

## #AUTOVELOX CODICE ESEGUIBILE

# passaggio dei dati alla memoria

```
                .data 0x10000000
IN_OUT:         .half 0x0000
                .text
```

### # AUTOVELOX

```
add $s5, $zero, $zero          #Contatore che viene utilizzato nei confronti
                                #delle varie velocità
li $t7, 125000000              #Per l'attesa di mezzo secondo
li $t8, 25000000               #Per l'attesa di 100ms secondo
```

#-----  
# **PRIMO SENSORE:** è in attesa del passaggio dell'auto, il bit numero 12 va forzato a 1 per indicare il passaggio dell'auto

```
ciclo1: la $t0, IN_OUT          #inserisco nel registro $t0 l'indirizzo
                                #della cella IN_OUT
                                #carico il valore di MASK in $t1
                li $t1, 0x1000
test1:  lh $s0, 0($t0)          #prelevo l'info da IN_OUT e la carico in s0
                and $s1, $s0, $t1 #and bit a bit tra la maschera e i bit di
                                #IN_OUT, carico il risultato in $s1
                bne $s1, $t1, test1 #ritorna alla test se i due valori sono
                                #diversi
```

#-----  
# **SECONDO SENSORE:** riceve il segnale del primo poi avvia il contatore finchè viene forzato ad 1 la linea 13 (uscita della macchina) per l'eventuale controllo e poi confronto della velocità

```
ciclo2:  li $t2, 0x2000        #carico il valore di MASK nel registro $t2
test2:   lh $s0, 0($t0)        #prelevo l'info da IN_OUT e la carico in s0
                and $s4, $s0, $t2 #and bit a bit tra la maschera e i bit di
                                #IN_OUT carico il risultato in $s4

                addi $s5, $s5, 1
                bne $s4, $t2, test2 #se l'auto è uscita va al confronto
```

## #CONTROLLO VELOCITA'

#Carico valori interi nei registri:

```
li $t3, 0x614658      #carico in $t3 il valore per 70km/h
li $t4, 0x55D4A8      #carico in $t4 il valore per 80km/h
li $t5, 0x4C4B40      #carico in $t5 il valore per 90km/h
```

#-----  
#Confronto val ciclo con i registri temporanei:

```
slt $t6, $s5, $t3      #se $s5 ossia il registro contatore della prima
                        #parte è minore di $t3 allora $t6 è 1 sennò 0
beq $t6, $zero, Confronto0 #se $t6 è 1 vado al confronto finale senno proseguo
```

```
slt $t6, $s5, $t4      #se $s5 ossia il registro contatore della prima
                        #parte è minore di $t4 allora $t6 è 1 sennò 0
beq $t6, $zero, Confronto1 #se $t6 è 1 vado al confronto finale sennò proseguo
```

```
slt $t6, $s5, $t5      #se $s5 ossia il registro contatore della prima
                        #parte è minore di $t5 allora $t6 è 1 senn
beq $t6, $zero, Confronto2 #se $t6 è 1 vado al confronto finale sennò proseguo
```

```
j Confronto3           #questo jump si raggiunge solo se la macchina ha
                        #superato i 90km/h
```

#-----  
#Adesso per i confronti devo rendere i bit 9 e 8 di IN\_OUT 00, 01,10,11 pertanto  
uso 4 diversi valori per caricare il tutto in IN\_OUT--> uso la notazione  
esadecimale quindi: | 00 = 0xFCFF) | 01 = 0xFDFF) | 10 = 0xFEFF | 11 = 0xFFFF |

```
Confronto0:la $t0, IN_OUT      #carico l'indirizzo di IN_OUT in $t0
li $s0, 0xFCFF                #carico i bit di $t6 in $s0
sh $s0, 0($t0)                #carico i bit di nuovo in IN_OUT

j ciclo1                      #ritorno all'inizio perchè la macchina andava
                              #a meno di 70km/h
```

```
Confronto1:la $t0, IN_OUT      #carico l'indirizzo di IN_OUT in $t0
li $s0, 0xFDFF                #carico i bit di $t7 in $s0
sh $s0, 0($t0)                #carico i bit di nuovo in IN_OUT

j attesafoto                  #passo ad attesafoto perchè ha sfiorato il
                              #limite di 70km/h
```

```
Confronto2:la $t0, IN_OUT      #carico l'indirizzo di IN_OUT in $t0
li $s0, 0xFEFF                #carico i bit di $t6 in $s0
sh $s0, 0($t0)                #carico i bit di nuovo in IN_OUT

j attesafoto                  #passo ad attesafoto perchè ha sfiorato il
                              #limite di 80km/h
```

```
Confronto3:la $t0, IN_OUT      #carico l'indirizzo di IN_OUT in $t0
li $s0, 0xFFFF                #carico i bit di $t6 in $s0
sh $s0, 0($t0)                #carico i bit di nuovo in IN_OUT

j attesafoto                  #passo ad attesafoto perchè ha sfiorato il
                              #limite di 90km/h
```

**#FOTOCAMERA:** Dopo che la macchina è passata, attendiamo 0,5s e poi scattiamo la foto il cui impulso dura 100ms

```
attesafoto: addi $t7,$t7,-1
beq $t7,$zero,camera1
```

```
#Decremento di 1 per far passare mezzo secondo
#Se il mezzo secondo è passato,salto acamera1
#per scattare, altrimenti aspetto ancora
```

```
j attesafoto
```

```
camera1:  lh $t1, 0($t0)
          andi $t1,$t1,0xFFFF0
          ori $t1,$t1,0x0002
          sh $t1,0($t0)
```

```
#Carico i 16 bit di IN_OUT per poter
#lavorare su quelli necessari
#Modifico valore del bit 1 per
#comandare lo scatto della fotocamera
#Scrivo in Memory($t0+0)il valore di $t1
```

```
j impulso
```

```
# Salto per l'attesa di 100ms (impulso)
```

```
impulso:  addi $t8,$t8,-1
beq $t8,$zero,camera2
```

```
#Decremento di 1 per far passare 100ms
#Se i 100 ms sono passati, salto a camera2
#per terminare lo scatto
```

```
j attesafoto
```

```
#Altrimenti aspetto ancora
```

```
camera2:  lh $t1, 0($t0)
          andi $t1, $t1, 0xFFFF0
          sh $t1, 0($t0)
          sh $zero, 0($t0)
```

```
#Carico i 16 bit di IN_OUT
#Termino lo scatto della fotocamera
#Scrivo in Memory($t0+0)il valore di $t1
#Riporto alla condizione iniziale IN_OUT
#prima di tornare a controllare il primo
#sensore
#Torno a ciclo1: aspetto una nuova macchina
```

```
j ciclo1
```