

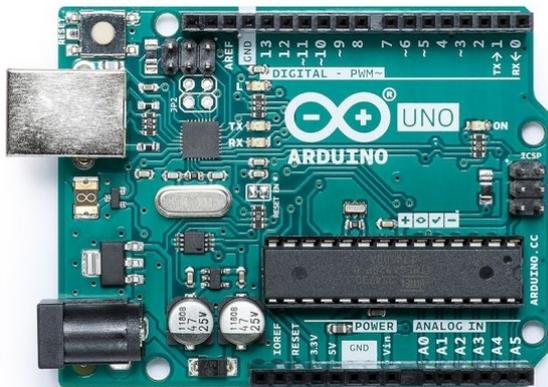
ITIS-LS “Francesco Giordani” Caserta

prof. Ennio Ranucci

a.s. 2019-2020

Realizziamo semplici circuiti con Arduino

Esercitazioni svolte in ambiente Arduino



ITIS-LS "Francesco Giordani" Caserta
Anno scolastico: 2019/2020
Classe 3^A sez.B spec. Informatica e telecomunicazioni

Data:

Numero progressivo dell'esercizio: es0

Versione: 1.0

Programmatore/i:

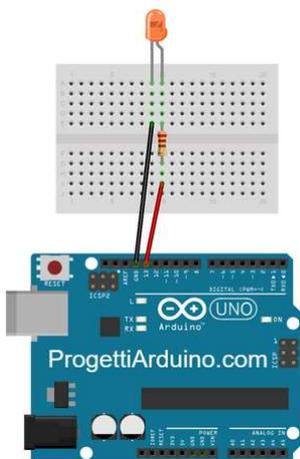
Sistema Operativo: Windows 10

Compilatore/Interprete: Arduino 1.8.10

Obiettivo didattico: Conoscere l'ambiente di programmazione;

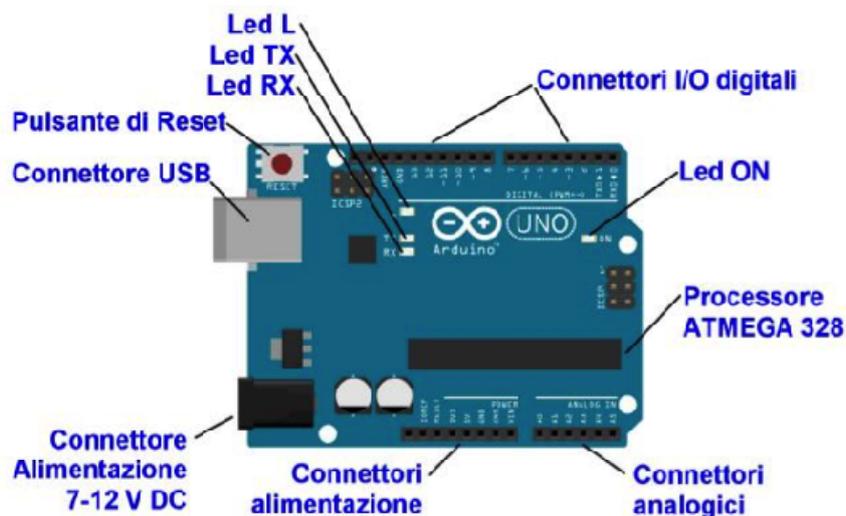
Obiettivo del programma:

Programma c++ e circuito per far lampeggiare un LED.



```
// Arduino blink facciamo lampeggiare un led
#define LED 13 // LED collegato al pin digitale 13
void setup()
{
  pinMode(LED, OUTPUT); // imposta il pin digitale come output
}
void loop()
{
  digitalWrite(LED, HIGH); // accende il LED
  delay(1000); // aspetta un second
  digitalWrite(LED, LOW); // spegne il LED
  delay(1000); // aspetta un second
}
```

Arduino Uno è un dispositivo basato su microcontrollore che permette di realizzare diversi tipi di circuiti elettronici. Possiede 15 pin digitali programmabili come ingressi o uscite e 6 ingressi per l'acquisizione ed elaborazione di segnali analogici. Il microcontrollore è l'ATmega328 prodotto da Atmel, ha una velocità di 16MHz, una memoria flash da 32KB, una sram da 2KB e una memoria EEPROM da 1KB. L'alimentazione della scheda avviene tramite porta usb o tramite l'apposito connettore.



1) Arduino-Compatible UNO R3 board x 1

2) UNO Development expansion board x 1

3) Mini bread board x 1

4) Bread board (830 fori)x 1

5) Contenitore di componenti SMD x 1

6) LED (Rossi) x5

7) LED (Gialli) x5

8) LED (Verdi) x5

9) Buzzer x2

10) Bottoni x5

11) LED display a 7 segmenti (1-digit) x 2

12) Giroscopio x 2

13) Fotoresistenza (light sensor) x 1

14) Potenzimetro x5

15) Sensore incendio x 1

16) Sensore a infrarossi x 1

17) 220 Ohm resistore x 5

18) 1k Ohm resistore x 5

19) 10k Ohm resistore x 5

20) LM35 sensore di temperatura x 1

21) Cavo usb x1

22) Connettori (wire jumper) x20

23) Plug batteria 9V + batteria 9V x 1

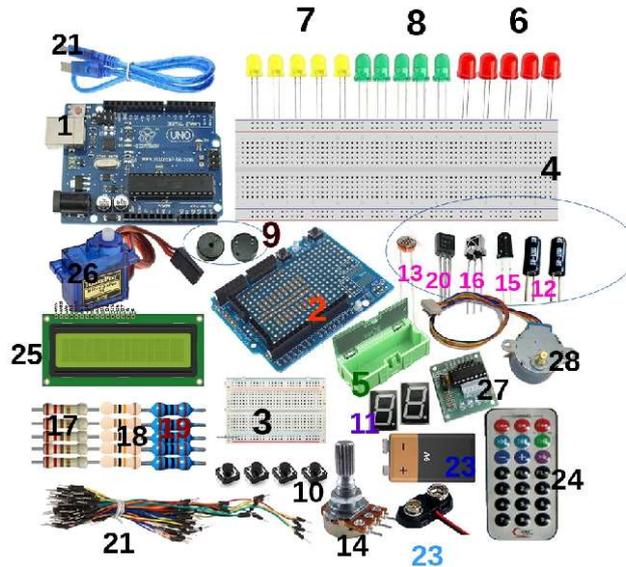
24) IR Controllo remoto x1

25) 1602 LCD modulo x 1

26) SG90 9G Servo motore x1

27) ULN2003 driver board x 1

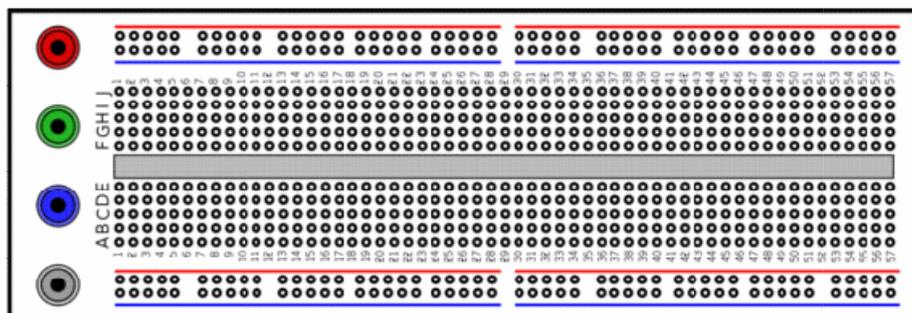
28) 5V motore a step x 1



Arduino Uno R3 Board Starter Kit with LCD Servo Motor Sensor

La **breadboard** permette di realizzare velocemente prototipi di circuiti senza bisogno di fare saldature. E' una basetta con dei fori dove, con una lieve pressione, si inseriscono i terminali dei componenti, collegandoli uno con l'altro. Il circuito così realizzato può essere facilmente modificato e smontato per riutilizzare i componenti.

I fori della breadboard sono collegati tra loro elettricamente secondo questo schema:
i fori di ogni riga rossa e blu orizzontale sono collegati tra loro
i fori delle righe verticali sono collegati tra loro cinque a cinque



I diodi **LED** (Light Emetting Diode) sono costantemente impiegati nel fermodellismo in quanto non si riscaldano eccessivamente, hanno un limitato assorbimento di corrente e sono di diversi colori, forme e dimensioni. Il terminale contraddistinto dalla lettera A è l'anodo e per distinguerlo, risulta sempre più lungo del Catodo, contraddistinto dalla lettera K.

Il catodo può essere identificato anche da una smussatura alla base del corpo cilindrico del led. Per accendere il led è necessario collegare l'anodo al positivo ed il catodo verso la massa od al negativo di alimentazione. Se accidentalmente si invertisse il collegamento il led rimarrebbe spento.

E' assolutamente necessario interporre in serie all'alimentazione, indifferente dal lato del catodo o dell'anodo, una resistenza che limiti il passaggio della corrente, in quanto il valore ottimale per il corretto funzionamento del led è compreso fra 15 e 17 milliampere.

Correnti superiori a 25 milliampere metterebbero presto il led fuori uso, mentre con correnti non inferiori a 3 milliampere è comunque assicurato un corretto funzionamento ma con minore luminosità.

La semplice formula per calcolare il valore della resistenza è la seguente:

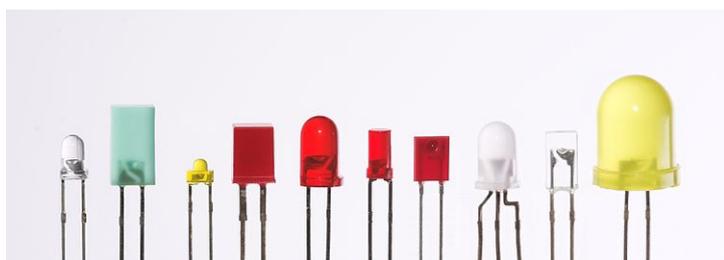
$$ohm = (V_{cc} - 1,5) : 0,016$$

Il risultato (ohm) darà il valore della resistenza da collegare al circuito.

– Vcc è la tensione continua di alimentazione indicata in Volt

– 1,5 è il valore caratteristico della caduta interna del led

– 0,016 rappresenta il valore medio di corrente in amper.



	Simbolo K = Catodo A = Anodo	Misure in mm. A = 5 - B = 5,8 C = 9 A = 3 - B = 4 C = 5,5	Colore Rosso Giallo Verde	Note Monocolore. La smussatura e la dimensione (più corto) indica il Catodo
--	---	--	---	--

ITIS-LS "Francesco Giordani" Caserta

Anno scolastico: 2019/2020

Classe 3^a sez.B spec. Informatica e telecomunicazioni– articolazione Informatica

Data:

Numero progressivo dell'esercizio: es1

Versione: 1.0

Programmatore/i:

Sistema Operativo: Windows 10

Compilatore/Interprete: Arduino 1.8.10

Obiettivo didattico: Utilizzare il costrutto "if then else"

Obiettivo del programma:

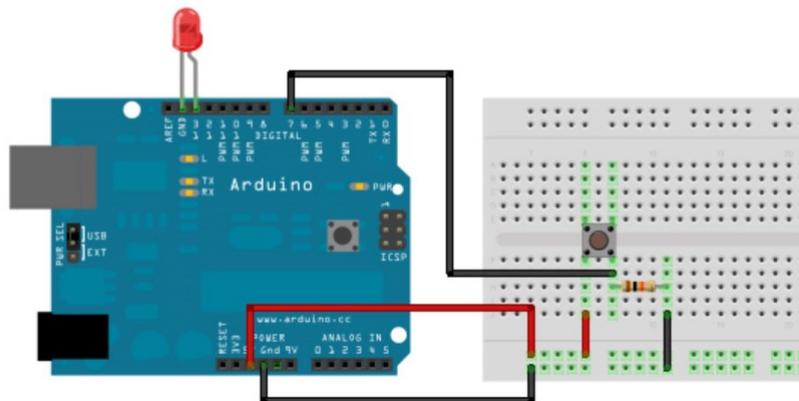
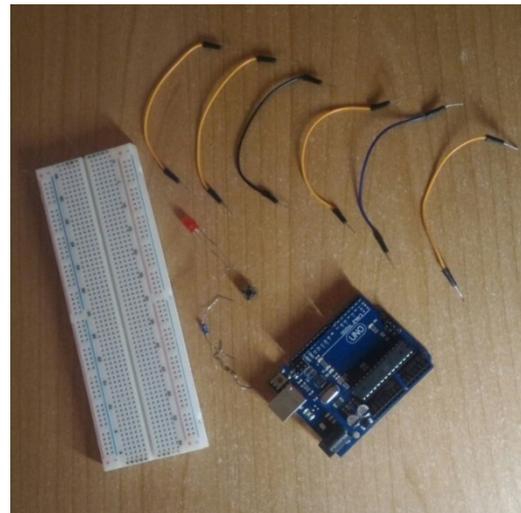
far illuminare un led mediante un pulsante

Elenco attrezzature e materiali:

scheda arduino-led-pulsante-resistore 220 ohms ,

resistore 10 k-cavi di collegamento (x6)

breadboard-cavo usb per collegare arduino con il pc



```
//pulsanteconled
```

```
#define LED 13 //led collegato al pin 13
```

```
#define BUTTON 7 //pulsante collegato al pin 7
```

```
int ButtonStato; //dichiarazione della variabile stato del pulsante
```

```
bool stato= LOW; //dichiariamo lo stato del led basso
```

```
void setup()
```

```
{
  pinMode (LED, OUTPUT); //imposta il led come output
  pinMode (BUTTON, INPUT); //imposta il pulsante come input
}

void loop()
{
  ButtonStato= digitalRead (BUTTON); //leggi stato sul pin del pulsante
  if (ButtonStato==HIGH) //se il pulsante e' premuto

  {
    if (stato==HIGH) stato=LOW; //se il led e' gia' acceso spegnilo
    else stato=HIGH; //altrimenti accendilo
  }
  digitalWrite(LED, stato); //scrivi su pin LED il contenuto della variabile stato (1/0)
  delay(100); //aspetta un decimo di secondo
}
```

ITIS-LS "Francesco Giordani" Caserta

Anno scolastico: 2019/2020

Classe 3^a sez.B spec. Informatica e telecomunicazioni– articolazione Informatica

Data:

Numero progressivo dell'esercizio: es2

Versione: 1.0

Programmatore/i:

Sistema Operativo: Windows 10

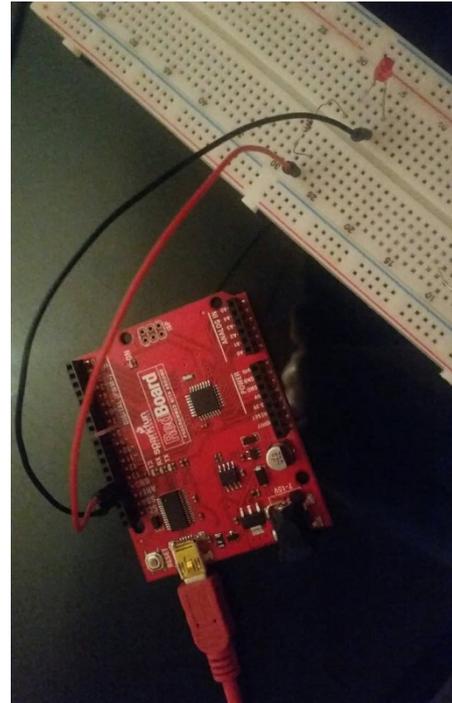
Compilatore/Interprete: Arduino 1.8.10

Obiettivo didattico: Programmare Arduino con Scratch

Obiettivo del programma:

far illuminare un led mediante un pulsante

1. Scaricare S4A dal sito <http://s4a.cat/>
2. Installare il programma S4A
3. Scaricare il firmware S4AFirmware16.ino
4. Aprire l'ambiente di programmazione Arduino, caricare S4AFirmware16.ino
5. Lanciare S4A ed attendere il riconoscimento della scheda arduino
6. Realizzare il circuito in figura
7. Scrivere il programma Scratch in figura
8. Click sulla bandierina verde



S4A 1.6

Based on Scratch from the MIT Media Lab

File Modifica Aiuto

Movimento Controllo
Aspetto Sensori
Suono Operatori
Penna Variabili

quando si clicca su

quando si preme il tasto spazio

quando si clicca su Arduino1

attendi 1 secondi

per sempre

ripeti 10 volte

invia a tutti

invia a tutti e attendi

quando ricevo

per sempre quando

se

se

Arduino1

x: 0 y: 0 direzione: 90

Script Costumi Suoni

quando si clicca su

per sempre

accendi relè 13

attendi 1 secondi

spegni relè 13

attendi 1 secondi

Arduino1 Searching

Analog0 0
Analog1 0
Analog2 0
Analog3 0
Analog4 0
Analog5 0
Digital2 false
Digital3 false

Arduino1 Searching board...

Nuovo sprite:

Arduino...

Stage

x: 147 y: 462