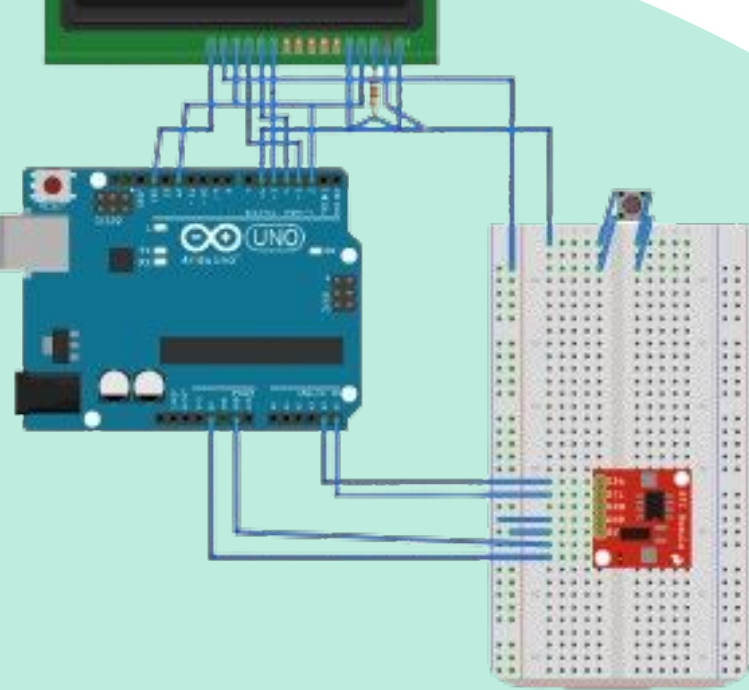


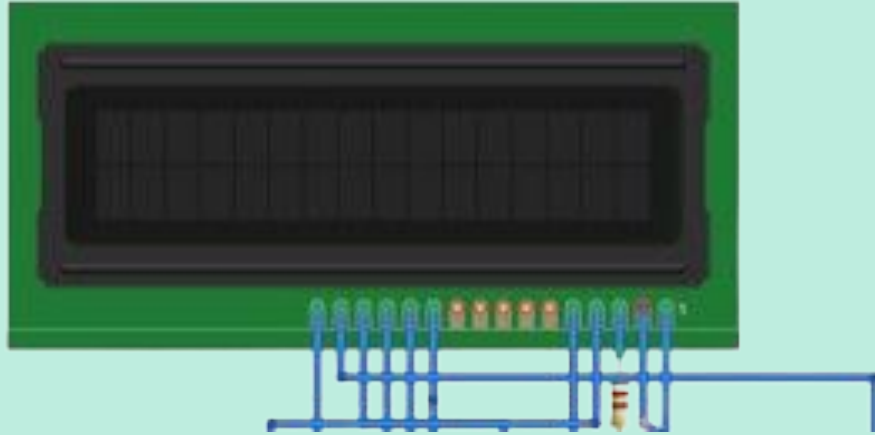
Laboratorio di Robotica

RTCM + LCD



Real Time Clock Module

Componente ideato per mantenere l'ora attuale in modo preciso e continuo, anche quando il dispositivo è spento.



Schermo LCD

Display a cristalli liquidi per visualizzare il testo, le immagini o altre informazioni. In questo caso mostrerà l'orario attuale tramite i numeri.

```

#include <Wire.h>           // Includi la libreria Wire per la comunicazione I2C
#include <RTClib.h>          // Includi la libreria RTClib per la gestione dell'RTC
#include <LiquidCrystal.h>   // Includi la libreria LiquidCrystal per la gestione dello schermo LCD

char buffer1[20];           // Dichiarazione di un buffer di caratteri per la prima riga del display LCD
char buffer2[20];           // Dichiarazione di un buffer di caratteri per la seconda riga del display LCD
RTC_DS1307 RTC;             // Inizializzazione dell'oggetto RTC per gestire l'RTC
LiquidCrystal lcd(12, 11, 5, 4, 3, 2); // Inizializzazione dell'oggetto lcd per gestire lo schermo LCD, specificando i pin a cui è collegato

const int setpin = 6;       // Dichiarazione della costante setpin per il pin di impostazione
boolean statepin = 0;       // Dichiarazione della variabile booleana statepin per memorizzare lo stato del pin di impostazione

void setup() {
  pinMode(6, INPUT);        // Imposta il pin setpin come input
  digitalWrite(6, HIGH);    // Abilita la resistenza di pull-up interna sul pin setpin
  Wire.begin();             // Inizia la comunicazione I2C
  RTC.begin();              // Inizia la comunicazione con l'RTC
  lcd.begin(16, 2);         // Inizializza lo schermo LCD con una dimensione di 16 colonne per 2 righe

  if (! RTC.isrunning()) {   // Verifica se l'RTC non è in esecuzione (cioè se non è impostato correttamente)
    RTC.adjust(DateTime(__DATE__, __TIME__)); // Imposta l'RTC alla data e ora attuali
  }
}

void loop() {
  statepin = digitalRead(setpin); // Leggi lo stato del pin di impostazione e memorizzalo in statepin

  if (statepin == LOW) {     // Se il pin di impostazione è a LOW (ossia è attivato)
    RTC.adjust(DateTime(__DATE__, __TIME__)); // Imposta l'RTC alla data e ora attuali
  }

  DateTime now = RTC.now(); // Leggi l'ora attuale dall'RTC
  sprintf(buffer1, "%02d:%02d:%02d", now.hour(), now.minute(), now.second()); // Formatta l'ora corrente come una stringa e memorizzala in buffer1
  lcd.setCursor(0,0);        // Imposta il cursore dello schermo LCD alla posizione (0,0) (prima riga)
  lcd.print(buffer1);        // Stampare buffer1 sulla prima riga del display LCD

  sprintf(buffer2, "%02d/%02d/%04d", now.day(), now.month(), now.year()); // Formatta la data corrente come una stringa e memorizzala in buffer2
  lcd.setCursor(0,1);        // Imposta il cursore dello schermo LCD alla posizione (0,1) (seconda riga)
  lcd.print(buffer2);        // Stampare buffer2 sulla seconda riga del display LCD

  delay(1000);              // Attendi un secondo prima di ripetere il loop
}

```

Codice