

Il sensore di temperatura eroga una tensione da 0V a 1V; infatti, la temperatura massima è 100 °C e la tensione in uscita aumenta di 10mV per 1 °C. Quando la temperatura è di 100°C, la tensione sarà di 1000 mV e quindi di 1 V.

Se il convertitore A/D ha come Vref=5V, bisogna portare il valore massimo del sensore da 1V a 5V ed utilizzare un operazionale in configurazione non invertente.

Il MUX deve avere 8 ingressi e quindi tre linee di selezione.

Il rapporto $(S/N)_{dB} = 72dB$

$$72 \text{ dB} = 20 \log \frac{S}{N}$$

$$\log \frac{S}{N} = \frac{72}{20}$$

$$\frac{S}{N} = 10^{\frac{72}{20}}$$

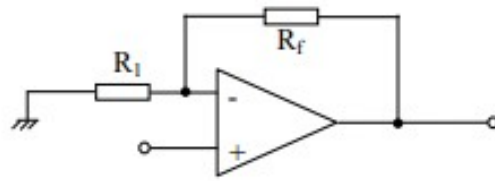
$$\frac{N}{S} = \frac{1}{10^{\frac{72}{20}}}$$

Questa informazione è utile per capire il numero di bit del convertitore A/D

Infatti, deve essere $E_q = \frac{1}{2^{n+1}} \leq \frac{N}{S}$ $10^{\frac{72}{20}} \leq 2^{n+1}$ $400 \geq 2^{n+1}$ $N=11$

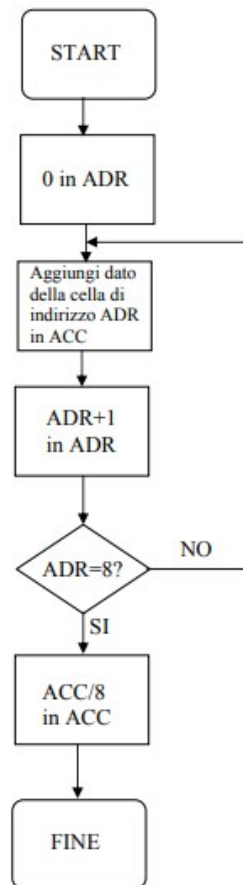
Condizionamento del segnale

$$A = V_{ref}/V_s = 5/1$$



$$A = 1 + R_f/R_1 \quad R_f/R_1 = 4$$

Fissato $R_1 = 10 \text{ K}\Omega$ si avrà $R_f = 40 \text{ K}\Omega$



La relazione tra la frequenza massima del segnale ed il tempo t_c di conversione è la seguente:

$$f_{max} \leq \frac{1}{2^n \pi t_c}$$

```
#include <LiquidCrystal.h>
```

```
#define RS 8
```

```
#define EN 9
```

```
#define D7 7
```

```
#define D6 6
```

```
#define D5 5
```

```
#define D4 4
```

```

float vref = 1.1;

LiquidCrystal lcd( RS, EN, D4, D5, D6, D7 );

void setup()
{
int lm35[8]={0,1,2,3,4,5,6,7};

  analogReference( INTERNAL ); // Vref interno all'ADC

  Serial.begin(9600);}

void loop()
{
float temp [8]= 0.0;

int val[8] = 0;

int n = 8;    //

float somma = 0.0;

for (int i = 0; i < n; i++)

{

  val[i] = analogRead( LM35[i] );

  temp = ( 100.0 * vref * val ) / 1024.0;

  somma += temp;

}

float t=somma / n;

  lcd.clear();

  lcd.setCursor( 0, 0 );

  lcd.print( "Temperatura: ");

  lcd.setCursor( 0, 1 );

  lcd.print( t );

  lcd.print( ' ' );

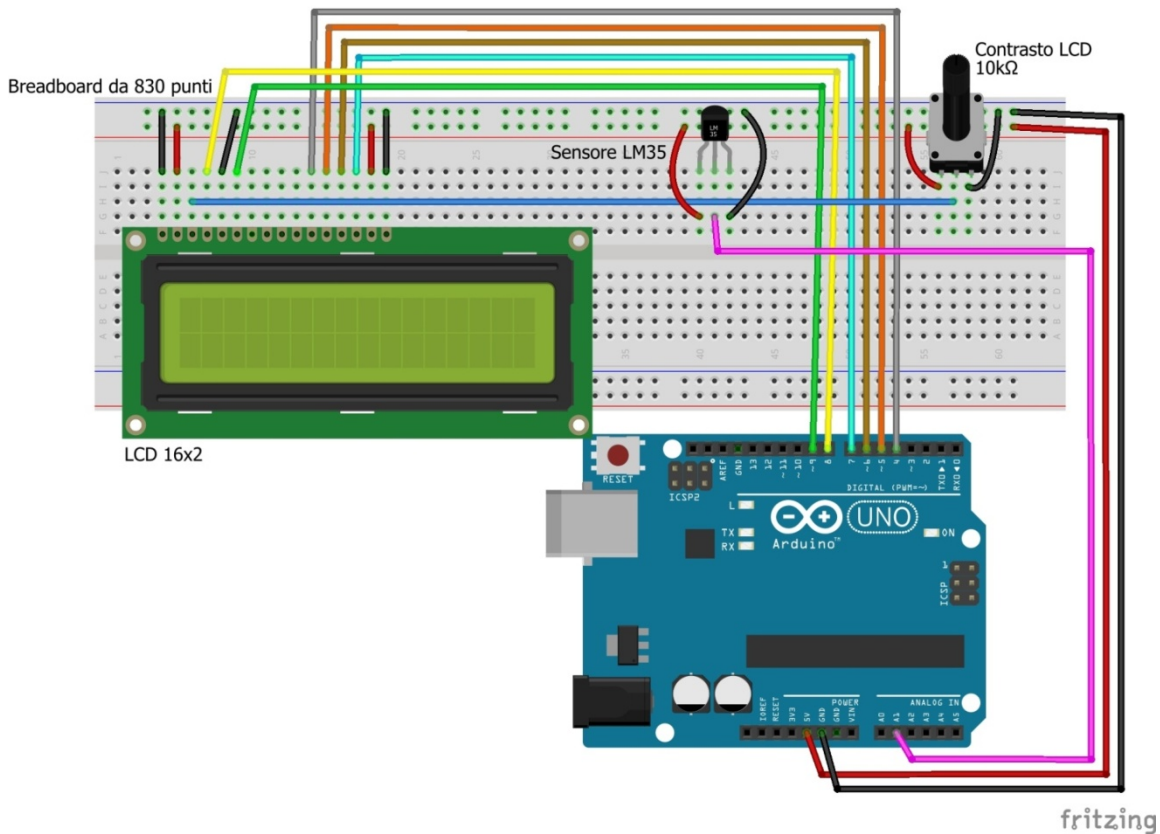
  lcd.print( ( char ) 223 );

```

```
lcd.print( 'C' );  
  
delay( 250 );
```

```
}
```

Schema Arduino UNO ed un solo sensore LM35



```
/*programma per monitorare i dati di temopratura da 6 sensori analogici*/
```

```
int lm35[]={0,1,2,3,4,5};//si dichiarano i 6 pin analogici di Arduino
```

```
float vref=1.1;//tensione di riferimento del convertitore A/D
```

```
void setup()
```

```
{
```

```
  Serial.begin(9600);}
```

```
void loop()
```

```
{
```

```

float temp [6]; //per ogni sensore si attribuisce un valore

int val[6];

int n = 56;    //

float somma = 0.0;

for (int i = 0; i < n; i++)

{

  val[i] = analogRead( A0 );

  temp[i] = ( 100.0 * vref * val[i] ) / 1024.0; // conversione a 10 bit

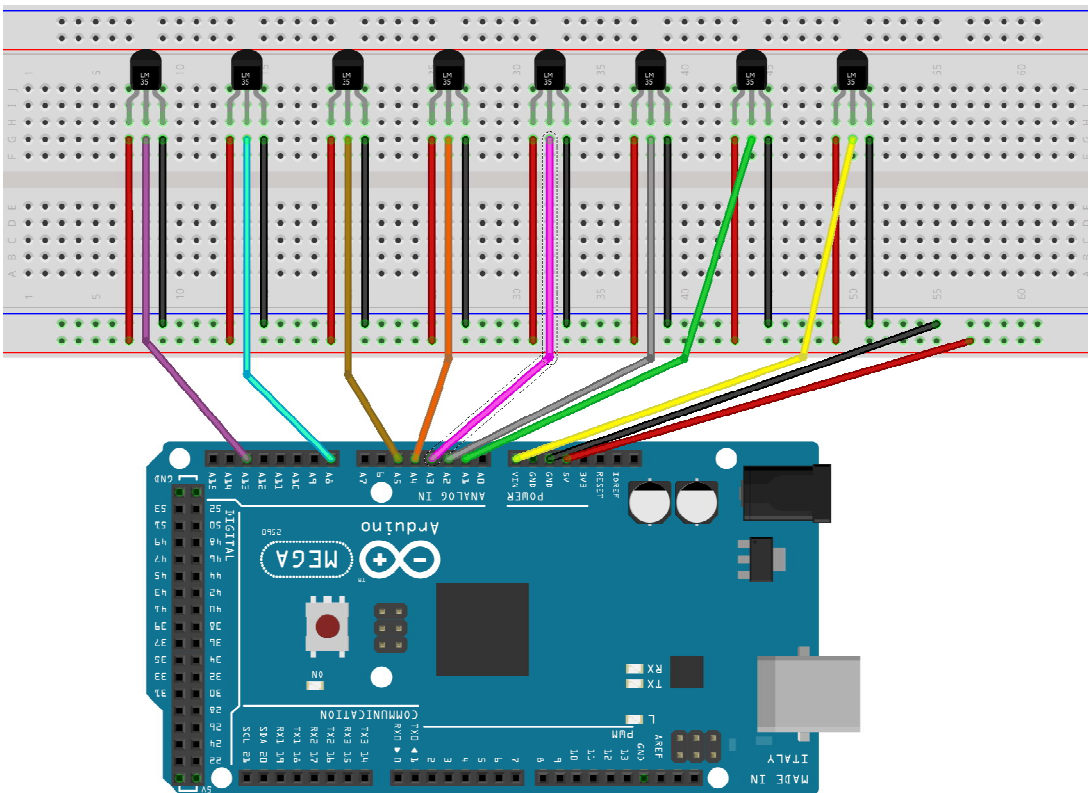
  somma += temp[i];

}

float t=somma / n; //media sui sei valori forniti dai sensori
}

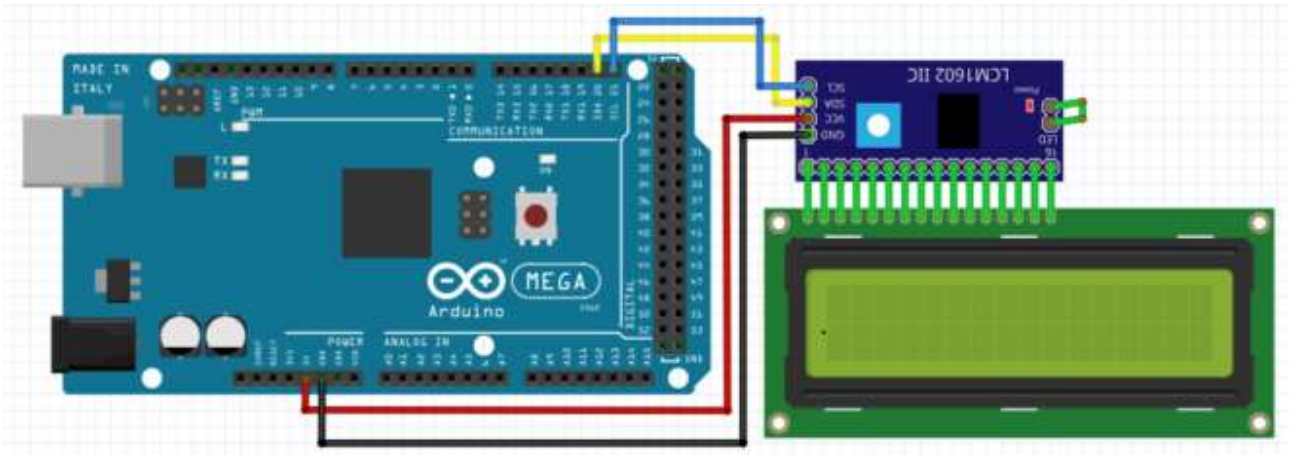
```

Con Arduino Mega si possono utilizzare 8 pin analogici come in figura



fritzing

e quindi, anche l'Icd con i2c



La sintassi base è la seguente per qualsiasi scheda Arduino

```
#include <Wire.h>
```

```
#include <LiquidCrystal_I2C.h>
```

```
LiquidCrystal_I2C lcd(0x27, 16, 2); //indirizzo del dispositivo 0x27, 16 colonne e 2 righe
```

```
void setup()
```

```
{ lcd.begin();
```

```
  lcd.backlight();
```

```
  lcd.setCursor(2, 0);
```

```
  lcd.print("Ciao Ragazzi");
```

```
  lcd.setCursor(6, 1);
```

```
  lcd.print("(GC)");
```

```
}
```

```
void loop()
```

```
{
```

```
}
```