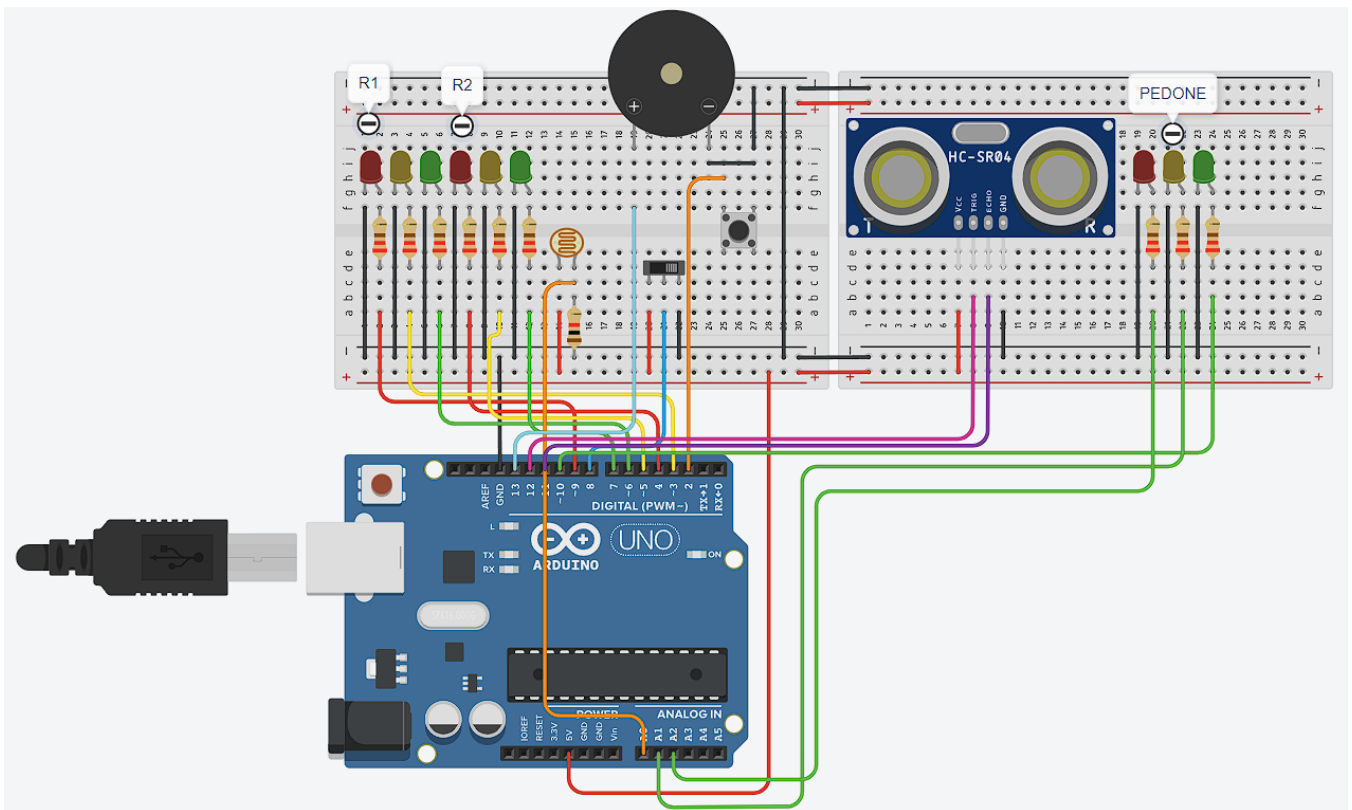


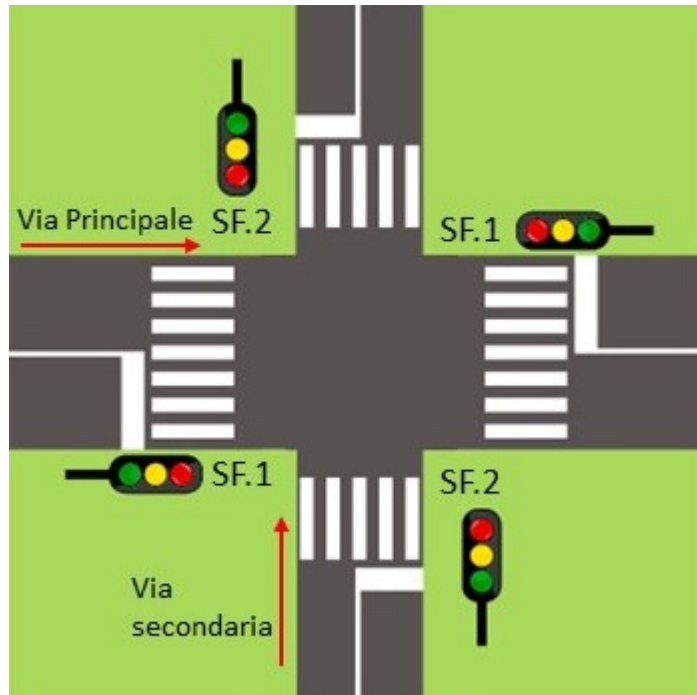
ESERCITAZIONE SU SEMAFORO INTELLIGENTE

SCHEMA ELETTRICO

(1)



(2)



### FORMULE ADOPERATE

$$S = V * t$$

(Spazio (cm) = 0,03431 x t (microsecondi), considerando che la velocità del suono a temperatura ambiente di 20°C è di 343,1 m/s cioè 0,03431 cm/microsecondi)

### TABELLA DEI VALORI

(3)

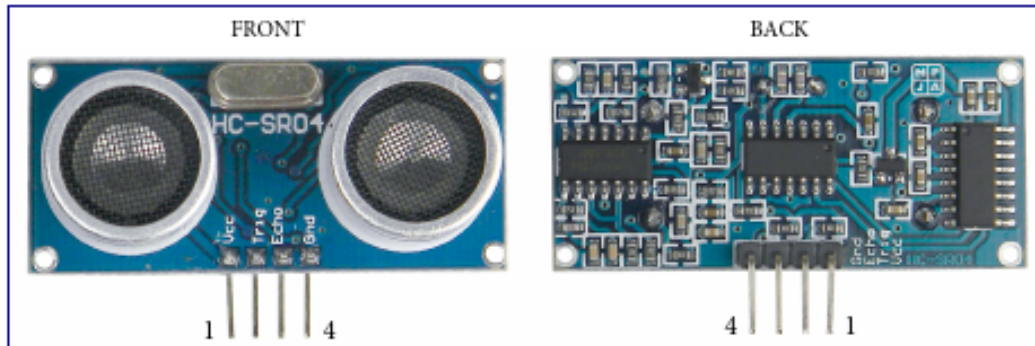
VARIABILE	PIN	DESCRIZIONE
R1	9	LED Rosso Via principale
G1	3	LED Giallo Via principale
V1	6	LED Verde Via principale
R2	4	LED Rosso Via secondaria
G2	5	LED Giallo Via secondaria
V2	7	LED Verde Via secondaria
PR	16	LED Rosso Pedone
PG	15	LED Giallo Pedone
PV	10	LED Verde Pedone
Fres	A0	Fotoresistenza
INT	8	Interruttore
PD	2	Pulsante Pedone
BUZ	13	Buzzer
TRIG	12	Trig Sensore ultrasuoni
ECHO	11	Echo Sensore ultrasuoni

## DATASHEET

### SENSORE ULTRASUONI HC-SR04:

(4)

#### 3. Product Views



#### 4. Module Pin Assignments

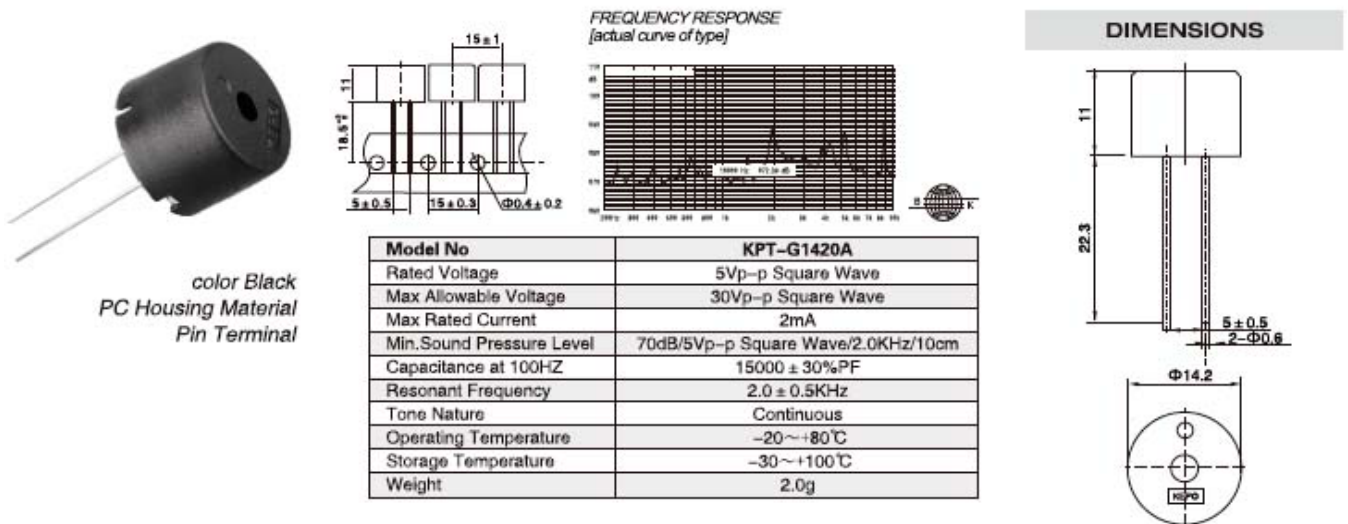
	Pin Symbol	Pin Function Description
1	VCC	5V power supply
2	Trig	Trigger Input pin
3	Echo	Receiver Output pin
4	GND	Power ground

#### 5. Electrical Specifications

Electrical Parameters	HC-SR04 Ultrasonic Module
Operating Voltage	5VDC
Operating Current	15mA
Operating Frequency	40KHz
Max. Range	4m
Nearest Range	2cm
Measuring Angle	15 Degrees
Input Trigger Signal	10us min. TTL pulse
Output Echo Signal	TTL level signal, proportional to distance
Board Dimensions	1-13/16" X 13/16" X 5/8"
Board Connections	4 X 0.1" Pitch Right Angle Header Pins

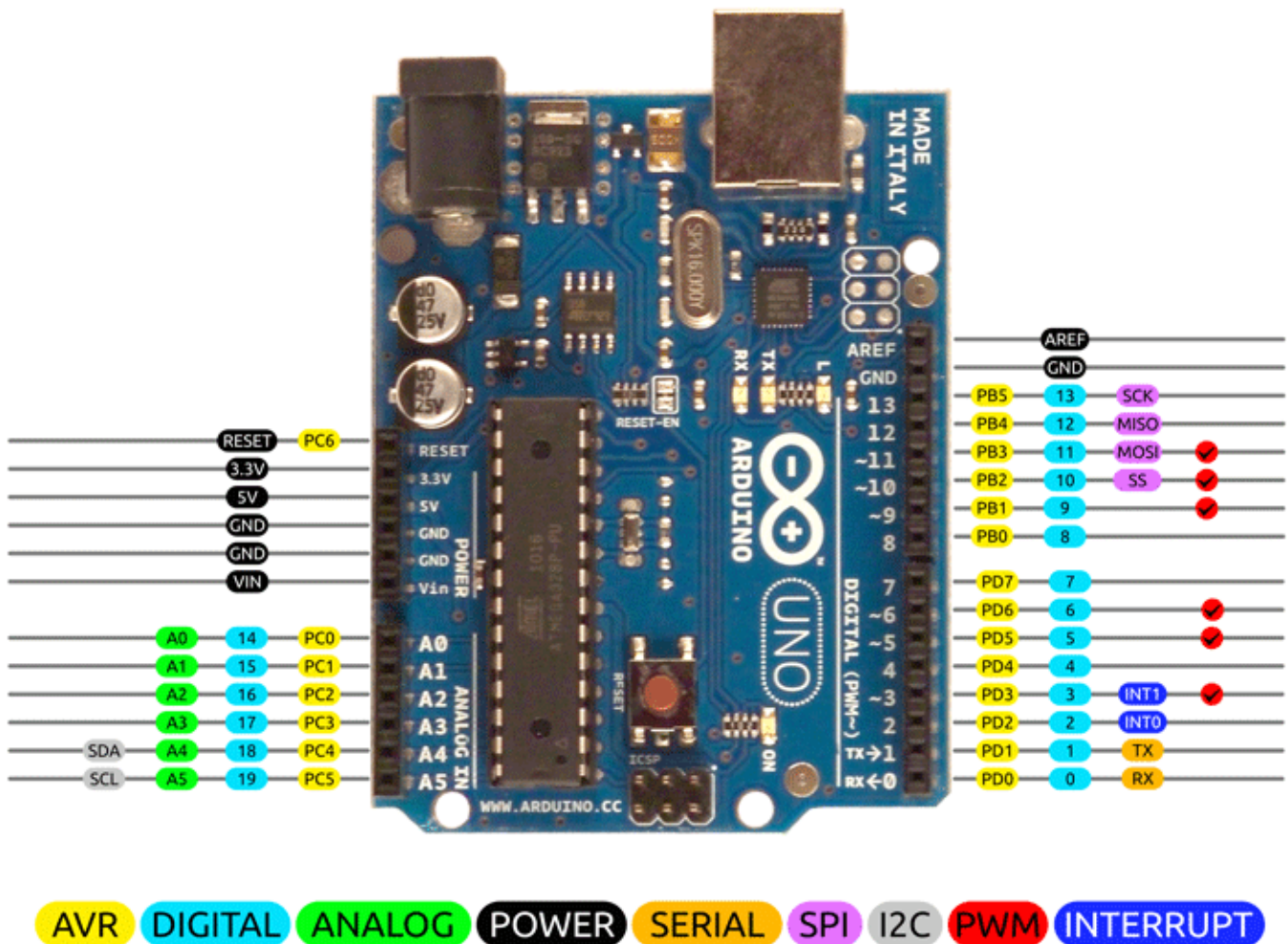
## BUZZER:

(5)



## PIN ARDUINO UNO R3

(6)



## ELENCO COMPONENTI UTILIZZATI

N.9 Resistori 220Ω

N.3 Diodi led rossi

N.3 Diodi led gialli

N.3 Diodi led verdi

N.1 Fotoresistenza

N.1 Interruttore

N.1 Pulsante

N.1 Sensore ultrasuoni HC-SR04

N.1 Resistore 1kΩ

N.1 Buzzer

N.1 Arduino UNO R3

## PROGRAMMA

```
const int PD = 2;
const int R2 = 4;
const int G1 = 3;
const int G2 = 5;
const int V1 = 6;
const int V2 = 7;
const int INT = 8;
const int R1 = 9;
const int PV = 10;
const int Fres = A0;
const int trig = 12;
const int echo = 11;
const int buz = 13;
const int PR = 16;
const int PG = 15;
int PREN = 0, a = 0, b = 0, c = 0, e = 0, d = 0, f = 0;
int distanza = 101;
void setup()
{
```

```
    pinMode(R1,OUTPUT);
    pinMode(R2,OUTPUT);
    pinMode(G1,OUTPUT);
    pinMode(G2,OUTPUT);
    pinMode(V1,OUTPUT);
    pinMode(V2,OUTPUT);
    pinMode(INT,INPUT);
    pinMode(Fres,INPUT);
    pinMode(PD, INPUT_PULLUP);
    pinMode(PV,OUTPUT);
    pinMode(trig, OUTPUT);
    pinMode(echo, INPUT);
    pinMode(buz, OUTPUT);
    pinMode(PR, OUTPUT);
    pinMode(PG, OUTPUT);
```

Fase dichiarativa degli ingressi e delle uscite

Utilizzo della resistenza di Pullup integrata di Arduino

```
    attachInterrupt(digitalPinToInterrupt(PD), PRENOTAZIONE, RISING);
```

Dichiarazione interrupt

```
    Serial.begin(9600);
    PREN = 0;
```

Apri porta seriale con velocità 9600 baud

```
}
```



```

void loop(){

  int ST = digitalRead(INT);
  int res = analogRead(Fres);

  if(INT == 0){
    f=0;
    digitalWrite(PR,HIGH);
    if(PREN == 0){
      a=0;
      if(res >= 300){
        if(b==0){
          Serial.println("Giorno");
          b=1;
          c=0;
        }
        NOTTE:
        digitalWrite(G1,LOW);
        digitalWrite(R2,LOW);
        digitalWrite(R1,HIGH);
        digitalWrite(V2,HIGH);

        delay(1000);
        digitalWrite(G2,HIGH);
        digitalWrite(V2,LOW);

        delay(1000);
        digitalWrite(G2,LOW);

        digitalWrite(V1,HIGH);
        digitalWrite(R2,HIGH);
        digitalWrite(R1,LOW);

        delay(1000);
        digitalWrite(G1,HIGH);
        digitalWrite(V1,LOW);
        digitalWrite(R2,HIGH);
        digitalWrite(R1,LOW);

        delay(1000);
        return;
      }else{
        b = 0;
        if(c==0){
          Serial.println("Notte");
          c=1;
        }
      }
    }
  }
}

```

In base al valore letto dalla fotoresistenza entra nella sezione Notte o nella sezione Giorno, in questo caso Giorno

Fase di normale lavoro del semaforo

Se il valore della fotoresistenza non è maggiore o uguale al valore che è stato impostato, si entra nella sezione Notte

```

digitalWrite(trig,LOW);
delayMicroseconds(5);
digitalWrite(trig,HIGH);
delayMicroseconds(10);
digitalWrite(trig,LOW);
delayMicroseconds(5);
unsigned long tempo = pulseIn(echo,HIGH);
distanza = tempo/58;
if(distanza <= 100){
    if(e==0)
        Serial.println("Macchina rilevata");
    d=0;
    e=1;
    digitalWrite(V1,LOW);
    digitalWrite(G1,HIGH);
    delay(1000);
    goto NOTTE;
}
else{
    if(d==0){
        Serial.println("Macchina non rilevata");
        d=1;
        e=0;
    }
    digitalWrite(R1,LOW);
    digitalWrite(G1,LOW);
    digitalWrite(V1,HIGH);
    digitalWrite(G2,LOW);
    digitalWrite(V2,LOW);
    digitalWrite(R2,HIGH);
    delay(1000);
}

}

}else{
    digitalWrite(G1,LOW);
    digitalWrite(V1,LOW);
    digitalWrite(G2,LOW);
    digitalWrite(V2,LOW);
    digitalWrite(PR,LOW);

```

Genero un onda quadra per attivare il sensore ultrasuoni HC-SR04

Registra il tempo dall'emissione alla ricezione del segnale in microsecondi

Calcolo la distanza con la formula sopra citata

Se la distanza rilevata dal sensore è minore o uguale di un metro, vuol dire che è stata rilevata una macchina e quindi viene eseguita la fase di normale lavoro del semaforo

Se non viene rilevata nessuna macchina, il semaforo è **verde** sulla strada principale e **rosso** sulla strada secondaria

Prenotazione pedonale avvenuta

```

if(distanza >=100){
  digitalWrite(G1, HIGH);
  delay(1000);
  digitalWrite(G1, LOW);
}
digitalWrite(R2,HIGH);
digitalWrite(R1,HIGH);
digitalWrite(PV,HIGH);
Serial.println("5 secondi al termine del passaggio pedone");
for(int i=0; i<=1; i++){
  tone(buz,500,1000);
  delay(900);
  noTone(buz);
  delay(100);
}
Serial.println("4 secondi al termine del passaggio pedone");
for(int i=0; i<=1; i++){
  tone(buz,500,1000);
  delay(900);
  noTone(buz);
  delay(100);
}
Serial.println("3 secondi al termine del passaggio pedone");
for(int i=0; i<=1; i++){
  tone(buz,500,1000);
  delay(900);
  noTone(buz);
  delay(100);
}
Serial.println("2 secondi al termine del passaggio pedone");
for(int i=0; i<=1; i++){
  tone(buz,500,1000);
  delay(900);
  noTone(buz);
  delay(100);
}
digitalWrite(PV,LOW);
digitalWrite(PG,HIGH);

```

Se la distanza è maggiore di un metro, completo la sequenza semaforica accendendo il Led **giallo**

Il semaforo del passaggio pedonale diventa **verde** ed è accompagnato dalla classica suoneria che cambia in base al tempo rimasto



```

Serial.println("1 secondo al termine del passaggio pedone");
for(int i=0; i<=4; i++){
tone(buz,500,500);
delay(400);
  noTone(buz);
  delay(100);
}
Serial.println("Fine passaggio pedone");
digitalWrite(PG,LOW);
digitalWrite(PR,HIGH);
PREN = 0;

```

Il semaforo  
diventa  
giallo e la  
suoneria più  
veloce

```

}
} else{
digitalWrite(PR,LOW);
digitalWrite(V2,LOW);
digitalWrite(R2,LOW);
digitalWrite(G1,HIGH);
digitalWrite(G2,HIGH);
digitalWrite(PG,HIGH);
delay(500);
digitalWrite(PG,LOW);
digitalWrite(G1,LOW);
digitalWrite(G2,LOW);
delay(500);
if(f==0)
Serial.println("Spento");
f=1;
b=0;
}

```

Semaforo spento,  
lampeggio giallo semafori

```

}
void PRENOTAZIONE(){
  if (a==0){
    Serial.println("Prenotato");
    a=1;
  }
  PREN = 1;
}

```

Funzione prenotazione, interrupt

## RELAZIONE

Il seguente progetto ha lo scopo di realizzare un semaforo intelligente con: una fotoresistenza che è utilizzata per cambiare modalità tra Giorno e Notte, una chiamata pedonale attraverso la pressione di un pulsante, un sensore ad ultrasuoni che rileva le vetture, un buzzer e la possibilità di accendere o spegnere il semaforo.

Per progettare questo semaforo si è dapprima partiti con il solo semaforo semplice che comprende la sola accensione dei Led: Rosso, Giallo e Verde.

Successivamente è stato aggiunto un pulsante per la prenotazione del passaggio pedonale con un interrupt che permette al pedone di poter premere in qualsiasi momento e per la durata che vuole il pulsante, così da non dover tener premuto fino al passaggio pedonale verde come invece sarebbe successo non utilizzandolo.

In seguito è stato previsto anche un interruttore che permette di spegnere il semaforo e di far lampeggiare il giallo.

Dopo di che è stata aggiunta una fotoresistenza che permette al semaforo di gestire 2 sequenze: la sequenza Giorno e la sequenza Notte. La sequenza Giorno include il normale lavoro del semaforo. La sequenza Notte invece, se non viene rilevata nessuna vettura sulla strada secondaria dal sensore ad ultrasuoni, illumina il semaforo sulla strada principale di verde e il semaforo sulla strada secondaria di rosso. Invece, se il sensore ad ultrasuoni rileva una vettura sulla strada secondaria, esegue la sequenza normale del Giorno.

Infine è stato aggiunto un buzzer che segnala ai pedoni sul passaggio pedonale il tempo che manca per il rosso. Viene riprodotto inizialmente un suono intermittente più lento dalla durata di 1s per 4 secondi, successivamente, dopo l'accensione del giallo viene riprodotto un suono intermittente più veloce dalla durata di 500ms per 1 secondo, dopo di che scatta il rosso.