20 09 2023 PROGETTO

Un processore z64 controlla il sistema di irrigazione di un giardino. L'impianto utilizza una periferica PROGRAMMATORE che permette all'utente di specificare l'orario di accensione dell'impianto e la durata dell'irrigazione. Entrambi questi valori sono rappresentati come interi a 32 bit. Nel caso dell'orario di accensione, l'intero descrive il numero di secondi a partire dalla mezzanotte dopo i quali si intende avviare l'impianto. La durata di irrigazione è anch'essa espressa come numero di secondi.

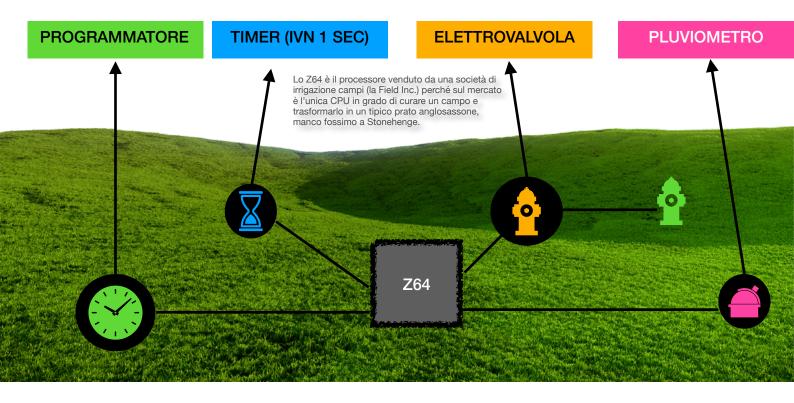
La periferica TIMER invia una richiesta di interruzione al processore a cadenza di un secondo. Alla ricezione della richiesta di interruzione, il processore determina se si è raggiunto l'orario di accensione. In caso affermativo, lo z64 programmerà la periferica ELETTROVALVOLA per attivare l'irrigazione. Al termine dell'intervallo richiesto dall'utente, la periferica ELETTROVALVOLA sarà programmata per arrestare l'irrigazione.

Una periferica PLUVIOMETRO permette allo z64 di determinare se sono avvenute precipitazioni nel recente passato. Tale periferica sincrona fornisce il numero di mm d'acqua caduti nelle 24 ore precedenti alla lettura. L'intervallo di irrigazione richiesto dall'utente viene modificato in funzione della quantità d'acqua caduta, secondo la seguente logica:

- Viene lasciato inalterato se sono caduti meno di 48 mm d'acqua;
- Viene dimezzato se sono caduti almeno 48 mm d'acqua ma non più di 96 mm;
- Viene ridotto a un quarto se sono caduti più di 99 mm d'acqua.

Realizzare:

- Le interfacce delle periferiche PROGRAMMATORE, TIMER e PLUVIOMETRO (non è richiesta l'interfaccia di ELETTROVALVOLA);
- Tutto il software necessario al funzionamento del sistema, comprensivo di eventuali driver.



Descrizione in "linea di massima" delle periferiche.

La periferica **PROGRAMMATORE** è un'interfaccia di input in Busy Waiting poiché internamente al suo registro di interfaccia sarà specificato l'orario di accensione dell'impianto con la relativa durata dell'irrigazione.

Ovviamente per scelta di progetto si assume che i dati siano già caricati all'interno del registro cosicché si possa semplicemente prelevare il valore di 8 byte (i primi 4 byte rappresentano l'orario di accensione dell'impianto e gli ultimi 4 byte la durata dell'irrigazione) per consentire alla CPU di poter trattare i dati stessi.

La specifica ricorda che: "Nel caso dell'orario di accensione, l'intero descrive il numero di secondi a partire dalla mezzanotte dopo i quali si intende avviare l'impianto. La durata di irrigazione è anch'essa espressa come numero di secondi".

La periferica **TIMER** è un'interfaccia che funziona con le interruzioni vettorizzate sprovvista di registri di interfaccia interni poiché all'atto dell'interruzione (tramite un driver) il processore verifica se si è raggiunto l'orario di accensione insito nel registro di interfaccia della periferica di input PROGRAMMATORE.

La periferica **ELETTROVALVOLA** è un'interfaccia di output in Busy Waiting poiché internamente al suo registro di interfaccia il processore scriverà una variabile booleana codificata ad otto bit per attivare/ disattivare l'irrigazione (1:attiva,0:disattiva). *Non verrà specificata nel disegno delle interfacce*.

La periferica **PLUVIOMETRO** è un'interfaccia di input in Busy Waiting alla cui lettura dell'apposito registro di interfaccia fornisce il numero di millilitri d'acqua caduti nelle ultime 24 ore.

```
.org 0x800
.data
       .equ $STATUS PROGRAMMATORE, 0x0001
       .equ $REG_PROGRAMMATORE, 0x0002
       .equ $IRQ_TIMER, 0x0003
       .equ $STATUS TIMER, 0x0004
       .equ $STATUS_ELETTROVALVOLA, 0x0005
       .equ $REG_ELETTROVALVOLA, 0x0006
       .equ $REG PLUVIOMETRO, 0x0007
       .equ $STATUS_PLUVIOMETRO, 0x0008
       dati: .quad 0
       orario accensione impianto: .quad 0
       durata irrigazione: .quad 0
       milliltri caduti: .quad 0
.text
main:
       sti
 .acquisizione:
       outb %al, $STATUS PROGRAMMATORE
       outb %al, $STATUS_PLUVIOMETRO #si presume che PLUVIOMETRO e PROGRAMMATORE rendano disponibili i dati
nell'esatto momento (risparmiando un ciclo di Busy Waiting)
 .bw:
       inb %STATUS_PROGRAMMATORE, %al
       btb $0, %al
       jnc .bw
       ing $REG PROGRAMMATORE, %rax
       movq %rax, dati
       ing $REG_PLUVIOMETRO, %rax
       movq %rax, millilitri caduti
       cmpq millilitri_caduti, $48 #se la sorg > dest
       jc .maggiore 48
  .maggiore 48:
       cmpq millilitri_caduti, $96 #se la dest > sorg
       inc .minore 96
       cmpq millilitri caduti, $99 #se la dest > sorg
       inc .maggiore_99
  .minore 96:
       shrq $2, durata_irrigazione
  .maggiore_99:
       shrq $4, durata irrigazione
       #in dati è presente l'orario di accensione dell'impianto (primi 4 byte) e la durata irrigazione.
       #utilizzo una maschera per scoprire l'orario di accensione effettuando un "bit-extraction".
andq $0xFFFFFFF00000000, dati
       movq dati, orario_accensione_impianto
       #utilizzo una maschera di bit per scoprire la durata di irrigazione.
andq $0x0000000FFFFFFFF, %rax
       movq %rax, durata_irrigazione
       outb %al, $STATUS_TIMER #la periferica timer è pronta per essere interrotta
       #ora il timer è a cadenza di 1 secondo
       imp .acquisizione
  .driver 0
       outb %al, $IRQ_TIMER #cancello la causa di interruzione
       outb %al, $STATUS_TIMER #la periferica timer è pronta per essere interrotta [DISPONIBILE PER INTERRUZIONE]
       iret
  check:
       subg $1, orario_accensione_impianto #quando arriva a zero allora attiva la peri. ELETTROVALVOLA
       cmpq $0, orario_accensione_impianto
       jz .attiva_elettrovalvola
       ret
  .attiva_elettrovalvola:
       outb $1, $REG_ELETTROVALVOLA #attiva elettrovalvola
       outb %al, $STATUS_ELETTROVALVOLA #avvio la periferica e notifico che c'è un dato da consumare
       inb $STATUS_ELETTROVALVOLA, %al
       btb $0, %al
       inc .bw1
```

#il dato l'ha consumato - elettrovalvola attivata

.ciclo:

subq \$1, durata_irrigazione #quando arriva a zero allora elettrovalvola deve essere spenta cmpq \$0, durata_irrigazione jz .disattiva_elettrovalvola:

.disattiva elettrovalvola:

outb \$0, \$REG_ELETTROVALVOLA outb %al, \$STATUS_ELETTROVALVOLA #avvio la periferica e notifico che c'è un dato da consumare bw2: inb \$STATUS_ELETTROVALVOLA, %al btb \$0, %al jnc .bw2 #il dato l'ha consumato - elettrovalvola disattivata ret

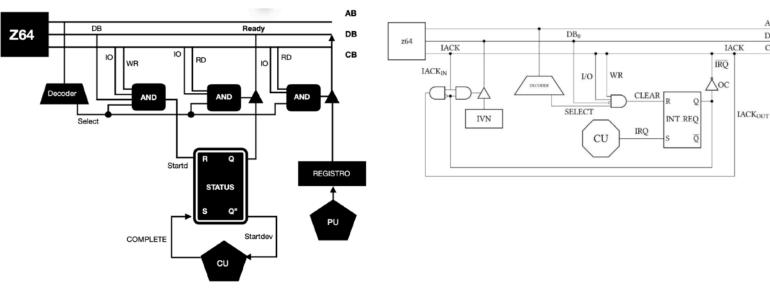
Di seguito vengono rappresentate le interfacce:

PROGRAMMATORE - Interfaccia Busy Waiting di input

TIMER - Interfaccia IRQ senza registri di interfaccia.

DB

CB



PLUVIOMETRO - Interfaccia Busy Waiting di input.

