CH-1092 Belmont, Switzerland
info@didel.com, www.didel.com

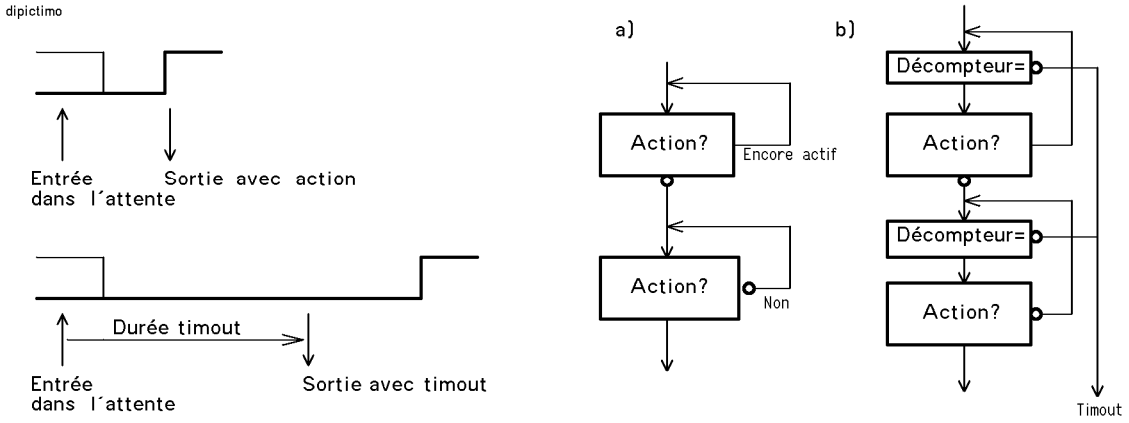
Attente d'un signal avec foreclos (time-out)

Le problème est d'attendre une action (poussoir, start bit d'une transmission série) avec une limite de temps. Le problème peut se résoudre avec une approche monotâche (le processeur gère les compteurs et surveille le signal), en utilisant le timer pour éviter de compter, en utilisant les interruptions du timer, en utilisant la programmation synchrone avec une tâche time-out et une tâche d'attente de l'action.

Solutions monotâche

Sans timeout, l'attente correspondant à l'organigramme de la figure a) s'écrit:

```
A$:     TestSkip,BC   Port:#bAction
        Jump      A$
B$:     TestSkip,BS   Port:#bAction
        Jump      B$
... Suite si action
```



L'insertion d'un décompteur de temps (initialisé à zéro s'il n'y est pas déjà) comme dans la figure b) s'écrit:

```
; Variable à déclarer: CTimo
A$:     TestSkip,BS   Port:#bAction    ; on attends le passage à zéro
        Jump      B$
        DecSkip,EQ  CTimo
        Jump      A$
        Jump      ExitTimout
B$:     TestSkip,BC   Port:#bAction
        Jump      C$
        DecSkip,EQ CTimo
        Jump      B$
        Jump      ExitTimout
C$:
... Suite si action avant timeout
ExitTimout:
... suite si timeout
```

La durée maximale du timeout, si le décompteur est à zéro au début, est de 256 x 5 μs $\simeq$ 1,3 ms (processeur à 4 MHz). Pour un délai supérieur, il faut un décompteur 16 bits

```
A$:     TestSkip,BS   Port:#bAction
        Jump      B$
        DecSkip,EQ  CTimoLow
        Jump      A$
        DecSkip,EQ  CTimoHigh
        Jump      A$
        Jump      ExitTimout
B$:     TestSkip,BC   Port:#bAction
        Jump      C$
        DecSkip,EQ  CTimoLow
        Jump      B$
        DecSkip,EQ  CTimoHigh
        Jump      B$
        Jump      ExitTimout
... Suite si action avant timeout
ExitTimout:
... suite si timeout
```

La durée maximale est maintenant de 256 x 256 x 5 μs $\simeq$ 0.34 s. Le temps de réponse à l'action est de 7 μs au maximum. Si on attends un signal série à 9600 bits/s, cette erreur de 7% sur le centrage des échantillonages suivants est encore acceptable.

Solutions avec le timer

Le timer est un compteur par 256 avec prédiviseur par 2, 4, ... 256 au choix. Il peut donc mesurer jusqu'à 65 ms.

```
; Initilisation du timer
        IniOption = 2'00000101      ; 101 --> 2**(5+1) prédiv par 64
        TimoDur   = 1000/64                 ; Pour une durée de 1 ms
        Move      #IniOption,W      ; Prescaler :2
        Move      W,Option

; Attente action
        Move      #256-TimDur,W
        Move      W,TMR0
A$:     TestSkip,BC   IntCon:#TOIF
        Jump      Timout
        TestSkip,BC   Port:#bAction
        Jump      A$
B$:     TestSkip,BC   IntCon:#TOIF
        Jump      Timout
        TestSkip,BS   Port:#bAction
        Jump      B$
... Suite si action avant timeout
ExitTimout:
        Clr       IntCon:#TOIF
... suite si timeout
```

L'avantage principal est un temps de latence (temps de réponse à l'action) réduit à 5 ms. Si un timout supérieur à 65 ms, cette solution n'est plus très rentable, car il faut gérer un compteur dans la boucle d'attente de l'action.

Timer par interruption

Le timer par interruption donne la latence minimale du premier programme (3 μs). L'interruption étant dans ce cas pour avorter un processus, elle peut se gérer d'une façon simplifiée, sans sauver l'état, et sans réactivation prévue (il n'y a pas d'autre interruption en parallèle).

```
; Initilisation du timer
        IniOption = 2'00000101      ; 101 --> 2**(5+1) prédiv par 64
        TimoDur   = 1000/64                 ; Pour une durée de 1 ms
        Move      #IniOption,W      ; Prescaler :2
        Move      W,Option

; Instructions d'interruption (à l'adresse 4)
Inter:  Clr       IntCon    ; Flag, GIE, TOIE = 0
        ... suite si timeout

; Attente action
        Move      #256-TimDur,W
        Move      W,TMR0
        Move      #2'10100000,W    ; GIE et TOIE activés
        Move      W,IntCon
A$:     TestSkip,BC   Port:#bAction
        Jump      A$
B$:     TestSkip,BS   Port:#bAction
        Jump      B$
        Clr       IntCon    ; GIE, TOIE = 0 !! ne pas oublier
... Suite si action avant timeout
```

Le délai maximum est comme avant de 65 ms. On peut l'augmenter dans la routine d'interruption, qui doit être écrite plus proprement, et qui ajoute une latence de 20 à 30 μs, inacceptable si l'on surveille l'arrivée d'un message RS232, mais acceptable si c'est l'action sur un poussoir.