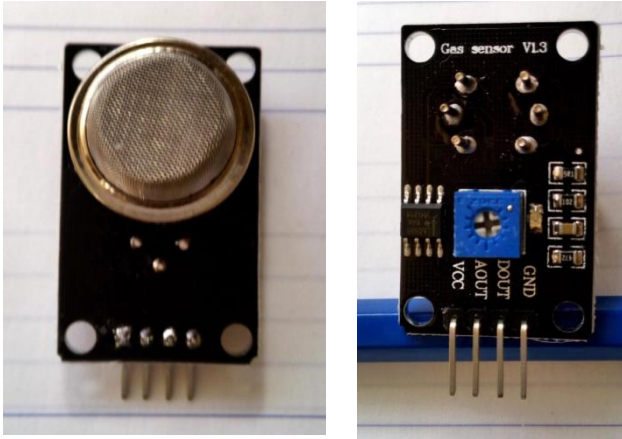


34 - sensore di gas



Il sensore di gas rappresentato in figura rileva la presenza di gas combustibili e fumo e genera un segnale digitale quando la concentrazione di detti gas supera una soglia prestabilita. Il sensore genera anche un segnale analogico il cui valore e' proporzionale alla concentrazione dei gas.

Si tratta di un sensore a semiconduttore (SnO_2), la cui conducibilita' viene modificata dall'assorbimento dei gas a contatto con la sua superficie, riscaldata elettricamente a una prestabilita temperatura.

Alcuni sensori sono in grado, a seconda della temperatura della superficie porosa del semiconduttore, di individuare il tipo di gas presente nell'aria.

Il sensore utilizzato in questo esercizio non e' in grado di distinguere tra le varie tipologie di gas e presenta sia un'uscita analogica che un'uscita digitale.

E' montato su di una basetta sulla quale trovano posto anche un led, un potenziometro, alcune resistenze ed un comparatore. Il potenziometro consente di "tarare" il sensore in modo che l'uscita digitale venga attivata solo quando il comparatore rileva che la concentrazione dei gas (e quindi il segnale generato dal sensore) supera la soglia predefinita dal potenziometro.

In questo esercizio Arduino lancia un segnale di allarme facendo lampeggiare un led ed attivando un cicalino quando si riceve un segnale digitale. Il programma evidenzia anche costantemente, sul monitor seriale, i valori forniti dall'uscita analogica e consente quindi di osservare come vengono percepiti differenti gas o comunque differenti agenti inquinanti.

Attenzione: il segnale digitale e' HIGH quando non viene rilevata la presenza di idrocarburi e viceversa e' LOW quando la loro presenza supera la soglia regolata dal potenziometro.

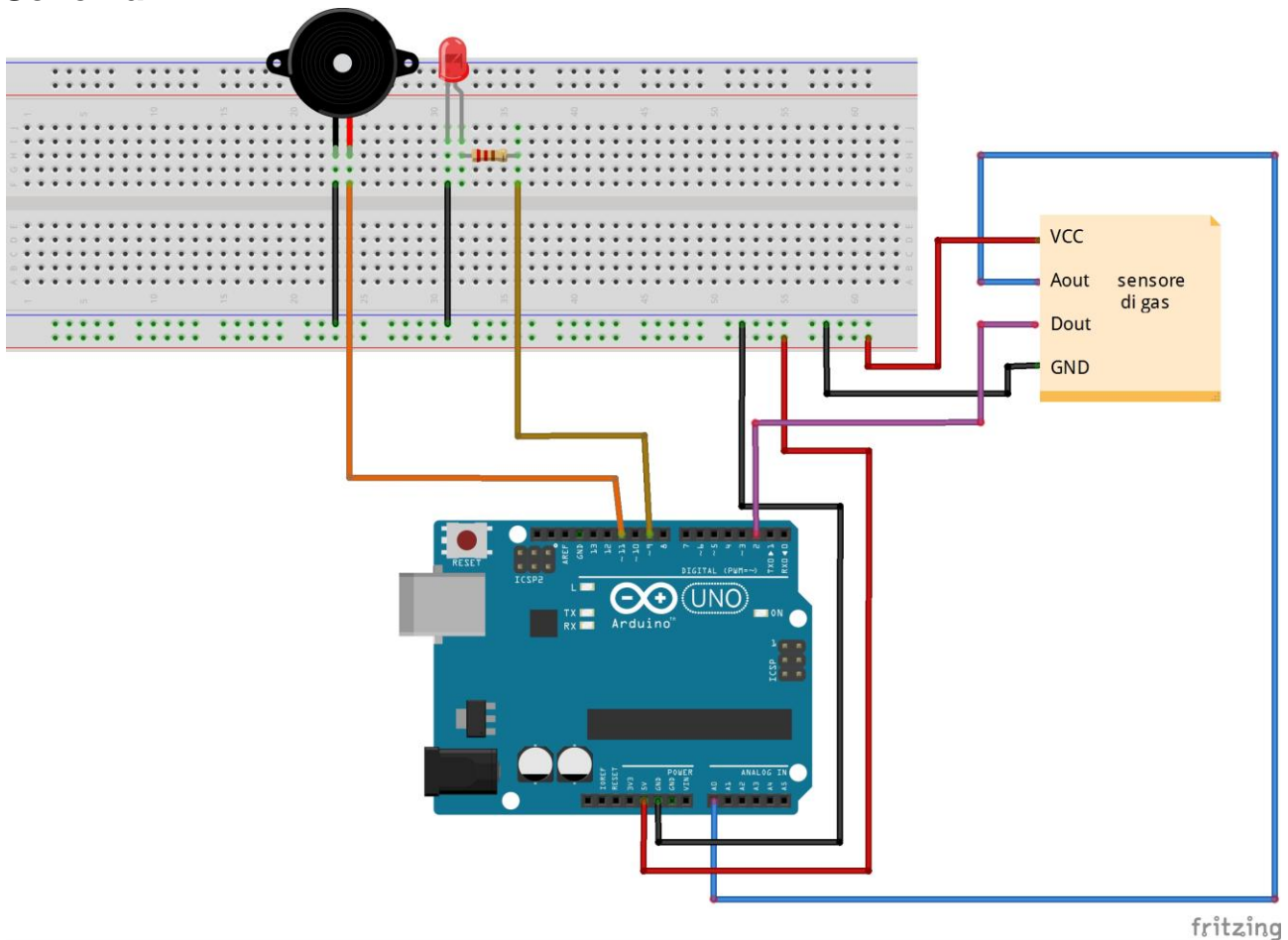
Nota: Questo esercizio e questa nota sono parte di una serie che vede protagonisti Arduino ed alcuni dei componenti ad esso collegabili. Per la maggior parte degli esercizi e' anche disponibile un filmato su youtube.

- [Esercizi facenti parte della raccolta](#)
- [Filmati presenti su youtube](#)
- [Informazioni su arduino e sui componenti collegabili](#)
- [Breve manuale di programmazione](#)

Materiali

- un sensore di gas montato su basetta e con uscita analogica e digitale
- un led
- un buzzer
- una resistenza da 220 ohm
- un po' di cavetteria

Schema



Programma

/*

Questo programma utilizza un sensore di gas a semiconduttore montato su di una basetta con uscite analogica e digitale.

Le uscite sono collegate, rispettivamente, alla porta analogica 0 ed alla porta digitale 2. Il circuito e' completato

da un buzzer collegato alla porta 11 e da un led, con relativa resistenza di contenimento, collegato alla porta 9.

La presenza di idrocarburi nell'aria aumenta la conduttanza del semiconduttore e, se la concentrazione supera un limite

regolato da un potenziometro montato sulla basetta, il sistema lancia un allarme visivo e sonoro

*/

```
const int sensorpin = 2; // l'uscita digitale del sensore e' collegato alla porta 3
```

```
const int ledpin = 9; // il led e' collegato alla porta 9
```

```
const int buzzerpin =11; // il buzzer e' collegato alla porta 11;
```

```
int statosensore = 0; // variabile di memorizzazione dello stato del sensore
```

```
int valoresensore = 0; // variabile di memorizzazione del valore analogico rilasciato dal sensore
```

```
void setup()
```

```
{
```

```
  pinMode(sensorpin,INPUT);
```

Arduino: sensore di gas

```
pinMode(ledpin,OUTPUT);
pinMode(buzzerpin,OUTPUT);
Serial.begin(9600);
}

void loop()
{
  statosensore = digitalRead(sensorpin); // acquisisce il segnale digitale del sensore
  valoresensore = analogRead (A0); // acquisisce il segnale analogico del sensore
  Serial.print (statosensore); // alimenta il monitor seriale
  Serial.print ( " ");
  Serial.println (valoresensore);
  if(statosensore == LOW) // se e' stata rilevata una concentrazione di gas
  {

    digitalWrite(ledpin,HIGH); // attiva l'allarme
    digitalWrite(buzzerpin,HIGH);
    delay (300); // attende 3/10 di secondo e spegne l'allarme
  }
  digitalWrite(ledpin,LOW);
  digitalWrite(buzzerpin,LOW);
  delay (300); // attende 3/10 di secondo e fa ripartire il loop
}
```