



**8 BITS MICROCONTROLLERS**

# HCS08

## MACRO'S MANUAL

### FUNZIONI DI USO GENERALE



**Document Revision : 2.70 17 Maggio.2018**

Author : Rech Marzio – **IW3FBA**

[marzio.rech@negrelliforcellini.gov.it](mailto:marzio.rech@negrelliforcellini.gov.it) : **IZ3JCN**

## **SIGNIFICATO DEI TERMINI USATI**

- MACRO** : Istruzione definita dal Programmatore che chiama un blocco di Istruzioni Assembly.  
E' definita da un nome che ricorda in qualche modo la funzione svolta dalla Macro
- PARAMETRI** : Dati passati insieme alla macro scritti alla destra del nome della Macro e separati da una virgola.  
I Parametri possono essere delle costanti numeriche o delle variabili definite dal programmatore.
- ARGOMENTI** : Vedi PARAMETRI.
- DATA TYPE** : E' il tipo di dato associato ai parametri passati alla macro. Esempio U8, S8, U16, S16, U32, S32..
- U08** : Il dato è un numero Intero senza segno di lunghezza 8 Bit ovvero 1 Byte : 0...255
- S08** : Il dato è un numero Intero con segno di lunghezza 8 Bit ovvero **BYTE** : -128..+127
- U16** : Il dato è un numero Intero senza segno di lunghezza 16 Bit ovvero **WORD** : 0...65535
- S16** : Il dato è un numero Intero con segno di lunghezza 16 Bit ovvero **WORD** : -32768...+32767
- U32** : Il dato è un numero Intero senza segno di lunghezza 32 Bit ovvero **LONG** .
- S32** : Il dato è un numero Intero con segno di lunghezza 32 Bit ovvero **LONG** .
- I08-I16-I32** : Il dato è un numero Intero di cui solo la lunghezza viene specificata a prescindere dal segno.
- FLOAT** : Variabile o costante a 32 Bit del tipo Floating Point (Virgola Mobile) che rappresenta un numero reale composto da Mantissa Esponente e Segno codificata secondo le specifiche IEEE754.
- COST** : Costante: Valore numerico definito e immutabile
- VAR** : Variabile : Valore numerico che può variare all'interno di un determinato intervallo
- LOCAZIONE** : Cella di memoria
- FLAGS** : Bit di una variabile il cui stato segnala una condizione logica piuttosto che un valore numerico
- BYTE** : Variabile o costante a 8 Bit
- WORD** : Variabile o costante a 16 Bit (2 Bytes)
- LONG** : Variabile o costante a 32 Bit (4 Bytes)
- SHORT** : Vedi WORD
- 'In modo circolare'** : Se riferito ad una variabile significa che il suo valore torna al valore minimo dopo un Incremento del suo valore massimo e viceversa.

| LISTA FUNZIONI MACRO GENERALI                       |  |
|---|--|
| NOME MACRO  | FUNZIONE   |
| <b>Delay</b> <i>arg1</i>                            | Effettua un ritardo espresso in ms in base al numero indicato nel parametro.                     |
| <b>DelayAbort</b>                                   | Abortisce il ritardo in corso.   |
| <b>MicrosDelay</b> <i>arg1</i>                      | Effettua un ritardo espresso in $\mu$ s in base al numero indicato nel parametro.                |
| <b>QuietWatchDog</b>                                | Tacita il WatchDog della MCU   |
| <b>DoEvery</b> <i>arg1, arg2</i>                    | Esegue un gruppo di istruzioni ogni N volte  |
| <b>BitMan</b> <i>arg1, arg2, arg3</i>               | Manipola lo stato logico di un Bit (Flag) di una locazione di memoria o registro.                |
| <b>BranchIfBit</b> <i>arg1, arg2, arg3, arg4</i>    | Effettua una diramazione in base allo stato del Bit/Flag di una locazione di memoria o registro. |
| <b>BranchIfValue</b> <i>arg1, arg2, arg3, arg4</i>  | Effettua una diramazione in base al valore contenuto in una locazione di memoria o registro.     |
| <b>GoTo</b> <i>arg1</i>                             | Salta in un punto specificato del programma.   |
| <b>Return</b>                                       | Ritorna al punto successivo dopo aver servito un sottoprogramma                                  |
| <b>IncValue</b> <i>arg1, arg2, arg3, arg4, arg5</i> | Incrementa una variabile numerica fra un valore minimo e massimo in modo circolare.              |
| <b>DecValue</b> <i>arg1, arg2, arg3, arg4, arg5</i> | Decrementa una variabile numerica fra un valore minimo e massimo in modo circolare.              |
| <b>MoveTeTable</b> <i>arg1, arg2, arg3</i>          | Copia un elemento con terminatore \$00 da una tabella ad una variabile                           |
| <b>MoveEITable</b> <i>arg1, arg2, arg3, arg4</i>    | Copia un elemento da una tabella dichiarando la lunghezza dell'elemento                          |
| <b>WaitEvent</b> <i>arg1, arg2, arg3, arg4</i>      | Attende per un determinato tempo che si verifichi un evento segnalato da un Flag .               |
| <b>GenTimeEvent</b> <i>arg1, arg2, arg3</i>         | Genera un evento di tempo ogni N volte una specifica Base di tempi                               |
| <b>ServiceEvent</b> <i>arg1, arg2</i>               | Serve l'evento di Tempo indicato se questo si è verificato                                       |
| <b>SetTimeEvent</b> <i>arg1, arg2</i>               | Setta lo stato di un evento di tempo   |
| <b>GetTimeEvent</b> <i>arg1, arg2, arg3</i>         | Legge il valore corrente del Timer relativo all'evento di tempo.                                 |
| <b>BranchIfTimeEvent</b> <i>arg1, arg2, arg3</i>    | Effettua un Branch se lo specifico TimeEvent è ON o OFF  |

| LISTA FUNZIONI MACRO GENERALI ...continuazione  |  |
|---|--|
| NOME MACRO                                      | FUNZIONE   |
| <b>CreateFlashLedDaemon</b> <i>arg1</i>         | Crea un Daemon per il Flashing di un Led                             |
| <b>DaemonFlashLed</b> <i>arg1</i>               | Serve il Daemon per il Flashing di un Led                            |
| <b>FlashLed</b> <i>arg1,arg2,arg3,arg4,arg5</i> | Controlla la modalità di Flashing di un Led dontrollato da un Daemon |
| <b>KillFlashLed</b> <i>arg1</i>                 | Arresta il Daemon che effettua il Flashing di un Led                 |
|   |  |

| Delay <i>arg1</i>                                      |         |  |      |   |
|--|---------|--|------|---|
| Effettua un ritardo in msec in base al valore indicato |         |  |      |   |
| ProtoType  |         | Delay   Ritardo  |      |   |
|  |         | Argomento  | Type | Descrizione   |
| ARGOMENTI  | ENTRATA | Ritardo  | U16  | Variabile o Costante numerica indicante i ms di ritardo |
|  | RITORNA | -  | -    | -   |
|  |         | A,H,X  | S08  | Invariati   |
| Va inserito in ..                                      |         | Moduli <i>Main</i> , <i>Boot</i>   |      |   |
| ESEMPI   |         | Delay   1300   ← Effettua 1300ms di ritardo<br>Delay   speed   ← Effettua un ritardo in ms in base al contenuto di 'speed' |      |   |

| DelayAbort                    |         |   |      |                  |
|-------------------------------|---------|---|------|------------------|
| Abortisce il ritardo in corso |         |   |      |                  |
| ProtoType                     |         | DelayAbort  |      |                  |
|                               |         | Argomento   | Type | Descrizione      |
| ARGOMENTI                     | ENTRATA | -   | -    | Nessun argomento |
|                               | RITORNA | -   | -    | -                |
|                               |         | A,H,X   | S08  | Invariati        |
| Va inserito in ..             |         | Moduli : TimeEvent o Interrupt Service Routines   |      |                  |
| ESEMPI                        |         | TimerSvc.100ms: ← Modulo TimeEvent base tempi 100 ms<br>brset P1,ptad,exit ← Testa lo stato pulsante P1<br>DelayAbort ← Abortisce il ritardo corrente se P1 e' premuto. |      |                  |

| MicrosDelay <i>arg1</i>                                |         |   |      |   |
|--|---------|---|------|---|
| Effettua un ritardo in µsec in base al valore indicato |         |   |      |   |
| ProtoType  |         | MicrosDelay   Ritardo                               |      |   |
|  |         | Argomento   | Type | Descrizione   |
| ARGOMENTI  | ENTRATA | Ritardo   | U16  | Costante numerica indicante i µs di ritardo<br>Min 1 µs   Max 5000 µs. La precisione è del 5% |
|  | RITORNA | -   | -    | -   |
|  |         | A,H,X   | S08  | Invariati   |
| Va inserito in ..                                      |         | Moduli <i>Main</i> , <i>Boot</i> , <i>TimeEvent</i> |      |   |
| ESEMPI   |         | MicrosDelay   50   ←   Effettua 50µs di ritardo     |      |   |

| <b>DoEvery arg1,arg2</b>                           |                |  |             |   |
|--|----------------|--|-------------|---|
| <b>Esegue un gruppo di Istruzioni ogni N volte</b> |                |  |             |   |
| <b>ProtoType</b>                                   |                | <b>DoEvery N,Label</b>   |             |   |
|  |                | <b>Argomento</b>   | <b>Type</b> | <b>Descrizione</b>  |
| <b>ARGOMENTI</b>                                   | <b>ENTRATA</b> | <b>N</b>   | U08         | Variabile o costante numerica che indica ogni quante volte devono essere eseguite le Istruzioni indicate fino all'etichetta indicata in label |
|  |                | <b>Label</b>   | -           | Etichetta a cui dirama quando non viene eseguito il blocco di istruzioni.   |
|  | <b>RITORNA</b> | A,H,X  | S08         | Invariati   |
| <b>Va inserito in ..</b>                           |                | Modulo <i>TimeEvent</i>  |             |   |
| <b>ESEMPI</b>                                      |                | <b>DoEvery 5,exit</b> ← Esegue il gruppo di istruzioni fino alla etichetta exit ogni 5 volte<br><b>DoEvery step,exit</b> ← Esegue il gruppo di istruzioni fino alla etichetta exit ogni 'step' Volte (step è una variabile U8) |             |   |

| BitMan <i>arg1,arg2,arg3</i>                                  |         |   |           |   |
|---|---------|---|-----------|---|
| Manipola un flag (Bit) di una locazione di memoria o registro |         |   |           |   |
| ProtoType   |         | BitMan Mode,Flag,Location   |           |   |
|   |         | Argomento   | Type      | Descrizione   |
| ARGOMENTI   | ENTRATA | Mode  | Etichetta | Modo in cui manipolare il Flag/Bit:<br>SET=1, CLEAR=0 , TOGGLE=Inversione |
|   |         | Flag  | U08       | Variabile o Costante del Flag da modificare : 0...7                       |
|   |         | Location  | U08       | Registro o variabile in cui si vuole modificare il flag                   |
|   | RITORNA | -   | -         | -   |
|   |         | A,H,X   | S08       | Invariati   |
| NOTE  |         |   |           |   |
| ESEMPI  |         | BitMan TOGGLE,5,ptad ← Inverte il Bit 5 del registro ptad<br>BitMan SET,star,moon ← Mette a 1 il bit indicato in 'star' della locazione 'moon'<br>BitMan CLEAR,1,moon ← Mette a 0 il bit 1 della locazione 'moon' |           |   |

| BranchIfBit <i>arg1,arg2,arg3,arg4</i>                  |         |   |           |  |
|---|---------|---|-----------|--|
| Effettua una diramazione a seconda del valore di un Bit |         |   |           |  |
| ProtoType   |         | BranchIfBit <i>Status,Flag,Location ,Label</i>  |           |  |
|   |         | Argomento   | Type      | Descrizione  |
| ARGOMENTI   | ENTRATA | Status  | Etichetta | Stato del Flag/Bit da testare: SET / CLEAR<br>SET=1, CLEAR=0 |
|   |         | Flag  | U08       | Variabile o Costante del Flag da testare : 0...7             |
|   |         | Location  | U08       | Registro o variabile in cui si vuole testare il flag         |
|   | RITORNA | Label   | Etichetta | Etichetta a cui dirama se la condizione è vera               |
|   |         | -   | -         | -  |
|   |         | A,H,X   | S08       | Invariati  |
| NOTE  |         |   |           |  |
| ESEMPI  |         | BranchIfBit SET,5,ptad,stop ← Dirama a ‘stop’ se il bit 5 di ‘ptad’ è 1<br>BranchIfBit CLEAR,flag,star,stop ← Dirama a ‘stop’ se il bit indicato nella variabile flag della variabile ‘star’ è 0. |           |  |

| BranchIfValue <i>arg1,arg2,arg,arg4</i>                        |         |  |           |   |
|--|---------|--|-----------|---|
| Effettua una diramazione a seconda del valore di una variabile |         |  |           |   |
| ProtoType  |         | BranchIfValue <i>Variable,Condition,Value,Address</i>  |           |   |
|  |         | Argomento  | Type      | Descrizione   |
| ARGOMENTI  | ENTRATA | Variable   | U08       | Variabile in cui si vuole comparare il valore       |
|  |         | Condition  | Etichetta | Modo di comparazione: EQUAL - DIVERSE               |
|  |         | Value  | U08       | Valore/costante da comparare                        |
|  |         | Label  | Etichetta | Etichetta a cui dirama se si verifica la condizione |
|  | RITORNA | -  | -         | -   |
|  |         | A,H,X  | S08       | Invariati   |
| NOTE   |         |  |           |   |
| ESEMPI   |         | BranchIfValue mode,EQUAL,50,main.1 ← Dirama a 'main.1' se 'mode' = 50<br>BranchIfValue mode,DIVERSE,timer,main.1 ← Dirama a 'main.1' se 'mode' è<br>Diverso dal contenuto di 'timer' |           |   |



| GoTo <i>arg1</i>   |         |  |      |                                    |
|--|---------|--|------|------------------------------------|
| Salta in modo incondizionato ad un indirizzo o etichetta del programma |         |  |      |                                    |
| ProtoType  |         | GoTo Indirizzo   |      |                                    |
|  |         | Argomento  | Type | Descrizione                        |
| ARGOMENTI  | ENTRATA | Indirizzo  | U16  | Indirizzo o etichetta dove saltare |
|  | RITORNA | -  | -    | -                                  |
|  |         | A,H,X  | S08  | Invariati                          |
| NOTE   |         |  |      |                                    |
| ESEMPI   |         | GoTo main ← Salta all'etichetta 'main'<br>GoTo \$E080 ← Salta all'indirizzo \$E080 |      |                                    |

| Return  |         |           |      |                  |
|---|---------|-----------|------|------------------|
| Ritorna al istruzione successiva dopo aver svolto un sottoprogramma |         |           |      |                  |
| ProtoType   |         | Return    |      |                  |
|   |         | Argomento | Type | Descrizione      |
| ARGOMENTI   | ENTRATA | -         |      | Nessun argomento |
|   | RITORNA | -         | -    | -                |
|   |         | A,H,X     | S08  | Invariati        |
| NOTE  |         |           |      |                  |
| ESEMPI  |         |           |      |                  |

| <b>IncValue arg1,arg2,arg3,arg4,arg5</b>   |                |  |                |  |
|--|----------------|--|----------------|--|
| <b>Incrementa una variabile numerica fra un valore minimo e massimo in modo circolare.</b> |                |  |                |  |
| <b>ProtoType</b>   |                | <b>IncValue VarType,Variable,Max,Min,Ival</b>  |                |  |
|  |                | <b>Argomento</b>   | <b>Type</b>    | <b>Descrizione</b>   |
| <b>ARGOMENTI</b>   | <b>ENTRATA</b> | <b>VarType</b>   | Etichetta      | Tipo di variabile : U08 S08 U16 S16  |
|  |                | <b>Variable</b>  | U/S08<br>U/S16 | Variabile da incrementare  |
|  |                | <b>Max</b>   | U/S08<br>U/S16 | Valore Massimo raggiunto dalla variabile   |
|  |                | <b>Min</b>   | U/S08<br>U/S16 | Valore minimo di partenza. Se Min = Max il valore incrementato si ferma la valore massimo.             |
|  |                | <b>Ival</b>  | U08            | Valore da incrementare. Se manca viene considerato 1   |
|  | <b>RITORNA</b> | CARRY  | Flag           | 0 ← La variabile ha raggiunto il valore massimo<br>1 ← La variabile non ha raggiunto il valore massimo |
|  |                | A,H,X  | S08            | Invariati  |
| <b>NOTE</b>  |                | Una volta raggiunto il valore massimo, viene caricato il valore minimo   |                |  |
| <b>ESEMPI</b>  |                | <b>IncValue U08,time,59,0</b> ← Incrementa la var. 'time' di una unità fino a 59 e riparte da 0.<br><b>IncValue U16,speed,1000,200</b> ← Incrementa la var. 'speed' di una unità fino a 1000 e riparte da 200.<br><b>IncValue S16,span,8000,-5000</b> ← span variera' fra -5000 ...+8000<br><b>IncValue S16,span,spmax,spmin,10</b> ← spmax ed spmin sono variabili. Increm. di 10 |                |  |

| <b>DecValue arg1,arg2,arg3,arg4,arg5</b>   |                |   |                |  |
|--|----------------|---|----------------|--|
| <b>Decrementa una variabile numerica fra un valore minimo e massimo in modo circolare.</b> |                |   |                |  |
| <b>ProtoType</b>   |                | <b>DecValue VarType,Variable,Min,Max,Dval</b>   |                |  |
|  |                | <b>Argomento</b>  | <b>Type</b>    | <b>Descrizione</b>   |
| <b>ARGOMENTI</b>   | <b>ENTRATA</b> | <b>VarType</b>  | Etichetta      | Tipo di variabile : U08 S08 U16 S16  |
|  |                | <b>Variable</b>   | U/S08<br>U/S16 | Variabile da decrementare  |
|  |                | <b>Min</b>  | U/S08<br>U/S16 | Valore minimo raggiunto dalla variabile  |
|  |                | <b>Max</b>  | U/S08<br>U/S16 | Valore massimo di partenza. Se Max = Min il valore incrementato si ferma la valore minimo.             |
|  |                | <b>Dval</b>   | U08            | Valore da decrementare. Se manca viene considerato 1   |
|  | <b>RITORNA</b> | CARRY   | Flag           | 0 ← La variabile ha raggiunto il valore massimo<br>1 ← La variabile non ha raggiunto il valore massimo |
|  |                | A,H,X   | S08            | Invariati  |
| <b>NOTE</b>  |                | Una volta raggiunto il valore minimo, viene caricato il valore massimo  |                |  |
| <b>ESEMPI</b>  |                | <b>DecValue U08,time,0,59</b> ← Decrementa la var. 'time' di una unità fino a 0 e riparte da 59.<br><b>DecValue U16,speed,200,1000</b> ← Decrementa la var. 'speed' di una unità fino a 200 e riparte da 1000.<br><b>DecValue S16,span,-5000,-1000</b> ← span variera' fra -5000 ...-1000<br><b>DecValue S16,span,spmin,spmax,10</b> ← spmin ed spmax sono variabili. Decr. di 10 |                |  |

| MoveTeTable <i>arg1,arg2,arg3</i>                                      |         |  |           |   |
|--|---------|--|-----------|---|
| Copia un elemento con terminatore \$00 da una tabella in una variabile |         |  |           |   |
| ProtoType  |         | MoveTeTable Table,element,Variable   |           |   |
|  |         | Argomento  | Type      | Descrizione   |
| ARGOMENTI  | ENTRATA | Table  | Etichetta | Indirizzo o etichetta della tabella   |
|  |         | Element  | U08       | Numero dell elemento da muovere 0..255<br>Imserire A o X per i Registri MCU |
|  |         | Variable   | U08..U32  | Nome della variabile in cui spostare l'elemento                             |
|  | RITORNA | X  | Flag      | Lunghezza dell'elemento   |
|  |         | A,H  | I8        | Unknown   |
| NOTE   |         |  |           |   |
| ESEMPI   |         | MoveTeTable ftable,3,fsignal ← Copia l'elemento N.3 dalla tabella 'ftable' nella variabile 'fsignal'<br>MoveTeTable ftable,step,fsignal ← Copia l'elemento 'step' dalla tabella 'ftable' nella variabile 'fsignal' |           |   |

| <b>MoveEITable</b> <i>arg1,arg2,arg3,arg4</i>   |                |  |             |   |
|---|----------------|--|-------------|---|
| <b>Copia un elemento da una tabella in una variabile dichiarando la lunghezza dell'elemento</b> |                |  |             |   |
| <b>ProtoType</b>  |                | <b>MoveTeTable</b> <i>Table,Length,element,Variable</i>  |             |   |
|   |                | <b>Argomento</b>   | <b>Type</b> | <b>Descrizione</b>  |
| <b>ARGOMENTI</b>  | <b>ENTRATA</b> | <b>Table</b>   | Etichetta   | Indirizzo o etichetta della tabella   |
|   |                | <b>Lenght</b>  | U08         | Lunghezza di ogni elemento della tabella                                    |
|   |                | <b>Element</b>   | U08         | Numero dell elemento da muovere 0..255<br>Imserire A o X per i Registri MCU |
|   |                | <b>Variable</b>  | U08..U32    | Nome della variabile in cui spostare l'elemento                             |
|   | <b>RITORNA</b> | X  | Flag        | Lunghezza dell'elemento   |
|   |                | A,H  | I8          | Unknown   |
| <b>NOTE</b>   |                |  |             |   |
| <b>ESEMPI</b>   |                | <b>MoveEITable</b> <i>fTable,4,7,fSignal</i> ← Copia l'elemento N.7 dalla tabella 'fTable' composta da elementi di lunghezza 4 bytes nella variabile 'fSignal'<br><b>MoveTeTable</b> <i>fTable,4,step,fSignal</i> ← Copia l'elemento 'step' dalla tabella 'fTable' nella variabile 'fSignal' |             |   |

| WaitEvent  arg1,arg2,arg3,arg4   |         |  |           |   |
|--|---------|--|-----------|---|
| Attende per un determinato tempo che si verifichi un evento segnalato da un Flag |         |  |           |   |
| ProtoType  |         | WaitEvent  Flag, Location,Status,Timeout   |           |   |
|  |         | Argomento  | Type      | Descrizione   |
| ARGOMENTI  | ENTRATA | Flag   | U08       | Costante del Flag da testare : 0...7                                    |
|  |         | Location   | U08       | Registro o variabile in cui si vuole testare il flag  **                |
|  |         | Status   | Etichetta | Stato del Flag/Bit da testare: SET / CLR<br>SET=1,  CLR=0               |
|  |         | Timeout  | U16       | Tempo in ms di attesa evento  |
|  | RITORNA | C (CCR)  | Flag      | 0 ← Evento avvenuto   1 ← Evento non avvenuto entro il tempo dichiarato |
|  |         | A,H,X  | S08       | Invariati   |
| NOTE   |         | ** Location deve essere in pagina Zero   |           |   |
| Va inserito in ..  |         | Moduli   Main  |           |   |
| ESEMPI   |         | WaitEvent   5,ptad,SET,50           ← Attende per 50 ms che il Bit 5 di ptad vada alto<br><br>WaitEvent   ECHO,mode,CLR,30   ← Attende per 30 ms che il Bit  ECHO di mode vada a zero. |           |   |

| CreateTimeEvent <i>arg1</i> |         |  |       |                   |
|-----------------------------|---------|--|-------|-------------------|
| Crea un evento temporale    |         |  |       |                   |
| ProtoType                   |         | CreateTimeEvent <i>Event</i>   |       |                   |
|                             |         | Argomento  | Type  | Descrizione       |
| ARGOMENTI                   | ENTRATA | Event  | LABEL | Nome dell' Evento |
|                             |         |  |       |                   |
|                             | RITORNA | -  |       |                   |
|                             |         | A,H,X  | S08   | Invariati         |
| Va inserito in ..           |         | Structures    o    MyData  |       |                   |
| NOTE                        |         | <i>CreateTimeEvent</i> non è strettamente necessario poichè l'evento può essere creato nello stesso momento della sua generazione in <i>GenTimeEvent</i> in questo caso però l'evento può essere intercettato con <i>ServiceTimeEvent</i> solo nel modulo <b>Main</b><br>Creando prima l'evento invece è possibile intercettare l'evento anche nei moduli o files inclusi nel <b>Modules</b> |       |                   |
| ESEMPI                      |         | CreaTimeEvent    STAR    ← Crea l'evento di tempo STAR   |       |                   |

| GenTimeEvent <i>arg1,arg2,arg3</i>                                 |         |  |       |  |
|--|---------|--|-------|--|
| Genera un evento di tempo ogni N volte una specifica Base di tempi |         |  |       |  |
| ProtoType  |         | GenTimeEvent <i>Event, N,Mode</i>  |       |  |
|  |         | Argomento  | Type  | Descrizione  |
| ARGOMENTI  | ENTRATA | Event  | LABEL | Nome dell' Evento  |
|  |         | N  | U08   | Costante o Variabile che indica ogni quanto tempo l'evento sarà generato secondo questa relazione:<br>Tempo= N*BaseTempi   |
|  |         | Mode   | LABEL | WAIT : Attende che l'evento sia servito prima di conteggiare il successivo<br>GOON: Continua a conteggiare per l'evento successivo anche se l'evento corrente non è ancora servito |
|  | RITORNA | -  |       |  |
|  |         | A,H,X  | S08   | Invariati  |
| Va inserito in ..  |         | Solo nel modulo <i>TimeEvent</i>   |       |  |
| ESEMPI   |         | <i>TimerSvc.100ms:</i><br><b>GenTimeEvent    STAR,50,GOON</b> ← Genera l'evento di tempo STAR ogni 5 sec senza attendere che questo sia servito conteggiando subito per il successivo.<br><i>Return</i><br><br><i>TimerSvc.10ms:</i><br><b>GenTimeEvent    STAR,50,WAIT</b> ← Genera l'evento di tempo STAR ogni 500 msec aspettando che l'evento sia servito prima di generare il successivo<br><i>Return</i> |       |  |

| ServiceEvent <i>arg1,arg2</i>                     |         |  |       |   |
|---|---------|--|-------|---|
| Serve l'evento indicato se questo si è verificato |         |  |       |   |
| ProtoType   |         | ServiceEvent <i>Event, Address</i>   |       |   |
|   |         | Argomento  | Type  | Descrizione                                       |
| ARGOMENTI   |         | Event  | LABEL | Nome dell' Evento                                 |
|   | ENTRATA | Address  | LABEL | Etichetta di Indirizzo dove l'evento sarà servito |
|   | RITORNA | -  |       |   |
|   |         | A,H,X  | S08   | Invariati   |
| Va inserito in ..                                 |         | Modulo <i>Main</i>   |       |   |
| ESEMPI  |         | ServiceEvent    STAR,StarService    ← Va a StarService se l'evento di tempo STAR si verifica |       |   |

| SetTimeEvent    arg1,arg2            |         |   |       |  |
|--------------------------------------|---------|---|-------|--|
| Setta lo stato di un evento di tempo |         |   |       |  |
| ProtoType                            |         | SetTimeEvent    Event, Status   |       |  |
|                                      |         | Argomento   | Type  | Descrizione  |
| ARGOMENTI                            | ENTRATA | Event   | LABEL | Nome dell' Evento  |
|                                      |         | Status  | LABEL | START: avvia la generazione dell'evento<br>STOP : ferma la generazione dell'evento |
|                                      | RITORNA | -   |       |  |
|                                      |         | A,H,X   | S08   | Invariati  |
| Va inserito in ..                    |         | Modulo    Boot    Main    Modules   |       |  |
| NOTA                                 |         | Per default l'evento è sempre avviato dopo la sua generazione   |       |  |
| ESEMPI                               |         | SetTimeEvent    STAR,STOP    ← Arresta la generazione di eventi STAR<br>SetTimeEvent    STAR,START    ← Riavvia la generazione di eventi STAR |       |  |

| GetTimeEvent    arg1,arg2,arg3                  |                   |  |                                  |   |
|---|-------------------|--|----------------------------------|---|
| Legge il valore del Timer di un evento di tempo |                   |  |                                  |   |
| ProtoType                                       |                   | GetTimeEvent    Event, MaxTime, Destination  |                                  |   |
|   |                   | Argomento  | Type                             | Descrizione   |
| ARGOMENTI                                       | ENTRATA           | Event  | LABEL                            | Nome dell' Evento   |
|   |                   | MaxTime  | U08                              | Valore Massimo raggiunto dal Timer.<br>Se assente la lettura è il conteggio in avanti:<br>1,2,3....MaxTime<br>Se presente la lettura è il conteggio indietro:<br>MaxTime,MaxTime-1,MaxTime-2 .... |
|   |                   | Destination  | U08                              | Se presente : variabile del conteggio attuale   |
|   | RITORNA           | A  | U08                              | Valore del Conteggio  |
|   |                   | H,X  | S08                              | Invariati   |
|   | Va inserito in .. |  | Modulo    Main    Modules    ... |   |
| NOTA  |                   |  |                                  |   |
| ESEMPI  |                   | GetTimeEvent    STAR,MAX    ← A < Count Down del Timer STAR<br>GetTimeEvent    STAR,MAX,alfa    ← A & alfa < Count Down del Timer STAR |                                  |   |

| BranchIfTimeEvent <i>arg1,arg2,arg3</i>         |         |  |       |  |
|---|---------|--|-------|--|
| Legge il valore del Timer di un evento di tempo |         |  |       |  |
| ProtoType                                       |         | BranchTimeEvent <i>Event, Mode,Address</i>   |       |  |
|   |         | Argomento  | Type  | Descrizione  |
| ARGOMENTI                                       | ENTRATA | Event  | LABEL | Nome dell' Evento  |
|   |         | Mode   | LABEL | ON   : effettua il Branch se il Timer è attivo<br>OFF : effettua il Branch se il Timer è in STOP |
|   |         | Destination  | U08   | Indirizzo o etichetta del Branch   |
|   | RITORNA | -  |       |  |
|   |         | A,H,X  | S08   | Invariati  |
| Va inserito in ..                               |         | Modulo <i>Main    Modules</i> ...  |       |  |
| NOTA  |         |  |       |  |
| ESEMPI  |         | BranchIfTimeEvent <i>STAR,ON,main.5</i> ← Dirama a main.5 se Timer STAR attivo<br>GetTimeEvent <i>STAR,OFF,main5</i> ← Dirama a main.5 se Timer STAR Fermo |       |  |



| CreateFlashLedDaemon <i>arg1</i>                   |         |  |       |                                       |
|--|---------|--|-------|---------------------------------------|
| Crea un Daemon temporale per il Flashing di un Led |         |  |       |                                       |
| ProtoType  |         | CreateFlashLedDaemon <i>LedName</i>  |       |                                       |
|  |         | Argomento  | Type  | Descrizione                           |
| ARGOMENTI  | ENTRATA | LedName  | LABEL | Nome del Led che effettua il Flashing |
|  |         |  |       |                                       |
|  | RITORNA | -  |       |                                       |
|  |         | A,H,X  | S08   | Invariati                             |
| Va inserito in ..                                  |         | Structures o MyData  |       |                                       |
| NOTE   |         | <i>CreateFlashLedDaemon</i> non è strettamente necessario poichè l'evento può essere creato nello stesso momento della sua generazione in <i>DaemonFlashLed</i> in questo caso però il Led può essere gestito con <i>FlashLed</i> solo nel modulo <b>Main</b><br>Creando prima l'evento invece è possibile intercettare l'evento anche nei moduli o files inclusi nel <b>Modules</b> |       |                                       |
| ESEMPI   |         | <b>CreaFlashEventDaemon</b> <b>SYSLED</b> ← Il Led SYSLED è gestito in background da un Daemon per il Flashing   |       |                                       |

| DaemonFlashLed <i>arg1</i>                          |         |  |       |   |
|---|---------|--|-------|---|
| Serve il Daemon temporale per il Flashing di un Led |         |  |       |   |
| ProtoType   |         | DaemonFlashLed <i>LedName</i>  |       |   |
|   |         | Argomento  | Type  | Descrizione                                     |
| ARGOMENTI   | ENTRATA | LedName  | LABEL | Nome del Led gestito dal Daemon per il flashing |
|   |         |  |       |   |
|   | RITORNA | -  |       |   |
|   |         | A,H,X  | S08   | Invariati                                       |
| Va inserito in ..                                   |         | Solo in <i>TimeEvent</i> (consigliato alla base di tempi di 100msec)             |       |   |
| NOTE  |         |  |       |   |
| ESEMPI  |         | DaemonFlashEvent <b>SYSLED</b> ← Gestione del Daemon per il Flash del Led SYSLED |       |   |

| FlashLed  arg1,arg2,arg3,arg4,arg5          |         |  |              |  |
|---|---------|--|--------------|--|
| Controlla la modalità di Flashing di un Led |         |  |              |  |
| ProtoType                                   |         | FlashLed  LedName,ONtime,OFFtime,Flash,Mode  |              |  |
|   |         | Argomento  | Type         | Descrizione  |
| ARGOMENTI                                   | ENTRATA | LedName  | LABEL        | Nome del Led gestito dal Daemon per il flashing  |
|   |         | ONtime   | U08          | Unità di tempo base in cui il led è acceso   |
|   |         | OFFtime  | U08          | Unità di tempo base in cui il  led è spento  |
|   |         | Flash  | U08<br>LABEL | Numero di Flash da effettuare<br>GOON: Flash Continuo fino a che non viene eseguito un |
|   |         | Mode   | LABEL        | Opzionale<br>WAIT: attende fino a che la sequenza di Flash è terminata                 |
|   | RITORNA | -  |              |  |
|   |         | A,H,X  | S08          | ?  |
| Va inserito in ..                           |         | Boot  Main  Module ...   |              |  |
| NOTE  |         |  |              |  |
| ESEMPI                                      |         | Se il Daemon è inserito nella base tempi di 100mS<br><br>FlashLed  SYSLED,2,5,5,WAIT   ← Il Led SYSLED effettua 5 Flash  200mS ON e 500mS OFF. Attende che i 5 Flash terminino.<br>FlashLed  SYSLED,2,5,5           ← Come sopra ma senza attesa<br>FlashLed  SYSLED,2,5,GOON      ← Come sopra senza attesa con flashing continuo |              |  |

| KillFlashLed <i>arg1</i>                             |         |  |       |   |
|--|---------|--|-------|---|
| Arresta il Daemon che effettua il Flashing di un Led |         |  |       |   |
| ProtoType  |         | KillFlashLed <i>LedName</i>  |       |   |
|  |         | Argomento  | Type  | Descrizione                                     |
| ARGOMENTI  | ENTRATA | LedName  | LABEL | Nome del Led gestito dal Daemon per il flashing |
|  |         |  |       |   |
|  | RITORNA | -  |       |   |
|  |         | A,H,X  | S08   | Invariati                                       |
| Va inserito in ..                                    |         | Boot Main Modules  |       |   |
| NOTE   |         | KillFlashLed viene usato quando il Led effettua il Flash in modalità continua GOON<br>Con lo scopo di spegnerlo. |       |   |
| ESEMPI   |         | KillFlashEvent SYSLED ← Arresta il Daemon SYSLED   |       |   |