Sommario

SY10EnterCriticalSection 3
IMPLEMENTAZIONE 3

SY10EnterCriticalSection

SY = *Funzione di sistema*

La funzione SY10EnterCriticalSection fa entrare l'unità task chiamante in una sezione critica.

IMPLEMENTAZIONE

SY10EnterCriticalSection(lookPool)

Parametri:

IN/OUT	TIPO VARIABILE	NOME DI ESEMPIO	DIM	
IN	CONST	unitID	L	ID unità task chiamante
IN	ARRGBL	pool	B	Pool informazioni sezione critica
IN/OUT	GLOBAL	locked	F	Stato di lock (0 = unlocked, 1 = locked)
IN	CONST	blocking	L	Modalità di blocco (0 = non bloccante, 1 = bloccante)

Descrizione:

Tramite la chiamata SY10EnterCriticalSection la unit task chiamante chiede al sistema di entrare in una sezione critica.

Se la sezione critica è libera verrà subito assegnata al task chiamante ritornando lo stato locked a 1. Se l'unità task che richiede di entra nella sezione critica vi è già entrata la funzione ritornerà stato locked 1 incrementando il contatore di richieste per il task in corso.

Se invice la sezione critica è già occupata da un'altra unità task e la modalità di blocking è posta a 0 si avrà l'uscita immediata con stato locked a 0.

altrimenti in base al parametro blocking si avrà il ritorno con stato locked a 0 oppure la unità task verrà messa in attesa ed eseguito automaticamente uno cambio di contesto.

Esempio

Nell'esempio viene inizializzato un pool informazioni utilizzato poi per una sezione critica.

unità task:

```
;===
; Unit B
;
GLOBAL
    Counter         L    INOUT

; local variables
CONST
    UNIT_ID            2

GLOBAL
    locked            F
    looping            L

; main entry point
BEGIN
    CALL TASK_INIT
    WHILE TRUE
        CALL TASK_EXECUTE
        WAIT A_LOOP
    ENDWHILE
END

;===
; Task initialization
;
SUB TASK_INIT
    WAIT INIT.Initialized
ENDSUB

;===
; Task execution
;
SUB TASK_EXECUTE

    SY10EnterCriticalSection(UNIT_ID, LockPool, locked, TRUE)
    CALL COUNTER_ADD
    SY10LeaveCriticalSection(UNIT_ID, LockPool, locked, TRUE)

ENDSUB

;===
; Counter add
;
SUB COUNTER_ADD
    Counter = Counter + 1

    ; this simulate an automatic context switch for a device write and read access
    WAIT A_LOOP
ENDSUB
```

Documento generato automaticamente da **Qem Wiki** - https://wiki.qem.it/
Il contenuto wiki è costantemente aggiornato dal team di sviluppo, è quindi possibile che la versione online contenga informazioni più recenti di questo documento.