

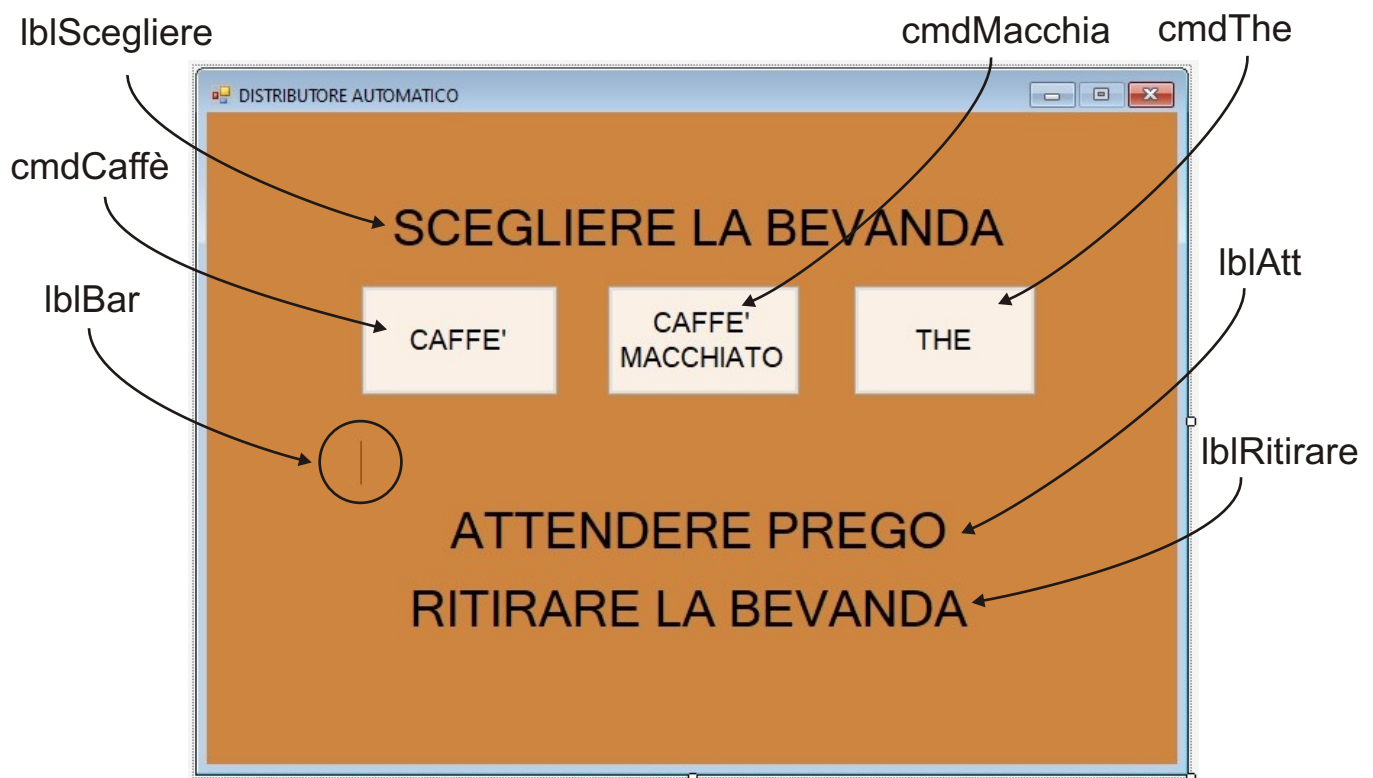
Gli approfondimenti di LABOROBOTICA

DISTRIBUTORE AUTOMATICO DI BEVANDE

Ci è stato chiesto di progettare il software per un distributore automatico. Dovremo fare in modo che tramite un display touch si possano selezionare tre tipi di bevanda, con comunicazione via seriale agli attuatori che forniranno il prodotto.

- I pulsanti saranno disattivati fino a quando il cliente non avrà inserito la moneta(o le monete).
- In questo caso la scritta visibile sul pannello dovrà essere "INSERIRE LE MONETE"
- Dopo aver inserito la moneta e verificato l'importo la scritta dovrà essere "SCEGLIERE LA BEVANDA"
- I pulsanti a questo punto si attiveranno.
- Dopo aver scelto la bevanda, una progress bar marrone dovrà scandire il tempo di preparazione.
- La scritta dovrà essere "ATTENDERE PREGO".
- Alla fine del processo di preparazione la scritta sarà "RITIRARE LA BEVANDA"
- Il processo a questo punto viene ripetuto dall'inizio.

In figura sono illustrati tutti gli elementi che andranno a costituire l'interfaccia utente.



serialPort1 Timer1

Si dovrà predisporre una comunicazione bidirezionale C#<-> Arduino per poter avere la conferma della moneta inserita, poi per comunicare alla macchina l'avvio dei processi (simulati da accensione di diodi led).

LABOROBOTICA

IL SOFTWARE IN C#

```
using System;
using System.Drawing;
using System.Windows.Forms;

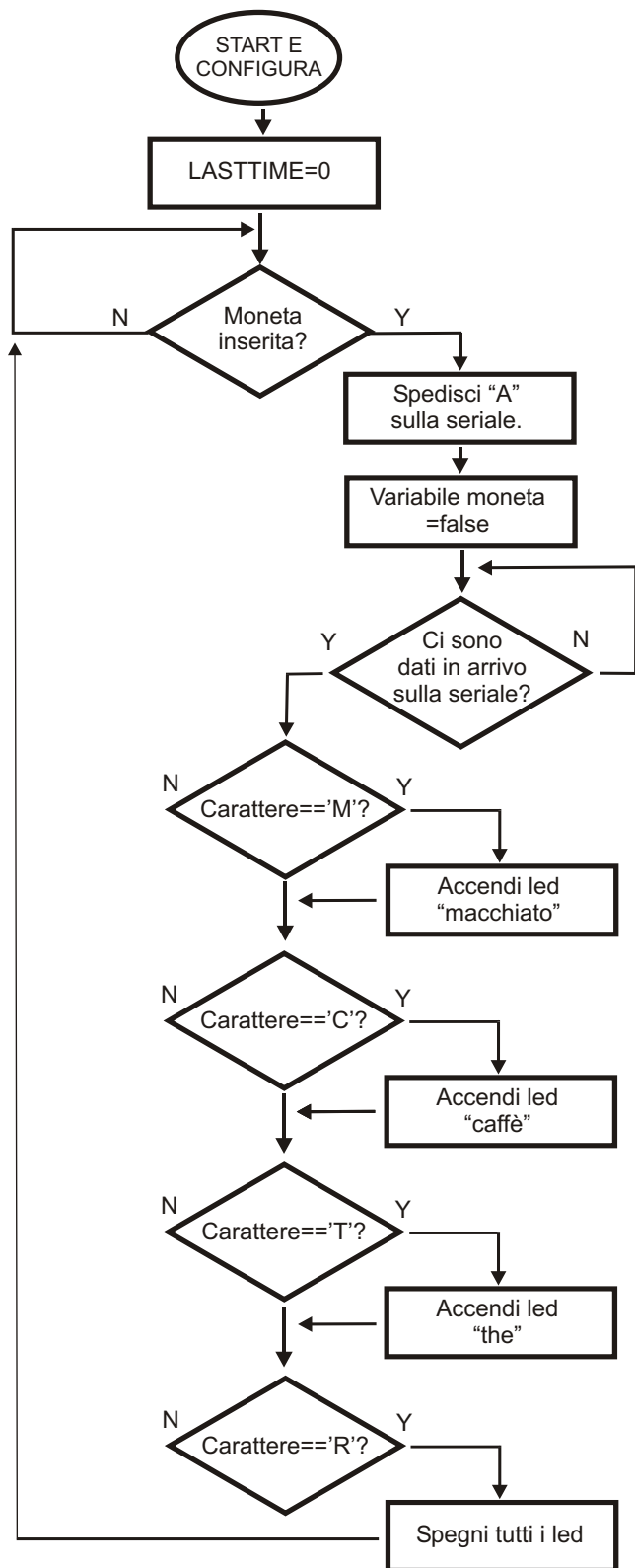
namespace progetto
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
            serialPort1.Open();    //apre la seriale

        }
        int n;
        int flag = 1;
        string monetaon;
        //assegna la variabile che identifica l'inserimento della moneta

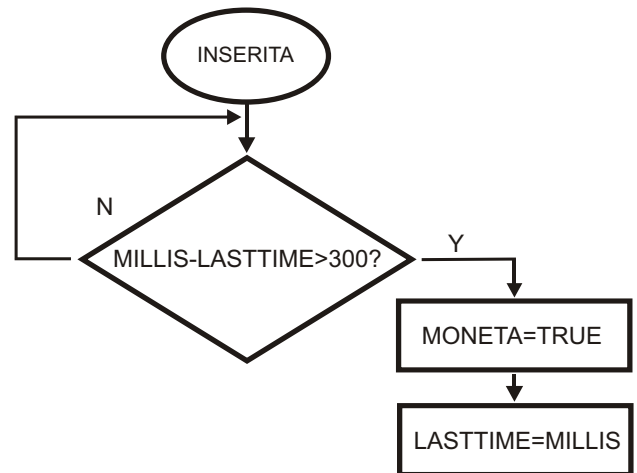
        private void serialPort1_DataReceived(object sender, System.IO.Ports.SerialDataReceivedEventArgs e)
        //IMPORTANTE!! per ottenere questo private void selezionare l'icona della seriale, poi portarsi
        //sulla casella "Eventi" delle propriet? (quella con il fulmine) e fare doppio click su "Data Received"
        {
            monetaon = serialPort1.ReadLine();    //quando arriva una stringa sulla seriale
            this.Invoke(new EventHandler(continua));    //chiama il private void continua
        }
        private void continua(object sender, EventArgs e)
        {
            if(monetaon.Substring(0,1)=="A")    //tratta il dato in arrivo come una stringa
            {
                lblInserire.Visible = false;
                lblScegliere.Visible = true;    //visualizza la label "SCEGLIERE BEVANDA"
                lblAtt.Visible = false;
                cmdCaffe.Enabled = true;    //abilita l'uso dei button (dichiarati disabilitati in fase di creazione)
                cmdMacchia.Enabled = true;
                cmdThe.Enabled = true;
            }
        }
        private void cmdCaffe_Click(object sender, EventArgs e) //cliccando sul button cmdCaffe
        {
            cmdCaffe.BackColor = Color.Sienna;    //cambia lo sfondo del button
            lblAtt.Visible = true;    //visualizza la label "ATTENDERE PREGO"
            lblScegliere.Visible = false;    //nasconde la label "SCEGLIERE BEVANDA"
            serialPort1.Write("C");    //manda ad Arduino su seriale il carattere C
            timer1.Enabled = true;    //abilita il timer 1
        }

        private void cmdMacchia_Click(object sender, EventArgs e)
        {
            cmdMacchia.BackColor = Color.Sienna;
            lblAtt.Visible = true;
            lblScegliere.Visible = false;
            serialPort1.Write("M");
            timer1.Enabled = true;
        }
    }
}
```

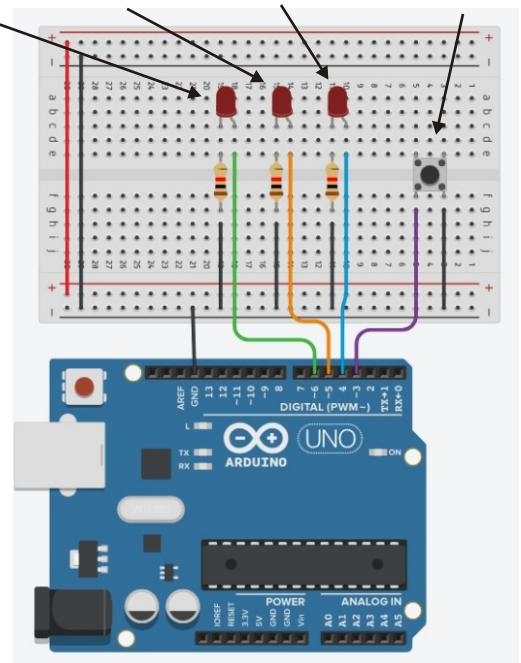

IL FLOW CHART DELLO SKETCH DI ARDUINO



Ogni volta che viene premuto il pulsante (moneta inserita) viene generato un interrupt che "chiama" la subroutine "INSERITA"



THE MACCHIATO CAFFÈ MONETA



LO SCHEMA DI MONTAGGIO

La spiegazione della gestione software e hardware dei rimbalzi di un pulsante durante la funzione di interrupt è descritta da pag.180 a pag.185 del libro Laborobotica vol.A (ed. Hoepli)

LO SKETCH DI ARDUINO

```
//24 09 2023 sketch che comunica con C# accendendo simulando
//l'inserzione di una moneta e accendendo i led relativi ai processi
//di funzionamento.
int pinMoneta=3; //pin al quale viene comunicato l'inserzione di una moneta
volatile bool moneta=false;
volatile unsigned long lastTime=0; //variabile per il debouncing
char carattere;

int caffe=4; //led relativo al procedimento "Caffe'"
int macchiato=5;
int the=6;

void setup()
{
  pinMode(caffe,OUTPUT);
  pinMode(macchiato,OUTPUT);
  pinMode(the,OUTPUT);
  attachInterrupt(1,inserita,FALLING); //interrupt associato al pin 3
  Serial.begin(9600);
  pinMode(pinMoneta, INPUT_PULLUP);
}
void loop()
{
  if(moneta==true) //se la moneta e' stata inserita
  {
    String comando ="A";
    Serial.println(comando); //spedisci sulla seriale la stringa "A"
    moneta=false;
  }

  if(Serial.available(>0) //controlla la seriale in arrivo
  {
    carattere=Serial.read(); //associa a "carattere" il dato in arrivo
    if(carattere=='M')// se il carattere e' "M"
    {
      digitalWrite(macchiato,HIGH); //accendi il led "caffè macchiato"
    }
    if(carattere=='C')
      digitalWrite(caffe,HIGH);

    if(carattere=='T')
      digitalWrite(the,HIGH);

    if(carattere=='R')//Carattere di reset che spegne tutti i led
    {
      digitalWrite(caffe,LOW);
      digitalWrite(macchiato,LOW);
      digitalWrite(the,LOW);
      digitalWrite(stick,LOW);
    }

  }
}
void inserita() //subroutine che viene richiamata dall'interrupt
{
  if((millis()-lastTime)>300)//debounce utilizzando il millis
  {
    moneta=true; lastTime=millis();
  }
}
```