



UNIVERSITÀ  
DI CAMERINO

Progetto  
**Architetture dei calcolatori**  
A.A. 2022-2023

docente  
Prof. Massimo Trojani

**Sistema di Gestione di un incrocio  
semaforico**

123294  
studente  
**Francesco**

**Capitani Bernardi**

Data ultimo aggiornamento: 14 settembre 2023

## Sommario

|  |    |
|--|----|
| 1. Descrizione del Problema .....                | 3  |
| 1.1 Funzionamento.....                           | 4  |
| 2. Schema del circuito.....                      | 6  |
| 3. FlowChart.....                                | 7  |
| 3.1 Funzione eseguiSemaforo.....                 | 8  |
| 3.2 Funzioni gestione traffico .....             | 9  |
| 3.3 Funzioni eseguiSemaforoSenzaTraffico() ..... | 10 |
| 3.4 Funzioni eseguiSemaforoSenzaTraffico() ..... | 11 |
| 4. Diagramma degli stati.....                    | 12 |
| 5. Funzioni piu' importanti.....                 | 13 |
| 5.1 Interrupt.....                               | 13 |
| 6. Bibliografia e Sitografia.....                | 14 |

# 1. Descrizione del Problema

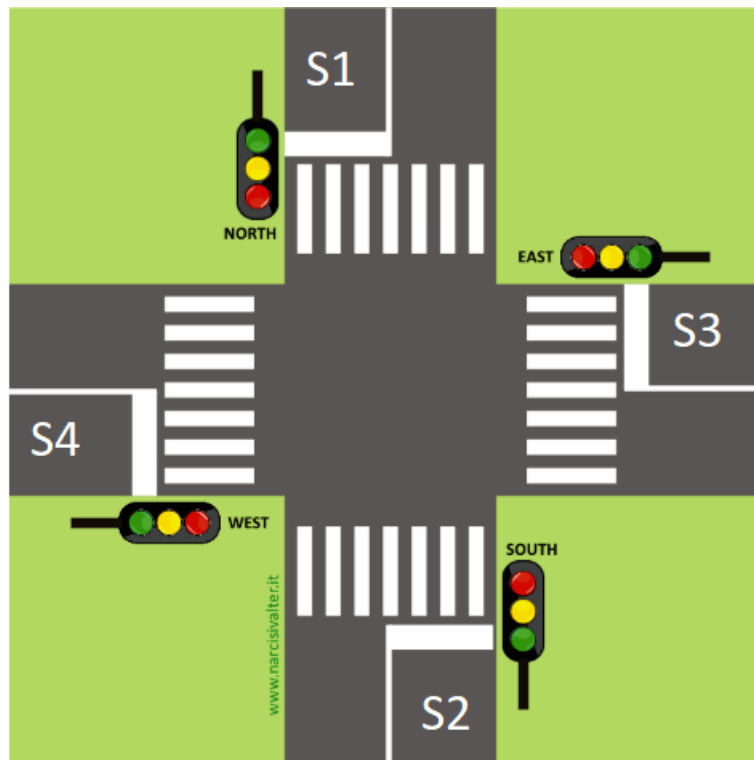


Figura 1

Realizzare un sistema di gestione dei semafori che venga eseguito in background, con le seguenti caratteristiche:

- a) Il sistema deve poter gestire coerentemente il ciclo "V-G-R" o lo stato "L=giallo lampeggiante"
- b) il sistema deve gestire le 4 chiamate pedonali con le seguenti regole:
  - 1) inviare al semaforo il segnale di attivare un ciclo di G-R per il traffico ortogonale ma non per quello parallelo
  - 2) inviare tale entro max 1 minuto solo se sono passati più di cinque minuti dalla fine dell'ultimo ciclo.
- c) Il sistema deve poter gestire correttamente il traffico (es.: Bloccare il Rosso per S2 nel caso il passaggio a livello posto a sud sia chiuso.
- d) avere attivo in foreground un programma di controllo che, tramite un menu' di scelte consente:
  - 1) di poter visualizzare lo stato di ciascun semaforo: ovvero V/G/R/L e secondi rimasti per il prossimo cambio stato
  - 2) di poter modificare i tempi standard previsti per V/G/R
  - 3) Avviare immediatamente un ciclo G-R su un semaforo

## 1.1 Funzionamento

I Semafori funzionano in background tramite un interrupt che viene generato ogni secondo. Il verde è impostato per durare 15 secondi, mentre il giallo 3 secondi e il rosso 22 secondi. Sono stati impostati questi tempi per velocizzare i test

Il flusso di esecuzione standard del semaforo è questo:

- 1- Tutti rossi (stato di assestamento di 2 secondi)
- 2- S1,S2 verdi, S3,S4 rossi (in figura 1)
- 3- S1,S2 gialli, S3,S4 rossi (in figura 1)
- 4- Tutti rossi (stato di assestamento)
- 5- S1,S2 rossi, S3,S4 verdi (in figura 1)
- 6- S1,S2 rossi, S3,S4 verdi (in figura 1)

Una volta arrivati allo stato 6 si riparte dal primo

Anche I pulsanti vengono gestiti in background. I pulsanti posizionati a nord e sud permettono il passaggio di stato da verde a giallo dei semafori S1 e S2 della figura 1. Mentre I pulsanti posizionati a est ed ovest permettono il passaggio di stato da verde a giallo dei semafori S3 e S4 sempre della figura 1.

C'è uno stato chiamato giallo lampeggiante che se viene selezionato nel menu, fa passare tutti I semafori nello stato giallo lampeggiante finchè sempre da menu non si modifica tale stato.

Da menù c'è la possibilità di simulare il traffico a nord, sud, est ed ovest, in modo bloccare solo I semafori desiderati.

## MENU

Menu:

1. Visualizza stato semafori
2. Modifica tempi
3. Avvia immediatamente ciclo G-R su un semaforo
4. attiva giallo lampeggiante
5. disattiva giallo lampeggiante
6. Simula traffico

Seleziona un'opzione:

- Cliccando 1 apparirà questo sotto-menù dove si potrà visualizzare lo stato e il tempo rimanente allo stato successivo:

1. Verifica stato e tempo rimanente del semaforo S1
2. Verifica stato e tempo rimanente del semaforo S2
3. Verifica stato e tempo rimanente del semaforo S3
4. Verifica stato e tempo rimanente del semaforo S4
5. Esci

-Cliccando 2 si potrà modificare I tempi:

-----  
Menu Modifica Tempi:

- 1: Modifica tempo verde
- 2: Modifica tempo giallo
- 3: Modifica tempo rosso
- 4: Esci

-cliccando 3 si può far scattare il ciclo giallo-rosso di un semaforo in modo immediato e apparirà questo menu che permette di scegliere su quale semaforo avviare il ciclo:

Avvia subito ciclo giallo-rosso su uno di questi semafori:

- 1: Semaforo S1(sopra)
- 2: Semaforo S2(sotto)
- 3: Semaforo S3(destro)
- 4: Semaforo S4(sinistro)
- 5: Esci

-cliccando 4 si passa allo stato giallo lampeggiante che viene avviato generalmente di notte

-cliccando 5 si esce dallo stato giallo lampeggiante

-cliccando 6 si può simulare il traffico a nord,sud,ovest o est, in modo da bloccare il semaforo e apparirà questo sotto-menu:

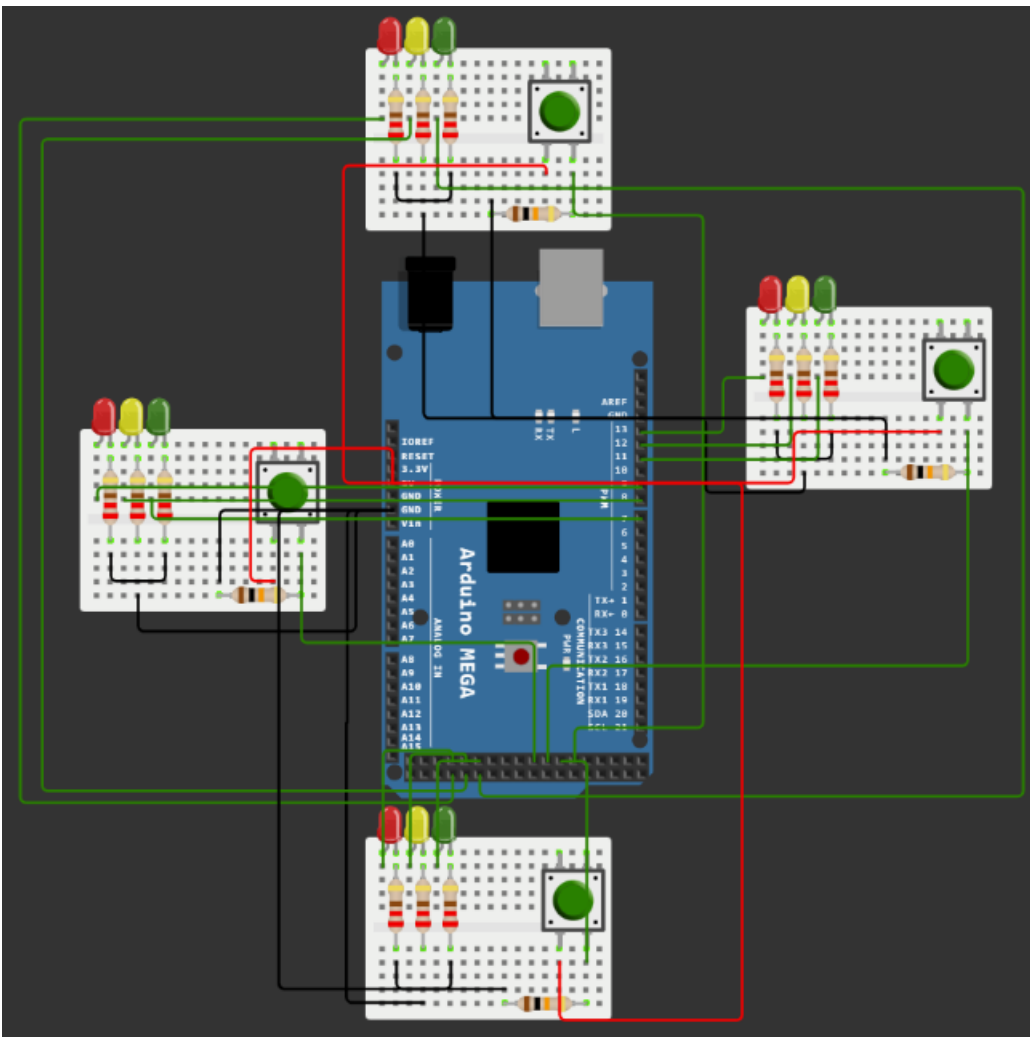
- 1. Blocca traffico
- 2. SBlocca traffico
- 3. Esci

-Se clicchiamo su 1 possiamo scegliere il semaforo da bloccare, mentre su 2 quello da sbloccare:

Quale semaforo vuoi bloccare?:

- 1: Nord
- 2: Sud
- 3: Est
- 4: Ovest
- 5: Esci

## 2. Schema del circuito



Per il circuito è stata utilizzata la scheda arduino Mega che controlla il funzionamento del circuito.

Le Componenti utilizzate sono state tutte collegate a pin di tipo digitale e sono:

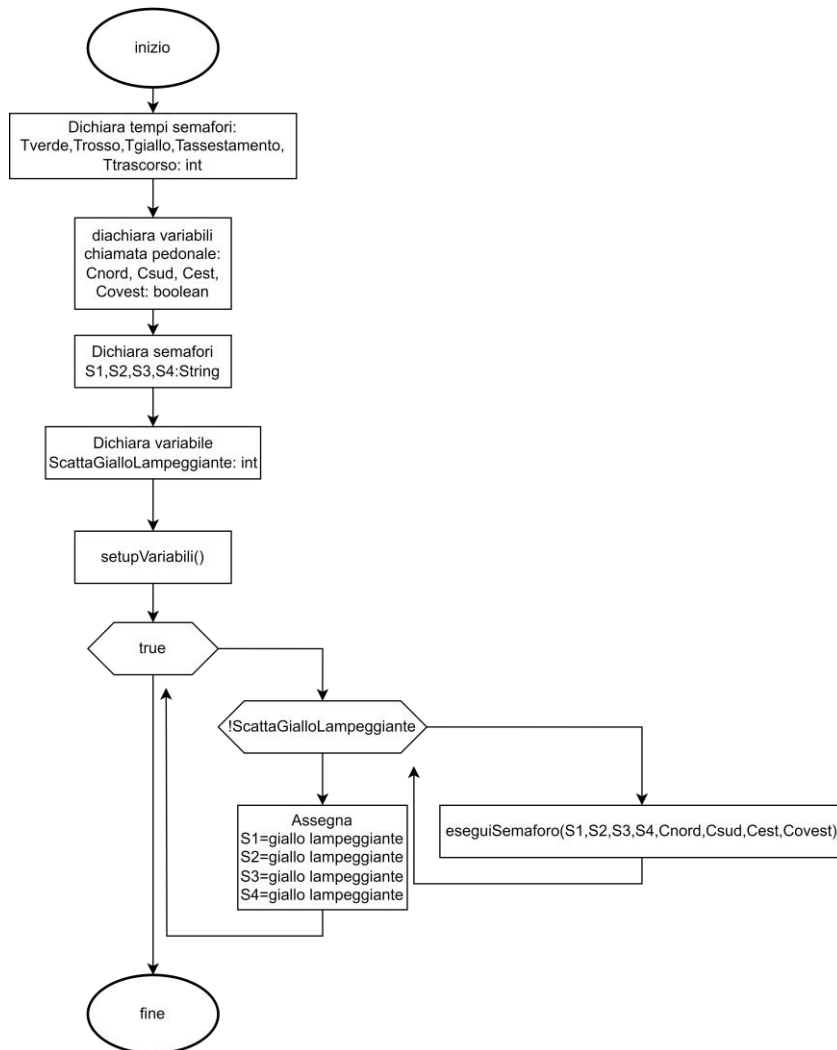
- 1- I LED (led1, led2, ..., led12): Ci sono dodici LED nel circuito, divisi in 4 gruppi di 3 LED ciascuno, ciascun gruppo di un diverso colore (rosso, giallo, verde).
  - I led posizionati a sud sono stati collegati nei pin PL1, PL3, PL5
  - I led posizionati a nord sono stati collegati nei pin PL0, PL2, PL4
  - I led posizionati a est sono stati collegati nei pin PB6, PB7, PB5
  - I led posizionati a ovest sono stati collegati nei pin PH6, PH5, PH4

Resistori (r1, r2, ..., r16): Ci sono sedici resistori nel circuito, ciascuno con un valore di resistenza di 220 ohm per i led o 10.000 ohm per i pulsanti. I resistori sono utilizzati per limitare la corrente che scorre attraverso i LED, evitando che si brucino e per evitare corto circuiti nei pulsanti.

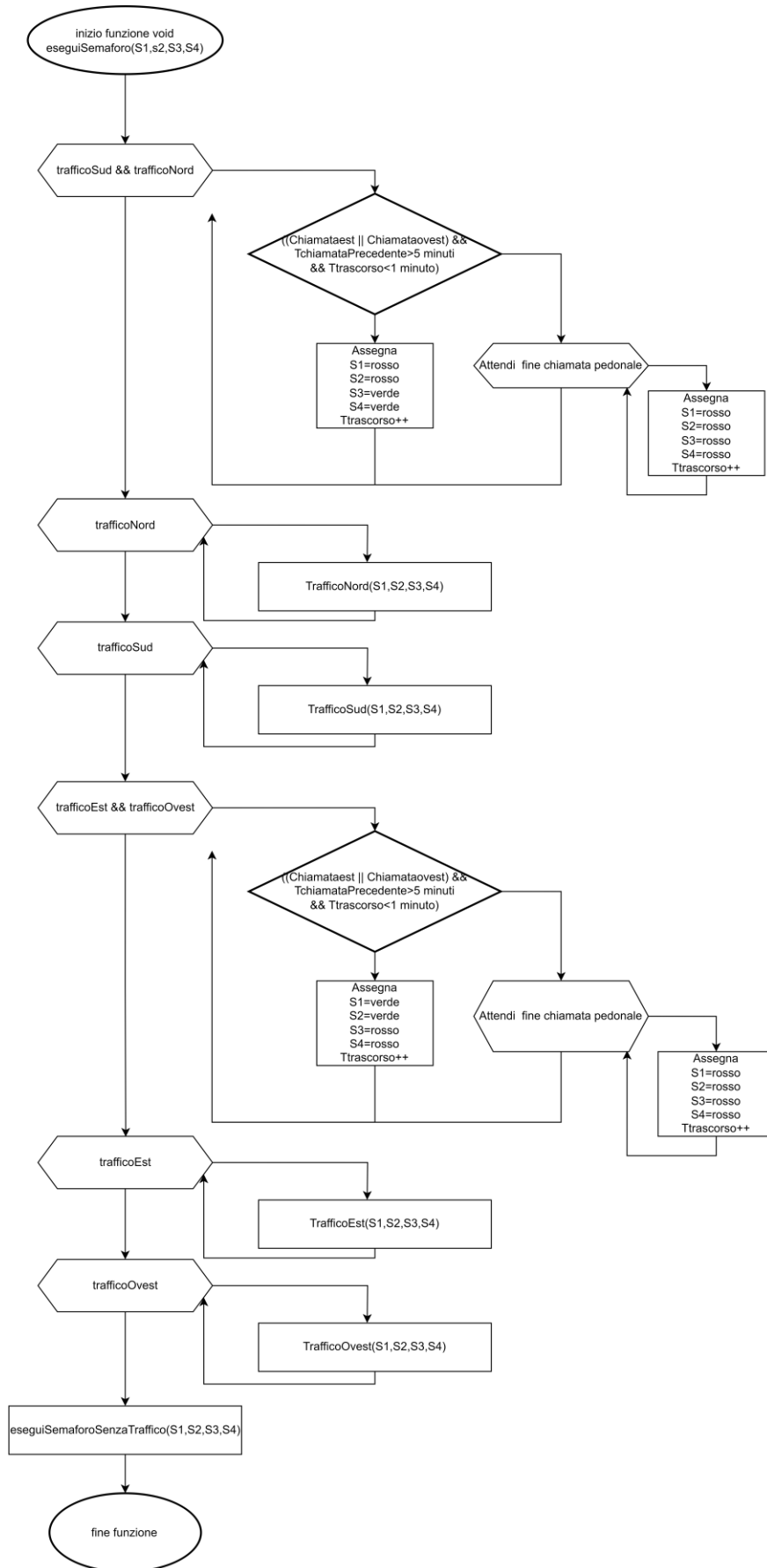
- 2- Pulsanti (btn1, btn2, btn3, btn4): Ci sono quattro pulsanti nel circuito.
  - Il pulsante a nord è stato collegato nel pin PC7
  - Il pulsante a sud è stato collegato nel pin PC5
  - Il pulsante a nord è stato collegato nel pin PC3

- Il pulsante a nord è stato collegato nel pin PC1

### 3. FlowChart

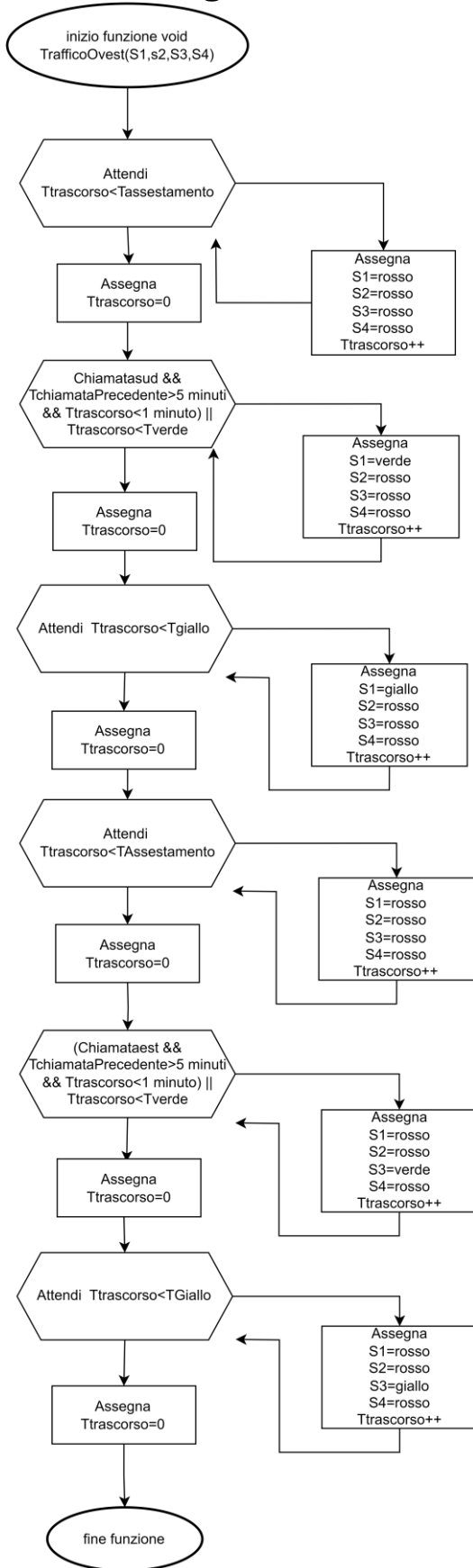


### 3.1 Funzione eseguiSemaforo



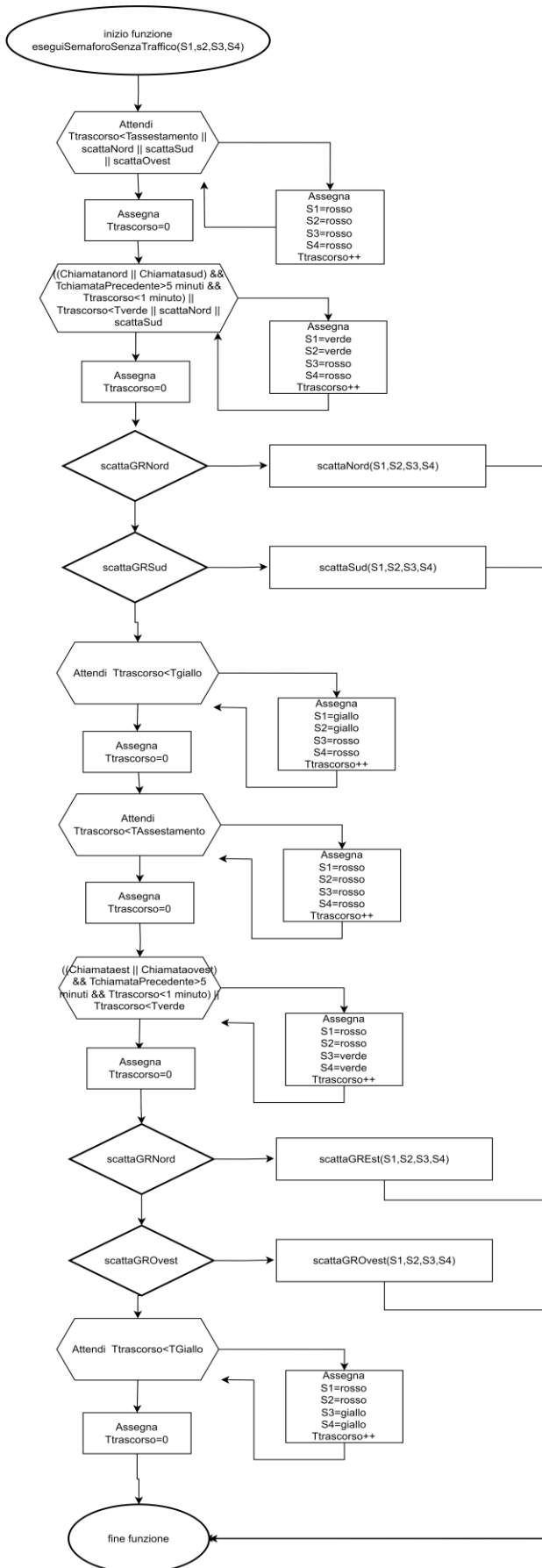


## 3.2 Funzioni gestione traffico

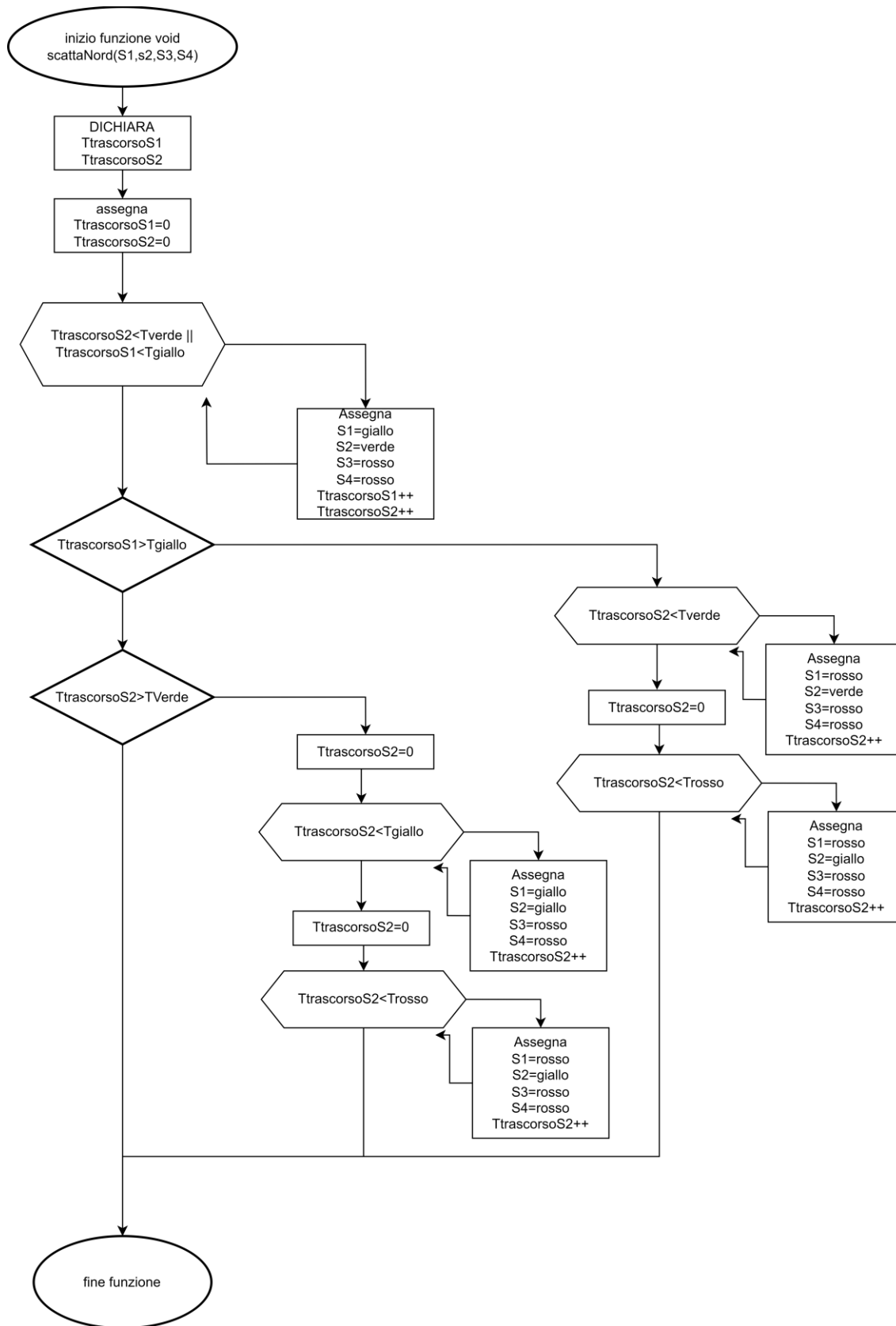


Le funzioni TrafficoNord, TrafficoSud e TrafficoEst sono analoghe

### 3.3 Funzione eseguiSemaforoSenzaTraffico()

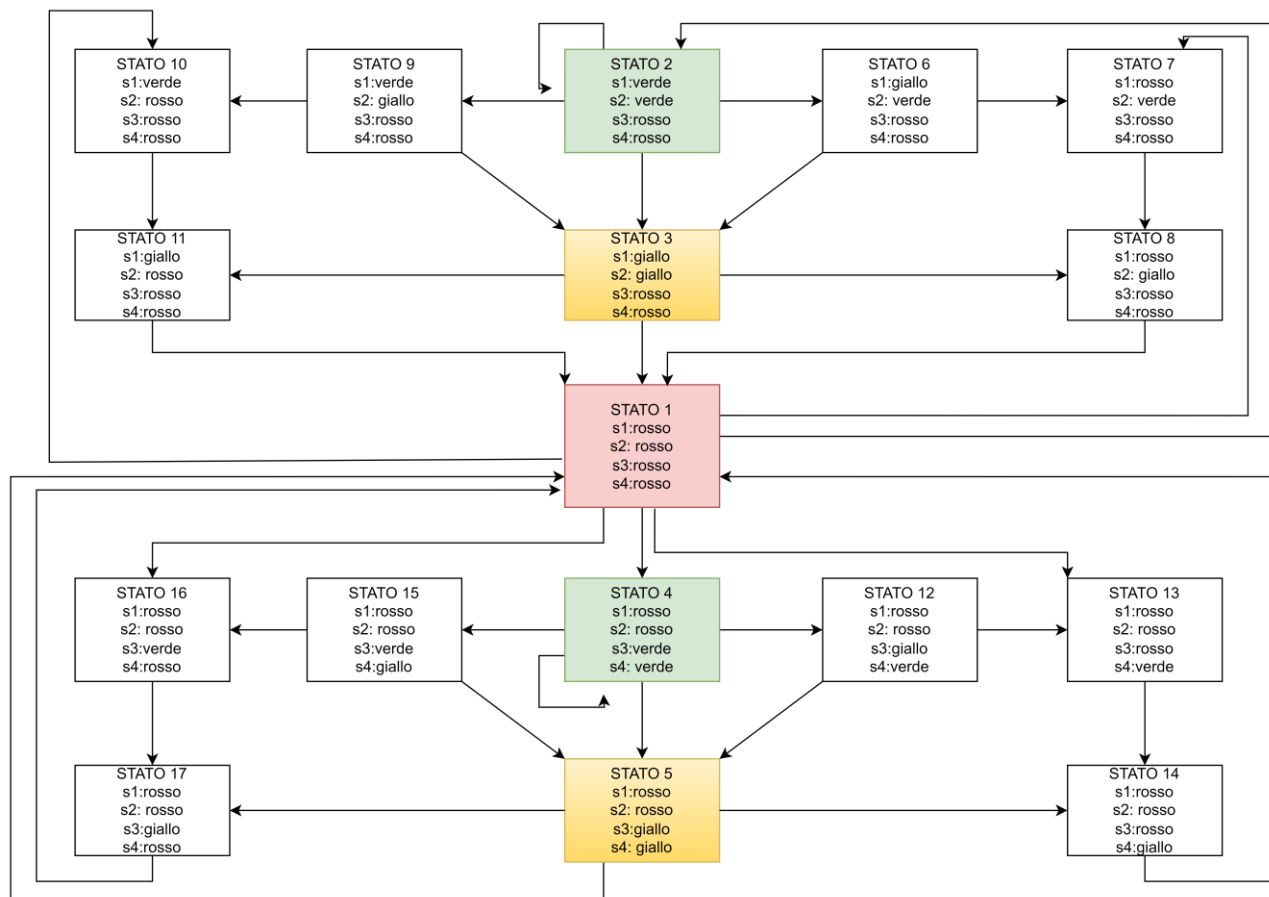


### 3.4 Funzione scattaNord()



Le funzioni scattaSud, scattaEst e scattaOvest sono analoghe

## 4. Diagramma degli stati



All'avvio del programma si parte dallo stato 1 in cui tutti i led sono tutti rossi per un tempo pari a 3 secondi, successivamente si passa allo stato 2 in cui i semafori posizionali e nord-sud diventano verdi e quelli a est-ovest rossi. Dallo stato 2 si passa:

- Allo stato 3 se il tempo del verde si conclude oppure se viene effettuata la chiamata pedonale che blocca il traffico proveniente da nord e sud
- Allo stato 6 se è stato chiamato il ciclo giallo-rosso per il semaforo a nord oppure c'è traffico a nord
- Allo stato 9 se è stato chiamato il ciclo giallo-rosso per il semaforo a sud oppure c'è traffico a sud

Nello stato 2 ci rimango nel caso in cui abbiamo traffico sia a est che a ovest.

Dallo stato 4 si passa:

- Allo stato 5 se il tempo del verde si conclude oppure se viene effettuata la chiamata pedonale che blocca il traffico proveniente da est e ovest
- Allo stato 12 se è stato chiamato il ciclo giallo-rosso per il semaforo a est oppure se c'è traffico a ovest
- Allo stato 9 se è stato chiamato il ciclo giallo-rosso per il semaforo a ovest oppure c'è traffico a est

Nello stato 4 ci rimango nel caso in cui abbiamo traffico sia a est che a ovest.

Gli stati 7,8,10,11,13,14,16,17 sono di transizione

## 5. Funzioni piu' importanti

- Setup(): dove vengono settate tutte le variabili di stato e le componenti come i led e i pulsanti, oltre a inizializzare i timer per gli interrupt
- Foreground(): per la stampa del menu opzioni
- TaskPulsante(): per la gestione della chiamata pedonale
- setSemafori(): per settare i semafori in base al loro stato
- loop(): dove viene eseguita la funzione foreground()

### 5.1 Interrupt

Ci sono 2 Timer per la gestione degli interrupt:

- il TIMER0: gestisce il click sui pulsanti

```
ISR(TIMER0_COMPA_vect)
```

Sotto troviamo il codice utilizzato per inizializzarlo:

```
TCCR0A = 0; // Imposta il registro di controllo A del timer0 a 0
TCCR0B = 0; // Imposta il registro di controllo B del timer0 a 0
TCNT0 = 0; // Azzerare il contatore del timer0
OCR0A = 1; // Imposta un valore di confronto basso per un intervallo molto breve
TCCR0B |= (1 << WGM01); // Modalità CTC (Clear Timer on Compare Match)
TCCR0B |= (1 << CS00); // Imposta il prescaler a 1 (nessuna divisione)
TIMSK0 |= (1 << OCIE0A); // Abilita l'interruzione di confronto A
```

- il TIMER1: gestisce i passaggi di stato dei semafori e viene ed è settato in modo che si generi un interrupt ogni secondo

```
ISR(TIMER1_COMPA_vect)
```

Sotto troviamo il codice utilizzato per inizializzarlo:

```
TIMSK0 |= (1 << OCIE0A); // Abilita l'interruzione di confronto A
TCCR1A = 0; // Azzerare i registri di controllo del timer1
TCCR1B = 0;
TCNT1 = 0; // Azzerare il conteggio del timer1
OCR1A = 15625; //16MHZ/(1024*1)=15625Hz Imposta il valore di confronto per
generare un interrupt ogni secondo (poiché il clock è a 16 MHz e il prescaler è
1024)
TCCR1B |= (1 << WGM12); // Imposta il timer in modalità CTC
TCCR1B |= (1 << CS12) | (1 << CS10); // Abilita il timer con un prescaler di 1024
TIMSK1 |= (1 << OCIE1A); // Abilita l'interrupt sul confronto con OCR1A
```

## **6. Bibliografia e Sitografia**

- [1] [https://ww1.microchip.com/downloads/en/DeviceDoc/Atmel-7810-Automotive-Microcontrollers-ATmega328P\\_Datasheet.pdf](https://ww1.microchip.com/downloads/en/DeviceDoc/Atmel-7810-Automotive-Microcontrollers-ATmega328P_Datasheet.pdf).