

CPU:
ATmega328P microcontrollore a 8 bit con clock a 16MHz

MEMORIA:
FLASH 32 KB: utilizzata per salvare il programma
SRAM 2 KB: usata per memorizzare le variabili del software
EEPROM 1KB: utilizzata per memorizzare le configurazioni

Blocchi Condizionali

```
if (condizione1)
{
  // istruzioni da eseguire se la condizione = true
}
else if (condizione2)
{
  // istruzioni da eseguire se condizione1 = false
  // e la condizione2 = true
}
else
{
  // istruzioni da eseguire se condizione1 = false
  // e la condizione2 = false
}
```

La funzione **delay** è utilizzata per **congelare** lo stato di Arduino

Esempio: delay(1000)
Questa istruzione blocca Arduino per 1000 millisecondi

Arduino

Input/Output

14 Pin di input/output digitale
6 Pin di input analogico
6 Pin di output "analogico" (PWM)

Linguaggio di programmazione: C++

- **Funzione di setup:**
Definisce la configurazione di Arduino mediante le istruzioni pinMode per settare i pin in modalità INPUT/OUTPUT
Esempio: pinMode(3,OUTPUT)

- **Funzione di loop:**
Definisce il processing svolto da Arduino utile per trasformare i relativi INPUT in degli OUTPUT desiderati

analogWrite(pin, valore)
pin: 3,5,6,9,10,11 (input)
valore: 0-255 (input)
return: nessuno (output)

Mediante l'istruzione analogWrite è possibile impostare una tensione analogica (nel range 0-5 volt) su una uscita specifica. L'istruzione analogWrite può essere usata solo su alcuni specifici PIN caratterizzati dal simbolo tilde. I valori di tensione compresi tra 0-5 volt vengono mappati nel range 0-255

Esempio: analogWrite(5,128)
Questa istruzione impone una tensione di 2,5 volt sul pin numero 5 di Arduino

Utilizzata per
Pilotare un motore DC oppure per controllare la luminosità di un led

Proporzioni
L'istruzione mappa le tensioni comprese tra 0-5 volt in un range di valori 0-255
Pertanto i valori di tensione possono essere facilmente ottenuti attraverso una semplice proporzione
tensione:5 = valore:255

int valore = analogRead(pin)
pin: A0-A5 (input)
valore: 0-1023 (output)

Mediante l'istruzione analogRead è possibile leggere una tensione analogica (nel range 0-5 volt) su un input analogico specifico. L'istruzione analogRead può essere usata solo sui pin analogici

Esempio: int valore = analogRead(A0)
Questa istruzione legge il valore di tensione analogica sul pin A0 di Arduino

Utilizzata per
Leggere i valori di tensione di un sensore (fotoresistenza, temperatura, etc) oppure su un potenziometro

int valore = digitalRead(pin)
pin: 0-13 (input)
valore: LOW-HIGH (output)

Mediante l'istruzione digitalRead è possibile leggere una tensione digitale (0/5 volt) su un input specifico. L'istruzione digitalRead può essere usata su tutti i pin digitali

Esempio: int valore = digitalRead(3)
Questa istruzione legge il valore di tensione digitale sul pin numero 3 di Arduino

Utilizzata per
Leggere i valori di tensione di un pulsante (On/Off)

digitalWrite(pin, valore)
pin: 0-13 (input)
valore: LOW-HIGH (input)
return: nessuno (output)

Mediante l'istruzione digitalWrite è possibile impostare una tensione digitale (0/5 volt) su una uscita specifica. L'istruzione digitalWrite può essere usata su tutti i pin digitali

Esempio: digitalWrite(3,HIGH)
Questa istruzione impone una tensione di 5 volt sul pin numero 3 di Arduino

Utilizzata per
Accendere/spegnere dei dispositivi

