l'espace

        end

```