

QUADERNO  
DI  
PROGETTO

CocktailMachine





## CONTESTO E OBIETTIVO

Gli studenti Daniele Digirolamo, Michele Lillo del c.d.l. ITPS 1° anno realizzano sotto supervisione del professore Mario A. Bochicchio, un robot capace di creare cocktail progettato con l'utilizzo di Arduino. Il robot, composto da pompe e tubi, è in grado di erogare più di un liquido, andando a formare, grazie alle varie miscele, alcuni dei cocktail più conosciuti. Per interracciarsi al robot vengono usati dei bottoni per scorrere i cocktails disponibili visibili su un display e selezionare quello desiderato.

---

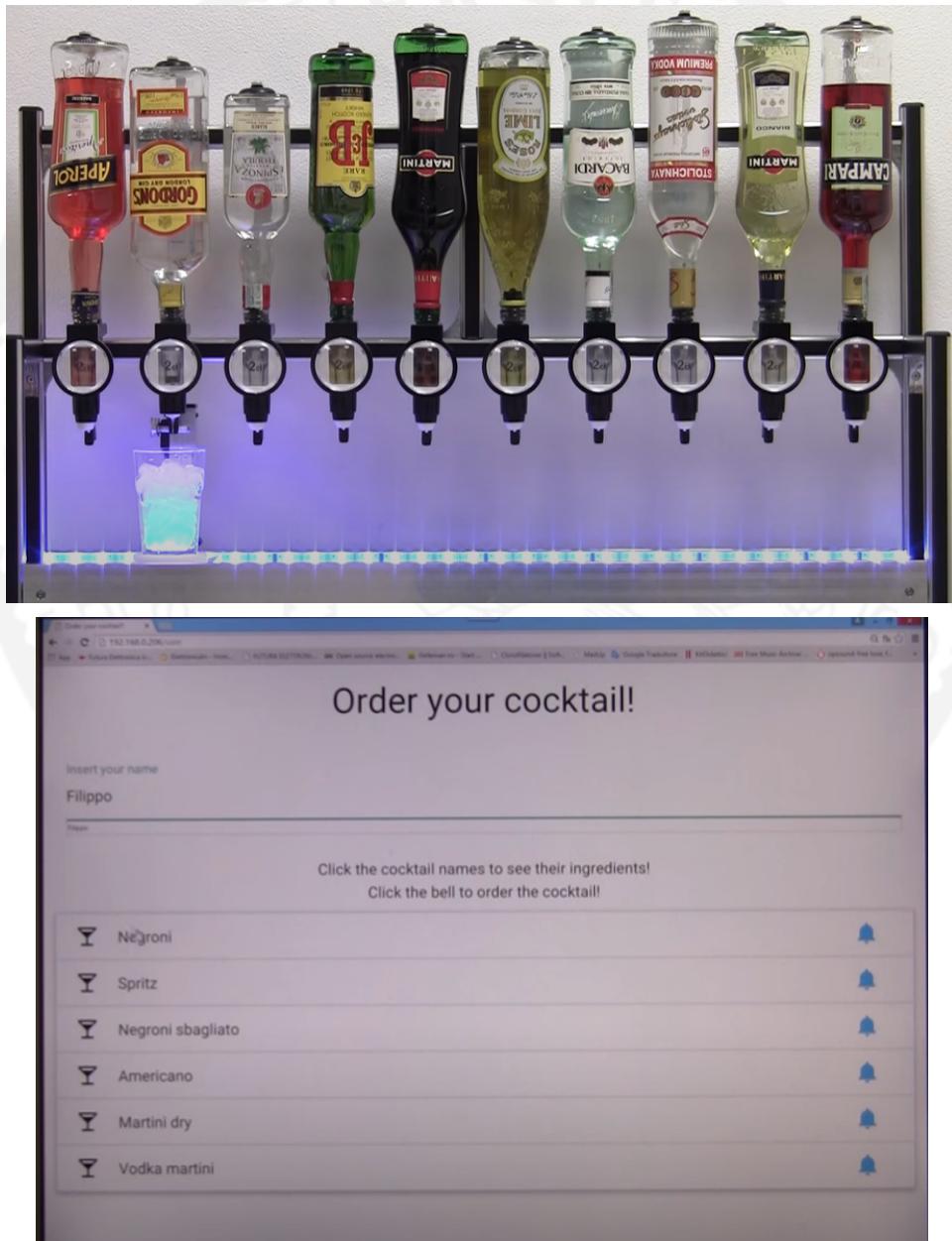
# STATO DELL'ARTE

Durante la ricerca in rete di precedenti esempi riguardanti il nostro progetto, abbiamo individuato 2 principali metodi di realizzazione, con diverse caratteristiche e punti di forza.



## ESEMPIO 1 : NASTRO TRASPORTATORE

Il primo esempio è realizzato mettendo in alto, capovolte, le varie bottiglie necessarie per comporre i cocktail, collegandole a un dosatore, capace quindi di misurare precisamente la quantità di liquido che fuoriesce in una singola erogazione. Ogni dosatore è dotato di una leva. Sotto le bottiglie è presente un nastro trasportatore su cui è poggiata una base, con a sua volta un bicchiere sopra, che in base al cocktail (selezionato tramite collegamento bluetooth a un'app) posiziona il bicchiere sotto la bottiglia desiderata, azionando la leva e aprendo il dosatore. Una volta terminata l'erogazione di una singola bottiglia il bicchiere si muove verso la prossima bottiglia, fino al completamento del cocktail.



[Modello con nastro trasportatore](#)

/APPENDICI NEL TESTO

## ESEMPIO 2 : POMPE AD ASPIRAZIONE

Il secondo esempio è realizzato inserendo nelle varie bottiglie, posizionate in piedi dove si desidera, dei tubicini, collegati a loro volta a una pompa, che aspirando porta il liquido dalla bottiglia al bicchiere. In base al tempo di attivazione della pompa si può individuare la quantità di liquido da erogare. Il bicchiere è posto in un punto fisso. Essendo ogni bottiglia collegata a una pompa indipendente, è possibile effettuare un' erogazione simultanea da più bottiglie.



[Modello con pompe ad aspirazione](#)

/APPENDICI NEL TESTO

---

## RIASSUNTO CARATTERISTICHE PRINCIPALI

- NASTRO TRASPORTATORE
  - Bottiglie capovolte
  - Bicchiere in movimento
  - Quantità liquido gestito dal numero di volte che vengono aperti i dosatori
  - Leva che apre i dosatori
  - App per scelta dei cocktail
  - Erogazione singola
  
- POMPE AD ASPIRAZIONE
  - Aspirazione del liquido
  - Tubicini per alimenti tra bottiglia e pompa
  - Bicchiere in un punto fisso
  - Bottiglie in piedi
  - Quantità liquido gestito dal tempo di aspirazione della pompa
  - App per scelta dei cocktail
  - Erogazione simultanea

---

# PROGETTO E REALIZZAZIONE DI ROBOT PER COCKTAIL

Sulla base di quanto visto nello [STATO DELL'ARTE](#), abbiamo raccolto le caratteristiche migliori dei due progetti d'esempio ed unito altre idee realizzando così il nostro.

## CARATTERISTICHE

- Bottiglie in piedi
- Bicchiere in un punto fisso
- Controllo alla scelta della disponibilità del liquido
- Tubicini per alimenti dalla bottiglia al collettore che verserà tutto nel bicchiere
- Pompe peristaltiche per aspirare il liquido dalla bottiglia
- Display per visualizzare le scelte e lo stato di preparazione del cocktail
- 3 bottoni di selezione: avanti, indietro, seleziona
- Erogazione sequenziale

# IMPLEMENTAZIONE

Per la realizzazione di questo progetto abbiamo usufruito di:

1. 4x pompa peristaltica 12V	€5 cad.
2. 4x relè 5V	€1 cad.
3. 1x breadboard	€2 cad.
4. 1x Arduino UNO	€5 cad.
5. 1x buzzer attivo	€0.10 cad.
6. 1x display LCD 16x2	€4 cad.
7. 1x modulo step-up&down	€0.50 cad.
8. 3x pulsanti	€0.05 cad.
9. 1x alimentatore 12V	già disponibile in casa

Per iniziare abbiamo fatto tutti i collegamenti necessari a tutti i componenti, nello specifico abbiamo collegato:

- Ogni pompa ad un relè 12V
- L'alimentazione di uscita dei relè all'alimentatore 12V
- Ogni input del relè ai pin di output di Arduino per ricevere l'impulso e cambiare stato
- L'ingresso del modulo step-up&down all'alimentatore 12V per avere in uscita 5V
- L'uscita del modulo ster-up&down all'alimentazione di Arduino
- I 3 bottoni a 3 pin di input diversi per associare ad ognuno di essi un comando,
- Il buzzer ad un pin di output per riprodurre un "bip" quando si riceve un comando proveniente dai pulsanti.
- Il Display LCD i pin SCL e SDA di Arduino

Dopo aver collegato tutto abbiamo scritto il seguente codice sull'IDE e poi implementato su Arduino

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

int pump1=2; //pin digitale 2 a cui è collegata la pompa N1 //RUM
int pump2=3; //pin digitale 3 a cui è collegata la pompa N2 //GIN
int pump3=4; //pin digitale 4 a cui è collegata la pompa N3 //VODKA
int pump4=5; //pin digitale 5 a cui è collegata la pompa N4 //APEROL

int buzz=10; //pin digitale 10 a cui è collegato il buzzer

int pulsanteSX=11; //pin digitale 11 a cui è collegato il pulsante sinistro
int pulsanteDX=12; //pin digitale 12 a cui è collegato il pulsante destro
int pulsanteOK=13; //pin digitale 13 a cui è collegato il pulsante di selezione
//dichiarazione delle variabili in cui verrà memorizzato il valore della digitalRead (0 non premuto, 1 premuto)
int valDX,valOK,valSX;

//inizializzazione delle variabili in cui verrà memorizzato lo stato precedente del pulsante
int valDXOld=0,valOKOld=0,valSXOld=0; // per impedire che vada in uno scroll veloce
```

```

const int nCocktail=5; //numero dei cocktail memorizzati
char* cocktail[nCocktail]={"Mojito","Americano","Moscow Mule","Gin Tonic","Aperol Spritz"}; //array di stringhe
int puntaCocktail=-1; //indica il numero del cocktail selezionato, inizializzato a -1 perchè nessuno è selezionato

//inizializzazione della libreria in cui è descritta la modalità di utilizzo dei pin
LiquidCrystal_I2C lcd(0x27,16,2); // impostazione dell'indirizzo dell'LCD 0x27 di 16 caratteri e 2 linee

//inizializzazione delle variabili per il multithreading
const int RSdelay = 10000; //ReStart delay, tempo da trascorrere prima di resettare la selezione (in ms)
unsigned long RStime = millis(); //ReStart time, inizializzato al tempo trascorso dall'avvio del programma (in ms)

#####
void setup()
{
    lcd.begin(); //inizializzazione dell'LCD
    lcd.backlight(); //attivazione della retroilluminazione

    //imposta i pin a cui sono collegate le pompe ad OUTPUT
    pinMode(pump1,OUTPUT);
    pinMode(pump2,OUTPUT);
    pinMode(pump3,OUTPUT);
    pinMode(pump4,OUTPUT);

    pinMode(buzz,OUTPUT); //imposta il pin collegato al buzzer come OUTPUT

    //imposta i pin a cui sono collegati i pulsanti ad INPUT
    pinMode(pulsanteDX,INPUT);
    pinMode(pulsanteOK,INPUT);
    pinMode(pulsanteSX,INPUT);

    avvio();
}

#####
void loop()
{
    if(puntaCocktail == -1) //se non è selezionato nessun cocktail
    {
        digitalWrite(pump1,HIGH); //attiva la prima pompa (RUM)
        digitalWrite(pump2,HIGH); //disattiva la prima pompa //2ml
        digitalWrite(pump3,HIGH); //disattiva la prima pompa //2ml
    }

    valDX=digitalRead(pulsanteDX); //lettura dell'input (pulsante) e memorizzazione in valDX
    valOK=digitalRead(pulsanteOK); //lettura dell'input (pulsante) e memorizzazione in valOK
    valSX=digitalRead(pulsanteSX); //lettura dell'input (pulsante) e memorizzazione in valSX

    //viene controllato se l'input sia HIGH (pulsante premuto) e prima prima fosse LOW (non premuto)
    if((valDX==HIGH)&&(valDXOld==LOW)){
        buzz(); //il buzzer riproduce un suono al click del pulsante
        puntaCocktail++; //incrementa il contatore dei cocktail
        if(puntaCocktail>nCocktail-1) puntaCocktail=0; //se è maggiore del numero dei cocktail, torna al primo
        cambiococktail();
        delay(250); //antirimbalzo, si attende che l'input si stabilizzi
    }
    valDXOld=valDX; //imposta lo stato precedente del pulsante destro al valore attuale
}

```

```

if((valSX==HIGH)&&(valSXOld==LOW)){
    buzzer(); //il buzzer riproduce un suono al click del pulsante
    puntaCocktail--; //decrementa il contatore dei cocktail
    if(puntaCocktail<0) puntaCocktail=nCocktail-1; //se è minore di 0, torna all'ultimo
    cambiococktail();
    delay(250); //antirimbalzo, si attende che l'input si stabilizzi
}
valSXOld=valSX; //imposta lo stato precedente del pulsante sinistro al valore attuale

if((valOK==HIGH)&&(valOKOld==LOW)){
    if(puntaCocktail==-1){
        digitalWrite(buzz,HIGH); //imposta ad HIGH (passa corrente) il pin del buzzer
        delay(60); //aspetta
        digitalWrite(buzz,LOW); //imposta a LOW (non passa corrente) il pin del buzzer
        delay(40); //aspetta
        digitalWrite(buzz,HIGH); //imposta ad HIGH (passa corrente) il pin del buzzer
        delay(60); //aspetta
        digitalWrite(buzz,LOW); //imposta a LOW (non passa corrente) il pin del buzzer
        delay(40); //aspetta
        digitalWrite(buzz,HIGH); //imposta ad HIGH (passa corrente) il pin del buzzer
        delay(60); //aspetta
        digitalWrite(buzz,LOW); //imposta a LOW (non passa corrente) il pin del buzzer

        lcd.clear(); //pulisce lo schermo
        lcd.setCursor(0,0); //posiziona cursore in colonna 0 e riga 0
        lcd.print("Prima seleziona"); //stampa del testo su display
        lcd.setCursor(0,1); //posiziona cursore in colonna 0 e riga 1
        lcd.print("il cocktail"); //stampa del testo su display
        delay(4000); //aspetta 4 secondi
        avvio();
    }
    else{
        //riproduce il suono per 350ms
        digitalWrite(buzz,HIGH);
        delay(350);
        digitalWrite(buzz,LOW);

        lcd.setCursor(0,0); //posiziona cursore in colonna 0 e riga 0
        lcd.print("Sto preparando: "); //stampa del testo su display
        prepara(puntaCocktail);
        delay(250); //antirimbalzo, si attende che l'input si stabilizzi
    }
}
valOKOld=valOK;
if(puntaCocktail!=-1) restart();
}

#####
void buzzer(){
    //riproduce il suono per 150ms
    digitalWrite(buzz,HIGH); //imposta ad HIGH (passa corrente) il pin del buzzer
    delay(150); //aspetta
    digitalWrite(buzz,LOW); //imposta a LOW (non passa corrente) il pin del buzzer
}
void pulisciriga1(){

```

```

//pulisce la riga 1
lcd.setCursor(0,1); //posiziona cursore in colonna 0 e riga 1
lcd.print("          "); //stampa 16 spazi (riga vuota)
}

void cambiococktail(){
RStime = millis(); //aggiorna il tempo passato dall'avvio del programma nella sua variabile
pulisciriga1();
lcd.setCursor(0,1); //posiziona cursore in colonna 0 e riga 1
lcd.print(cocktail[puntaCocktail]); //stampa testo dell'array in posizione selezionata
}

void restart(){
if( (millis() - RStime) > RSdelay ){ //se è passato più tempo del
    puntaCocktail=-1; //resetta la selezione del cocktail
    avvio();
}
}

void avvio(){
//testo iniziale
lcd.setCursor(0,0); //posiziona cursore in colonna 0 e riga 0
lcd.print("COCKTAIL MACHINE"); //stampa testo su display
lcd.setCursor(0,1); //posiziona cursore in colonna 0 e riga 1
lcd.print("Selezionane uno!"); //stampa testo su display
}

void prepara(int x){
switch(x){
    case 0:
        //in 10 secondi una pompa riempie circa 20ml
        //ergo, 2ml al secondo

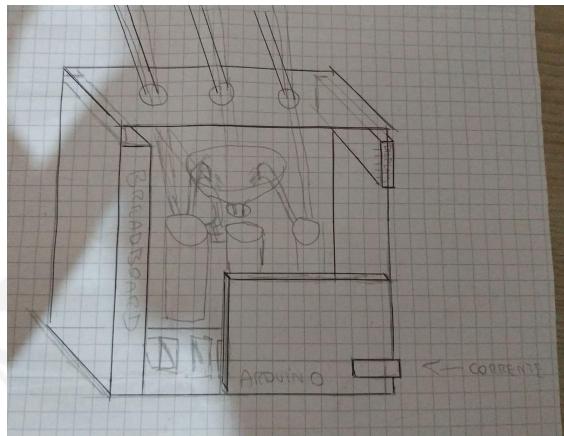
        digitalWrite(pump1,LOW); //disattiva la prima pompa //2ml
        delay(10000); //eroga per 2 secondi (1sec=1ml)
        digitalWrite(pump1,HIGH); //attiva la prima pompa (RUM)

        delay(1000);
        digitalWrite(pump2,LOW); //disattiva la prima pompa //2ml
        delay(2000); //eroga per 2 secondi (1sec=1ml)
        digitalWrite(pump2,HIGH); //attiva la prima pompa (RUM)

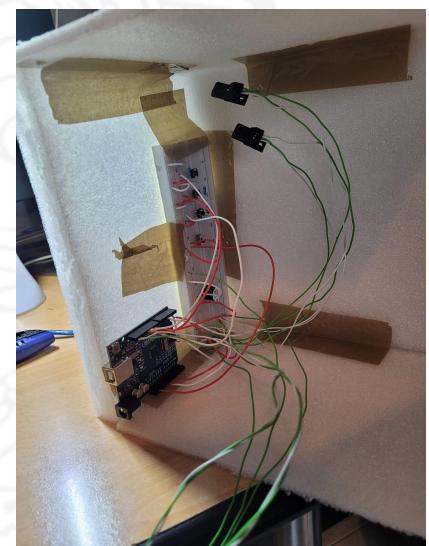
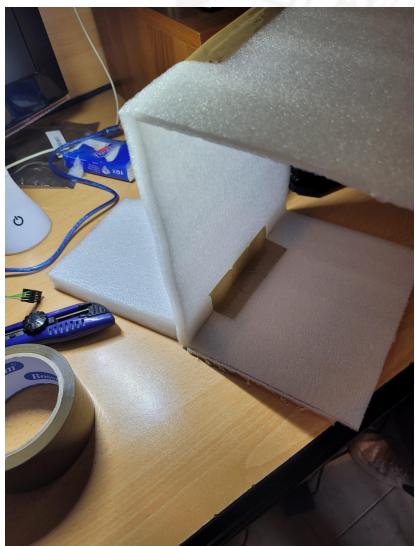
        delay(1000);
        digitalWrite(pump3,LOW); //disattiva la prima pompa //2ml
        delay(2000); //eroga per 2 secondi (1sec=1ml)
        digitalWrite(pump3,HIGH); //attiva la prima pompa (RUM)
        break;
}
puntaCocktail=-1;
delay(3000); //dopo che ha finito, aspetta 3 secondi e stampa
lcd.clear();
lcd.setCursor(0,0); //posiziona cursore in colonna 0 e riga 0
lcd.print("COCKTAIL PRONTO!"); //stampa testo su display
lcd.setCursor(0,1); //posiziona cursore in colonna 0 e riga 1
lcd.print(" Goditelo!"); //stampa testo su display
delay(3000);
avvio();
}

```

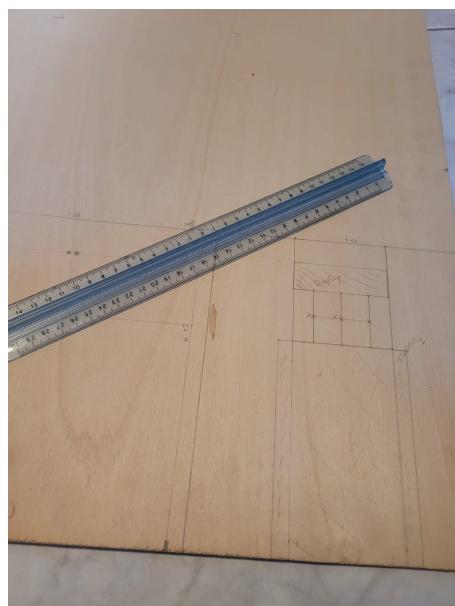
Per concludere abbiamo creato una struttura per contenere tutti i componenti e rendere agibile e confortevole la "CocktailMachine".



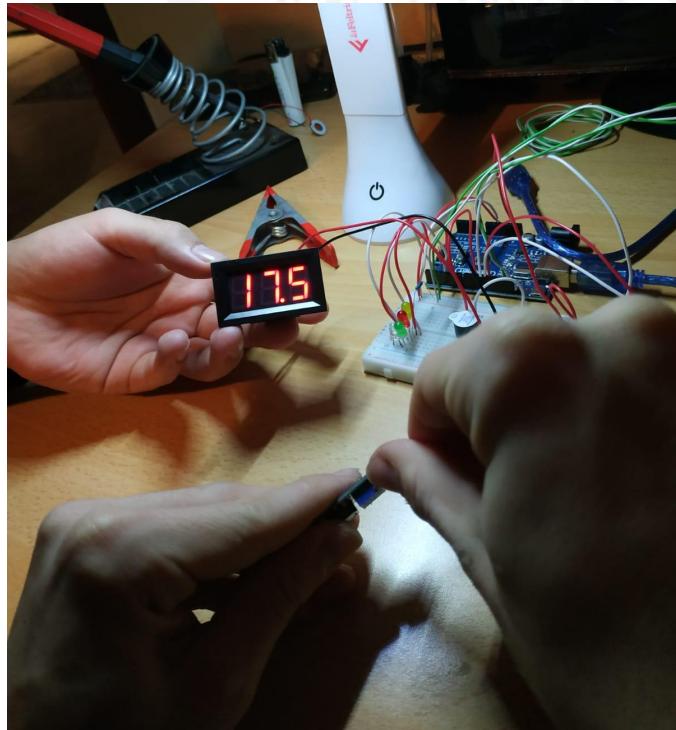
Come prima struttura di prova abbiamo utilizzato del polistirolo, in modo da capire bene quanto grande dovesse essere e come posizionare i componenti al suo interno.



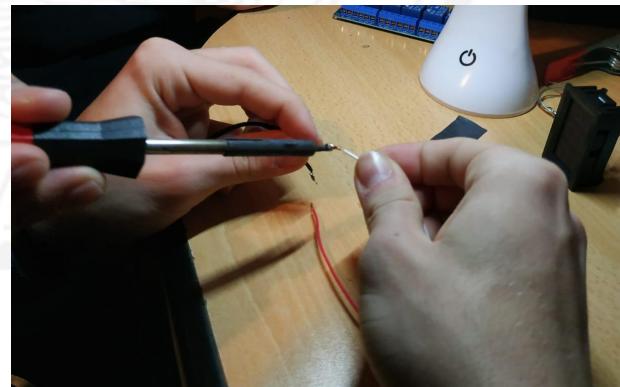
Per poi riprodurla in compensato, in modo da essere più solida e resistente.



## TEST



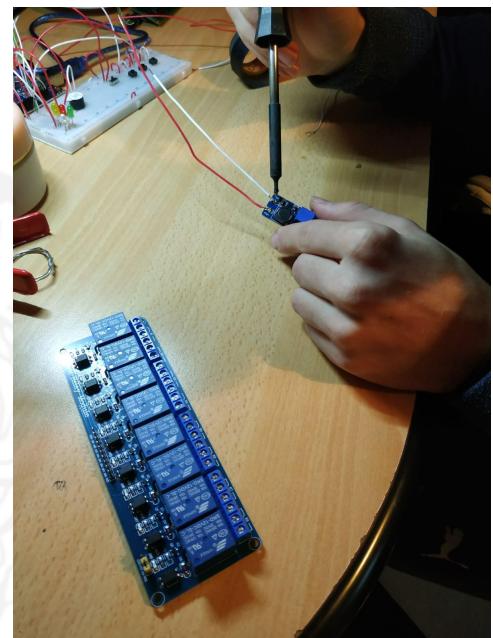
Uno dei problemi che abbiamo riscontrato è stato nel convertire i 12V emessi dall'alimentatore per alimentare le pompe, in 5V per passarli ad arduino. Ci siamo accorti che il nostro step-up non andava anche in step-down, quindi non convertiva in una Voltaggio più basso. Alche abbiamo dovuto cambiare la struttura in ingresso della corrente prendendo 5V dall'alimentatore.



---

Un altro problema sorto subito dopo è che se da arduino stesso convertiamo in 12V per le pompe, non riusciva a reggere la tensione, alimentando anche il resto, causando un rallentamento della pompa e degli sbalzi al display. Quindi abbiamo deciso di sdoppiare il filo che alimenta arduino e collegarlo direttamente allo step-up, senza passare per arduino appunto.

Dopo essere riusciti a collegare correttamente i relè alla breadboard, alimentati esternamente da arduino, li abbiamo collegati con le pompe. In fase di esecuzione, però, non cambiavano stato anche se la corrente passava.



---

## CONCLUSIONI E SVILUPPI FUTURI

