Freccie da bicicletta

Per questo concorso, il mio primo, non avendo molta esperienza e conoscenze di programmazione, se non basi di C e quel poco di arduino, ho deciso di fare qualcosa di semplice. Questa mia decisione è stata influenzata dal fatto che se no non avrei avuto abbastanza tempo per scrivere e debuggare la tastiera USB estendibile che volevo fare. Sarà per la prossima volta. Ho deciso quindi di fare una cosa che può tornare utile nella vita quotidiana, come del resto dovrebbe essere tutta la tecnologia. Pertanto ecco a voi le freccie da bicicletta. Questo programmino diventa utile quando in bibiletta, per curvare abbiamo problemi o di equilibrio o di macchine che passano, per cui non possiamo allargare le braccia.

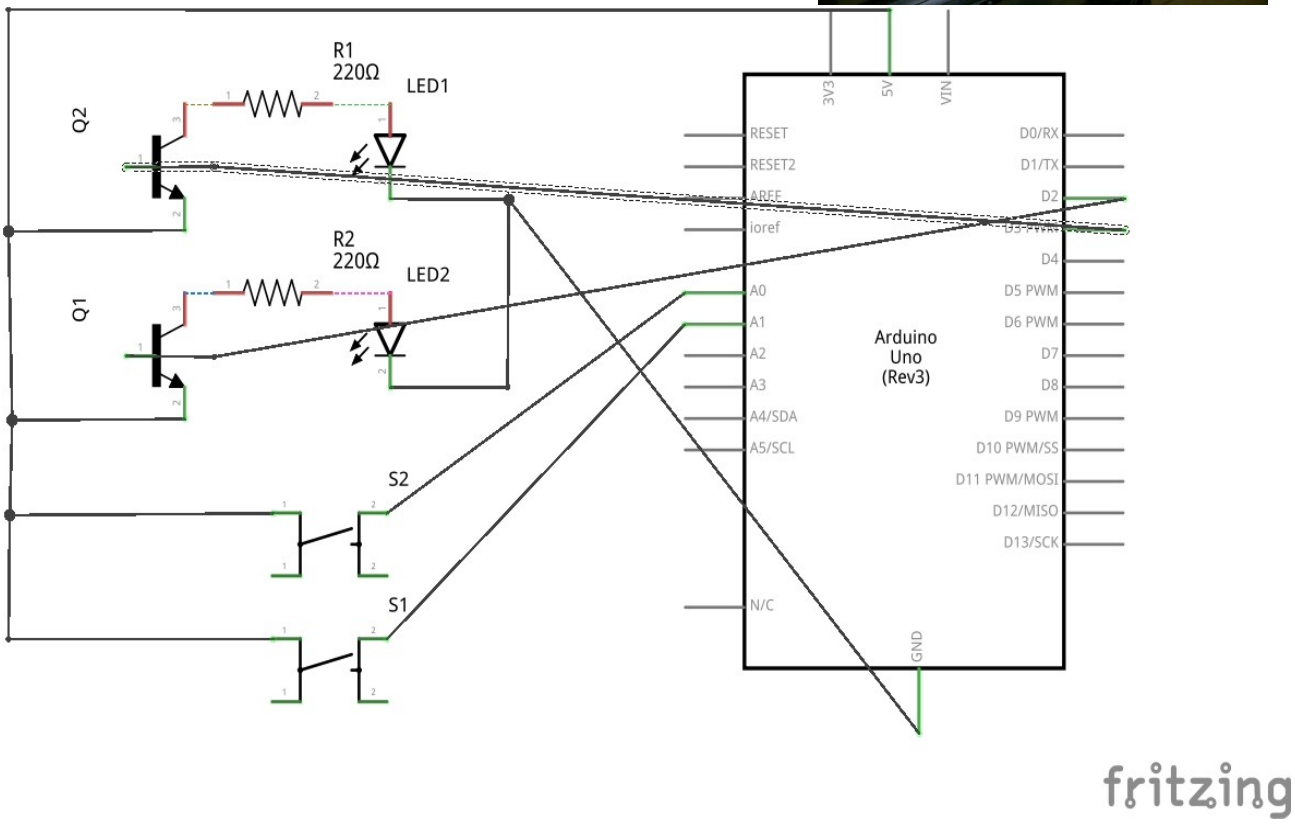
Materiali:

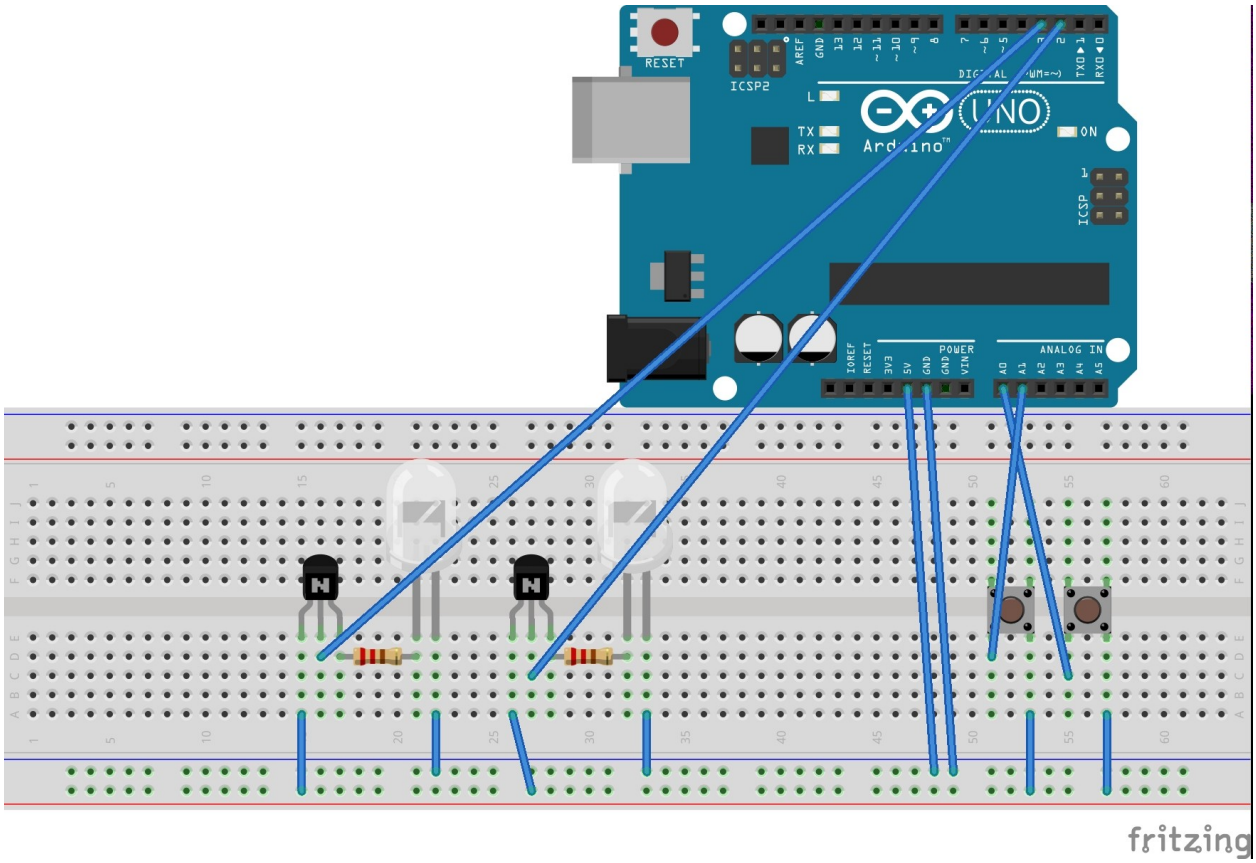
-arduino o attiny 25/45/85 con due ingressi e due uscite digitali, utile per le sue ridotte dimensioni
-led quanti ne volete usare, minimo due
-2xresistenze per i led, suggerisco resistenze da 220ohm
-2xnpn transistor, per non sovraccaricare la scheda con i led
-2xpulsanti
-cavetteria varia
-contenitore per il circuito, io ne ho uno attaccato al centro del manubrio
-altro in funzione da come vuoi montare i led e personalizzare

Opzionale:

-connettore per far passare le luci ad un eventuale carretto attaccato alla bicicletta, potendolo così staccare quando serve

Di seguito trovate lo schema di collegamento.





Il programma è ideato per arduino, ma consiglio di utilizzare un attiny 45/85 per le sue ridotte dimensioni. Il codice per arduino è il seguente:

```
const int ledR = 2; //i pin possono essere ovviamente cambiati a proprio piacimento,
const int ledL = 3; //in conformazione ai pin presenti sulla scheda
const int bR = A0; //io ho usato gli analogici perchè più affidabili, i digitali avevano problemi
const int bL = A1;
void setup(){
  pinMode(ledR, OUTPUT);
  pinMode(ledL, OUTPUT);
  pinMode(bR, INPUT);
  pinMode(bL, INPUT);
}
void loop(){
  if (analogRead(bR) >= 1000) {
    digitalWrite(ledR, HIGH);
    delay(333);
    digitalWrite(ledR, LOW);
    delay(333);
  }
  if (analogRead(bL) >= 1000) {
    digitalWrite(ledL, HIGH);
    delay(333);
    digitalWrite(ledL, LOW);
    delay(333);
  }
}
```

Per far funzionare il programma con un attiny 25/45/85, modificare le prime quattro linee relative alle variabili come segue:
const int ledR = 1; //pin 6 dell'attiny
const int ledL = 2; //pin 7
const int bR = A2; //pin 3
const int bL = A3; //pin 2
Ecco il pinout dell'attiny

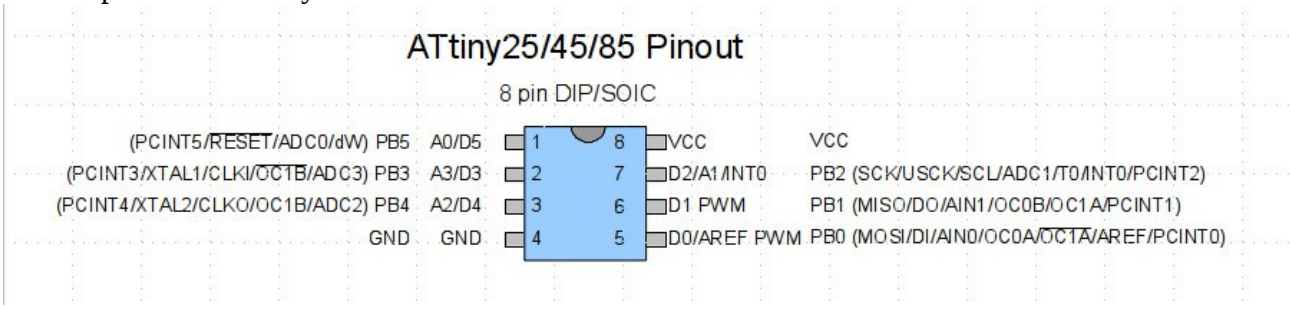
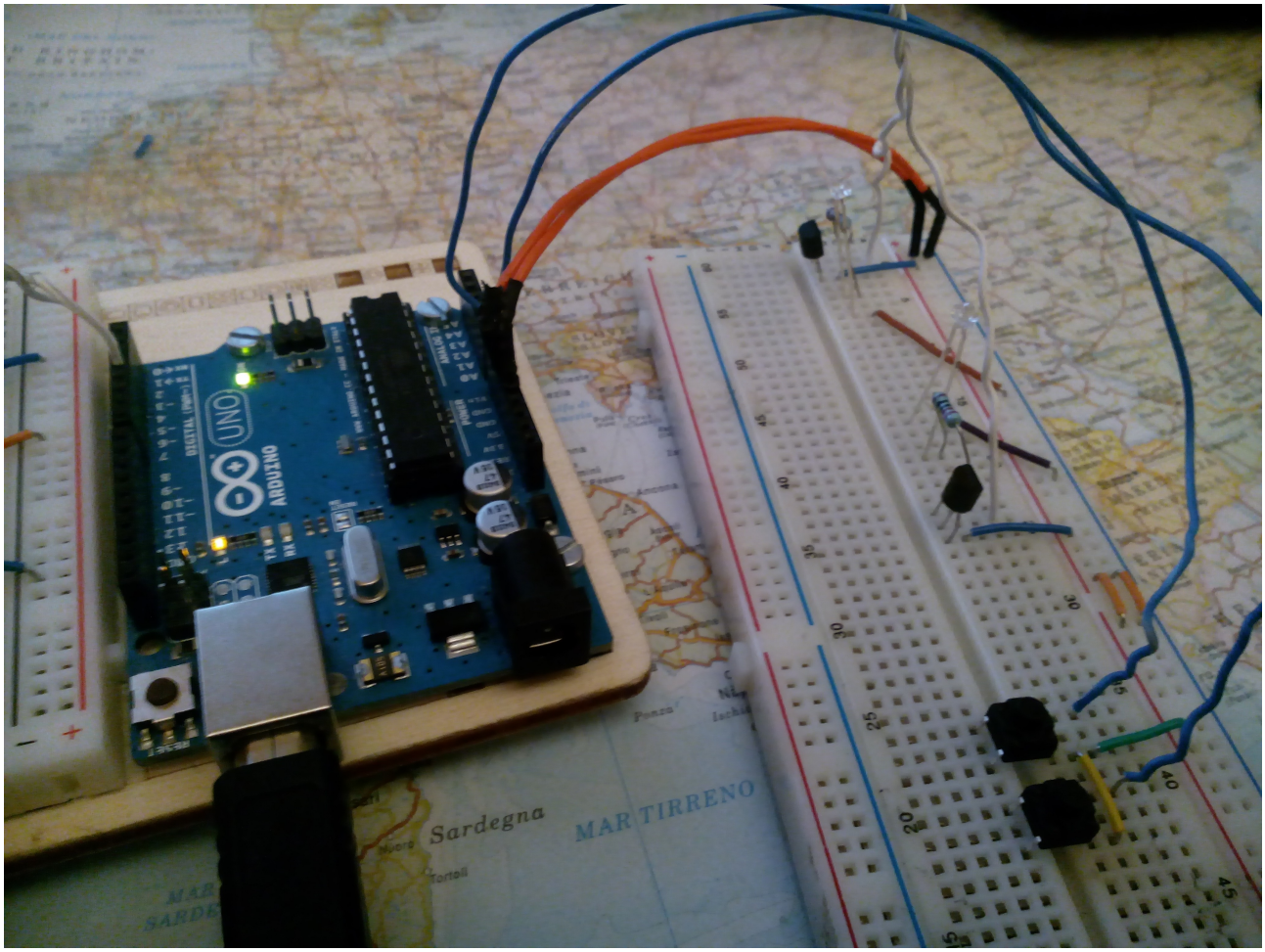


foto scaricata da provideyourown.com
Ed ecco il tutto montato su una breadborad



Autore: Filippo Falezza