

Autovelox

Partecipanti gruppo: Sudati Simone (mat. 104xxxx), Rebucini Leonardo (mat. 104xxx) e Baldi Federico (mat. 104xxxx)

- **Codice Sorgente:**

```
.data
IN:  .byte  0X00
OUT: .byte  0x00
.text
    la      $a0, IN
    la      $a1, OUT
CFS:  lb     $t0, 0($a0)
      andi   $t0, $t0, 0x80
      beq    $t0, $zero, CFS #controllo primo sensore
      add    $t5, $zero, $zero #inizializzazione contatore istruzioni
RS:   lb     $t0, 0($a0)
      andi   $t0, $t0, 0x40
      addi   $t5, $t5, 4      #aumento contatore istruzioni
      beq    $t0, $zero, RS   #controllo secondo sensore
      li     $t4, 36000000
      slt    $t0, $t4, $t5    #caso velocità < 50km/h (13,6m/s)
      bne    $t0, $zero, ENDV1
      li     $t2, 250000000 #attesa di 1s per lo scatto della foto
      jal    WAIT
      li     $t1, 0x08        #segnale per l'apertura dell'otturatore (linea 3 out da 0 a 1)
      sb     $t1, 0($a1)
      li     $t2, 62500000    #attesa di 250ms per la chiusura dell'otturatore
      jal    WAIT
      li     $t1, 0x00        #segnale per la chiusura dell'otturatore (linea 3 out da 1 a 0)
      sb     $t1, 0($a1)
      li     $t4, 32750000
      slt    $t0, $t4, $t5    #caso velocità < 55km/h (15,27m/s)
      bne    $t0, $zero, ENDV2
      li     $t4, 30000000
      slt    $t0, $t4, $t5    #caso velocità < 60km/h (16,6m/s)
      bne    $t0, $zero, ENDV3
      j      ENDV4            #caso velocità > 60km/h (16,6m/s)
RES:  sb     $t1, 0($a1)      #salvataggio della velocità in memoria
      jr     $ra
                                     ##procedura WAIR
WAIT: addi   $t2, $t2, -1     #verrà applicato un ritardo VAL = $t2
      bne    $t2, $zero, WAIT
      jr     $ra
ENDV1: li    $t1, 0x00        #valore in memoria nel caso v < 50km/h
      jal    RES
      j      CFS
```

| | | |
|----------|------------|---|
| ENDV2:li | \$t1, 0x10 | #valore in memoria nel caso 50km/h < v < 55km/h |
| jal | RES | |
| j CFS | | |
| ENDV3:li | \$t1, 0x20 | #valore in memoria nel caso 55km/h < v < 60km/h |
| jal | RES | |
| j CFS | | |
| ENDV4:li | \$t1, 0x30 | #valore in memoria nel caso v > 60km/h |
| jal | RES | |
| j CFS | | |

• **Commento codice sorgente:**

Il codice svolge queste funzioni:

Nelle primissime righe vengono definite le aree di memoria per le linee di input e output.

Poi, tra le righe 8 e 10 c'è il ciclo del primo sensore, che attende il rilevamento sul canale di input.

Quando viene interrotto questo primo ciclo, si passa al secondo, dove viene contato il numero di istruzioni trascorse tra il primo rilevamento ed il secondo, che causerà l'uscita da quest'ultimo ciclo.

Si passa quindi al primo controllo, dove viene verificato che la velocità sia minore di 50, in quanto questo è il più importante. Nel caso sia minore, allora si passa alla fine, dove restituiamo sul canale di out il valore corretto. Nell'altro caso la velocità è oltre il limite, quindi attendiamo un secondo, inviamo sul canale di out il segnale per aprire l'otturatore, attendiamo altri 250ms e cambiamo il segnale in out per far chiudere l'otturatore.

Chiuso l'otturatore facciamo gli ultimi controlli che verificano il range di velocità nel quale cade il veicolo, quindi tra 50 e 55, tra 55 e 60 e sopra i 60. In ciascuno di questi casi assegniamo il valore corretto nel canale di out da salvare in memoria.

Una volta salvato il dato, il programma torna ad aspettare un input sul primo sensore.

• **Registri utilizzati:**

- \$a0 contiene l'indirizzo della cella a 8 bit per l'input
- \$a1 contiene l'indirizzo della cella a 8 bit per l'output
- \$t0 è il registro utilizzato nelle operazioni di lettura nell'indirizzo contenuto in \$a0
- \$t1 è il registro utilizzato nelle operazioni di scrittura nell'indirizzo contenuto in \$a1
- \$t5 è il contatore utilizzato per ricavare il tempo tra il rilevamento di un sensore e l'altro (velocità)
- \$t4 è il registro che contiene il tempo in funzione del numero di istruzioni nelle velocità prestabilite (50, 55 e 60km/h)
- \$t2 viene utilizzato nel ciclo wait per ottenere il ritardo voluto

- **Canali input-output:**

- **Canale IN:**

| | | | | | | | |
|----|----|---|---|---|---|---|---|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| S1 | S2 | 0 | 0 | 0 | 0 | 0 | 0 |

- **Canale OUT:**

| | | | | | | | |
|---|---|-----|-----|------|---|---|---|
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 0 | 0 | Vel | Vel | Foto | 0 | 0 | 0 |

- **Formule e calcoli:**

Formule:

VAL per la procedura WAIT:
$$\text{Val ritardo} = \frac{T(s)}{n^{\circ} \text{istruzioni}} * 500MHz$$

Velocità in secondi al metro:
$$V(s/m) = \frac{1}{V(\frac{m}{s})}$$

Tempo per una singola istruzione:
$$Ti(\frac{s}{ist}) = \frac{1}{500MHz}$$

Istruzioni al metro alla velocità V:
$$Nist = \frac{V(\frac{s}{m})}{Ti(\frac{s}{ist})}$$

Conversioni:

$$50 \text{ km/h} = 13.889 \text{ m/s} = 0.072 \text{ s/m}$$

$$55 \text{ km/h} = 15.278 \text{ m/s} = 0.0655 \text{ s/m}$$

$$60 \text{ km/h} = 16.667 \text{ m/s} = 0.06 \text{ s/m}$$

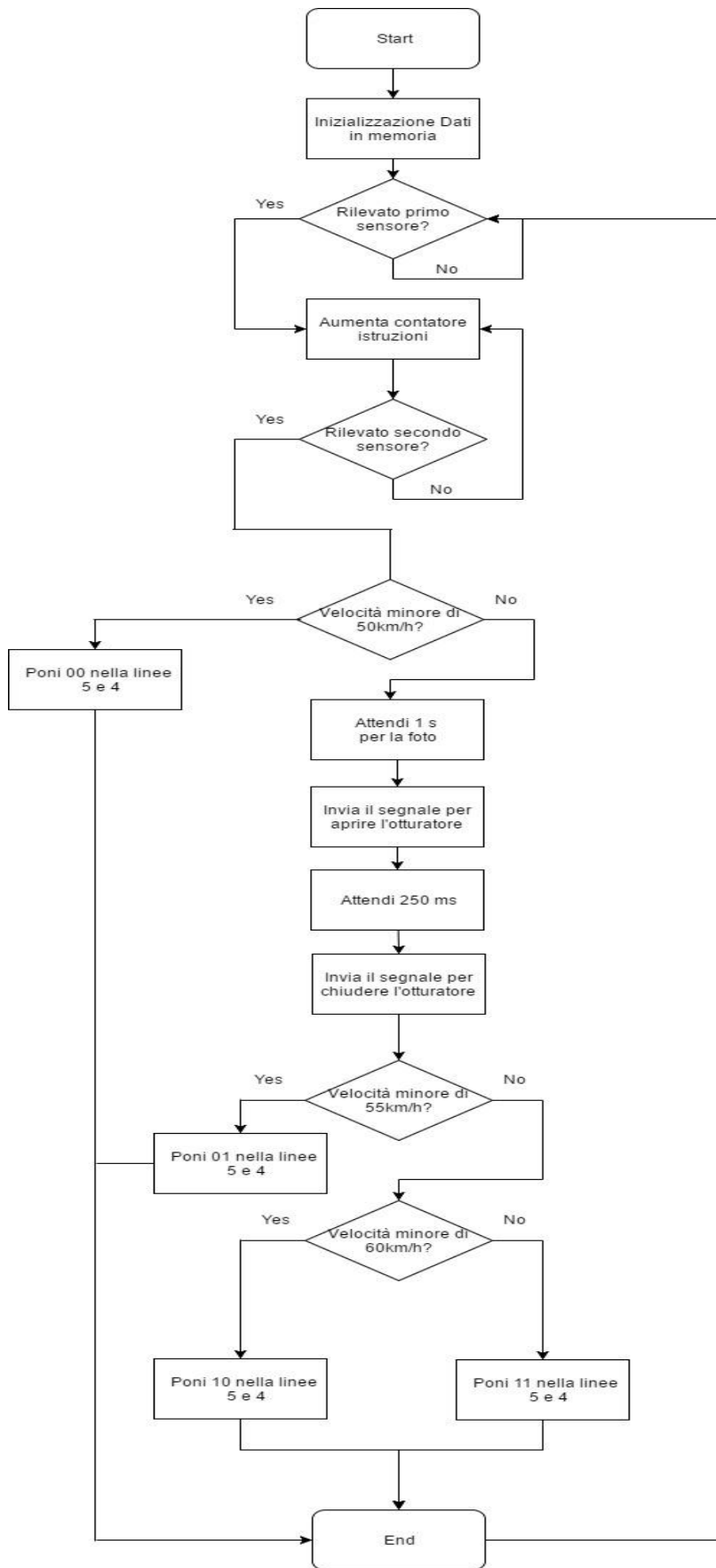
Calcoli:

$$VAL = \frac{1s}{2} * 500MHz = 250\,000\,000 \quad VAL = \frac{0.250s}{2} * 500MHz = 62\,500\,000$$

$$Ti(\frac{s}{ist}) = \frac{1}{500MHz} = 0.000\,000\,002$$

$$Nist = \frac{0.072(\frac{s}{m})}{0.000\,000\,002(\frac{s}{ist})} = 36\,000\,000 \quad Nist = \frac{0.0655(\frac{s}{m})}{0.000\,000\,002(\frac{s}{ist})} = 32\,750\,00 \quad Nist = \frac{0.06(\frac{s}{m})}{0.000\,000\,002(\frac{s}{ist})} = 30\,000\,000$$

- FlowChart:



- **Test del codice sorgente:**

Il codice è stato testato nel simulatore PC-Spim, utilizzando i valori di clock indicati nel testo.

La prima operazione di ogni test è l'assegnamento dell'indirizzo del registro pc al programma caricato.