

## Sensore di parcheggio

- **Obiettivo:**

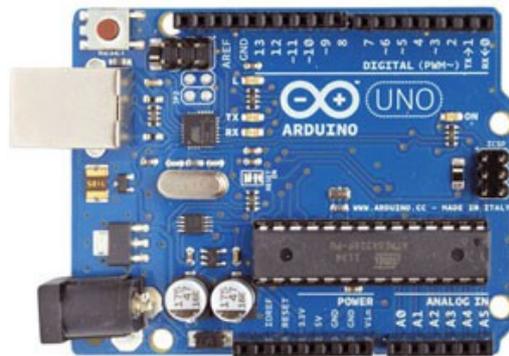
Assemblare un circuito e programmare un microcontrollore per realizzare le seguenti specifiche.

- **Specifiche:**

1. Tramite sensore ad ultrasuoni rilevare la distanza di un oggetto da esso.
2. In base alla distanza dell'oggetto dal sensore un cicalino emette un suono a intermittenza: maggiore è la distanza più l'intermittenza è lenta, minore è la distanza più rapida è l'intermittenza.
3. Anche un LED lampeggia con intermittenza uguale.

- **Hardware:**

Microcontrollore utilizzato: Arduino UNO Rev3



Arduino UNO Rev3

In questa esperienza si vuole realizzare un sensore di parcheggio ovvero un sensore di prossimità con Arduino UNO.

Arduino UNO è un microcontrollore programmabile in C++

Per far questo si necessita di un sensore ad ultrasuoni per il rilevamento della distanza di un oggetto dal sensore.

Il sensore ad ultrasuoni è composto da un emettitore (che emette suoni di frequenza maggiore ai 20 kHz) e da un ricevitore (che capta gli ultrasuoni).



Sensore ad ultrasuoni

La distanza rilevata viene “rappresentata” da un’intermittenza del suono di un cicalino e del lampeggio di un LED. Per creare l’intermittenza bisogna fornire ai componenti una tensione TTL (0V – 5V, segnale periodico ad onda quadra). Maggiore è la distanza maggiore sarà il periodo del segnale TTL, diminuendo la frequenza dell’intermittenza; minore è la distanza, minore sarà il periodo del segnale TTL, aumentando la frequenza dell’intermittenza.



Cicalino



LED

- **Software:**

Il programma che si carica su Arduino è scritto in linguaggio C++.

```
int triggerPort = 7;           // pin 7 di Arduino è collegato all'emettitore del sensore ad ultrasuoni
int echoPort = 8;             //pin 8 di Arduino è collegato al ricevitore del sensore ad ultrasuoni
int speaker = 13;            //pin 13 è collegato al cicalino
int led = 12;                //pin 12 è collegato al LED
void setup() {
    pinMode (triggerPort, OUTPUT);    //pin 7 è un OUTPUT
    pinMode (echoPort, INPUT);       //pin 8 è un INPUT
    pinMode (speaker, OUTPUT);
    pinMode (led,OUTPUT);
    Serial.begin (9600);
    Serial.println ("Sensore ultrasuoni:"); //visualizza sul monitor seriale "Sensore ultrasuoni"
}

void loop()
{
    digitalWrite (triggerPort, LOW);
    digitalWrite (triggerPort, HIGH);
    delayMicroseconds (10);
    digitalWrite (triggerPort, LOW);

    long duration = pulseIn (echoPort,HIGH);
    long r = 0.034 * duration / 2;
    Serial.print( "durata: " );
    Serial.print( duration );
    Serial.print( " , " );
    Serial.print( "distanza: " );
    if( duration > 38000 )
        Serial.println( "fuori portata");
    else
    {
        Serial.print( r );
        Serial.println( "cm" );
    }
    if( r > 3 && r <= 200)
    {
```

```

        delay(r*10);
        digitalWrite(speaker, HIGH);
        digitalWrite(led,HIGH);
        delay(r*10);
    }
    if( r <= 3)
    {
        digitalWrite (led,HIGH);
        digitalWrite(speaker, HIGH);
        delay(1000);
    }
    digitalWrite(speaker, LOW);
    digitalWrite(led,LOW);
    delay(10);
}

```

La distanza rilevata viene “letta” da Arduino che, se rileva un valore minore di 11764,70 (200 cm) e maggiore di 176,47 (3 cm), esegue le seguenti istruzioni:

```

{
    delay(r*10);

    digitalWrite(speaker, HIGH);

    digitalWrite(led,HIGH);

    delay(r*10);
}

```

Queste istruzioni permettono ad Arduino di far passare la corrente al LED e al cicalino per un tempo pari a  $r*10$  millisecondi dove  $r$  è la distanza in centimetri calcolata con il seguente calcolo

$r = 0.034 * duration / 2;$  // duration è la variabile a cui è associato il valore rilevato dal sensore

Se il valore rilevato è minore di 3 cm

```

{
    digitalWrite (led,HIGH);
    digitalWrite(speaker, HIGH);
    delay(1000);
}

```

al LED e al cicalino viene fornita una corrente continua, finché l’oggetto non supera i 3 cm dal sensore, quindi i due componenti rimangono attivi senza intermittenze