**Relazione Progetto Modulo Di Calcolatori Elettronici**

**Esercizio 004-17-18**

**Autori:** Pirotta Nicola, Boffelli Jacopo, Cantoni Giorgio

**Processore:** MIPS R3000 (clock pari a 250 MHz)

**Linguaggio:** Assembly

**Simulatore:** PCSpim

**IPOTESI:**

Una macchina passa tra due sensori. Nel momento del passaggio il primo sensore da 0 passa ad 1 e nel momento d’uscita il secondo sensore da 0 passa ad 1. Il programma si basa su ***IN\_OUT*** che possiede ***16 bit*** che servono per gestire nel modo corretto la verifica del passaggio del veicolo, la verifica dell’uscita del veicolo, i vari confronti per le velocità e per la fotocamera. In particolare:

● Il ***bit 15*** gestisce l’entrata della macchina ( assume il valore: 0 se la macchina non è passata, 1 se la macchina è passata )

● Il ***bit 14*** gestisce l’uscita della macchina ( assume il valore: 0 se la macchina non è passata quindi non è neanche uscita, 1 se la macchina è uscita )

● I ***bit 6 e 7*** gestiscono intervalli di velocità che vengono rappresentati mediante dei valori : - ‘00’ se velocità < 90 km/h

                                                                                                                                                 - ‘01’ se velocità compresa tra 90 e 100 km/h

                                                                                                                                             - ‘10’ se velocità compresa tra 100 e 110 km/h

                                                                                                                                                 - ‘11’ se velocità > 110 km/h

● Il ***bit 3*** gestisce lo scatto della fotocamera ponendo attenzione ai tempi richiesti dalla consegna.

***Note:*** *per semplicità si ipotizzi che una sola vettura alla volta sia nella zona di misura-fotografia.*

**CALCOLI:**

**Calcolo delle velocità equivalenti in numero di clock delle velocità fornite dal testo:**

V = S/t       S = 1 m → t = S/V       f = 250’000’000 Hz → N°clock = t \* f

● *Velocità* < 90 [25 m/s]

t = 1 / 25 = 0,04 s    → N°clock < 0,04 \* f = 10’000’000

● 90  < *Velocità* < 100 [27,78 m/s]

t = 1 / 27,78 = 0,0359    → 10’000’000 < N°clock < 0,0359 \* f = 8’975’000

● 100 < *Velocità* < 110 [30,55 m/s]

t = 1 / 30,55 = 0,0327    → 8’975’000 < N°clock < 0,0327 \* f = 8’175’000

● *Velocità* > 110

N°clock < 8’175’000

**Calcolo dei valori per lo scatto della fotocamera:**

 ● t = 1 s    → N°clock = t \* f = 250’000’000

 ● t = 0,5 s →  N°clock = t \* f = 125’000’000

**VALORI FORZATI IN SIMULAZIONE ‘Single Step’:**

● Ciclo1: forziamo $t0 con 0x8000, ossia il ***bit 15*** di ***IN\_OUT*** assume il valore 1 e quindi segnala al primo sensore la presenza del passaggio della macchina.

● Ciclo2: forziamo $t0 con 0x4000 ossia il ***bit 14*** di ***IN\_OUT*** assume il valore 1 e quindi segnala al secondo sensore l’uscita della macchina.

● Attesafoto: forziamo $t7 con 0x0000, quindi $t7 viene posto a 0 per simulare l’attesa di un secondo ( durata d’attesa prima di scattare la foto )

● Attesa: forziamo $t8 con 0x0000, quindi quindi $t8 viene posto a 0 per simulare l’attesa di un mezzo secondo ( durata dello scatto della foto )

**PROVE:**

Per testare il corretto funzionamento del programma abbiamo effettuato dei test di controllo:

● Inserendo valori differenti in $s0 (per controllare che effettivamente si entrasse nella corretta fascia di velocità)  
 ● Controllando che le le linee 6 e 7 contenessero i bit corrispondenti alla fascia di velocità ottenuta

● Controllando il corretto funzionamento dello scatto della fotocamera (linea 3 a 1 per 0,5 secondi)

● Verificando la corretta re-inizializzazione dei sensori dell’autovelox dopo aver controllato il passaggio di un’auto, al fine di effettuare un altro controllo all’eventuale passaggio di altre automobili