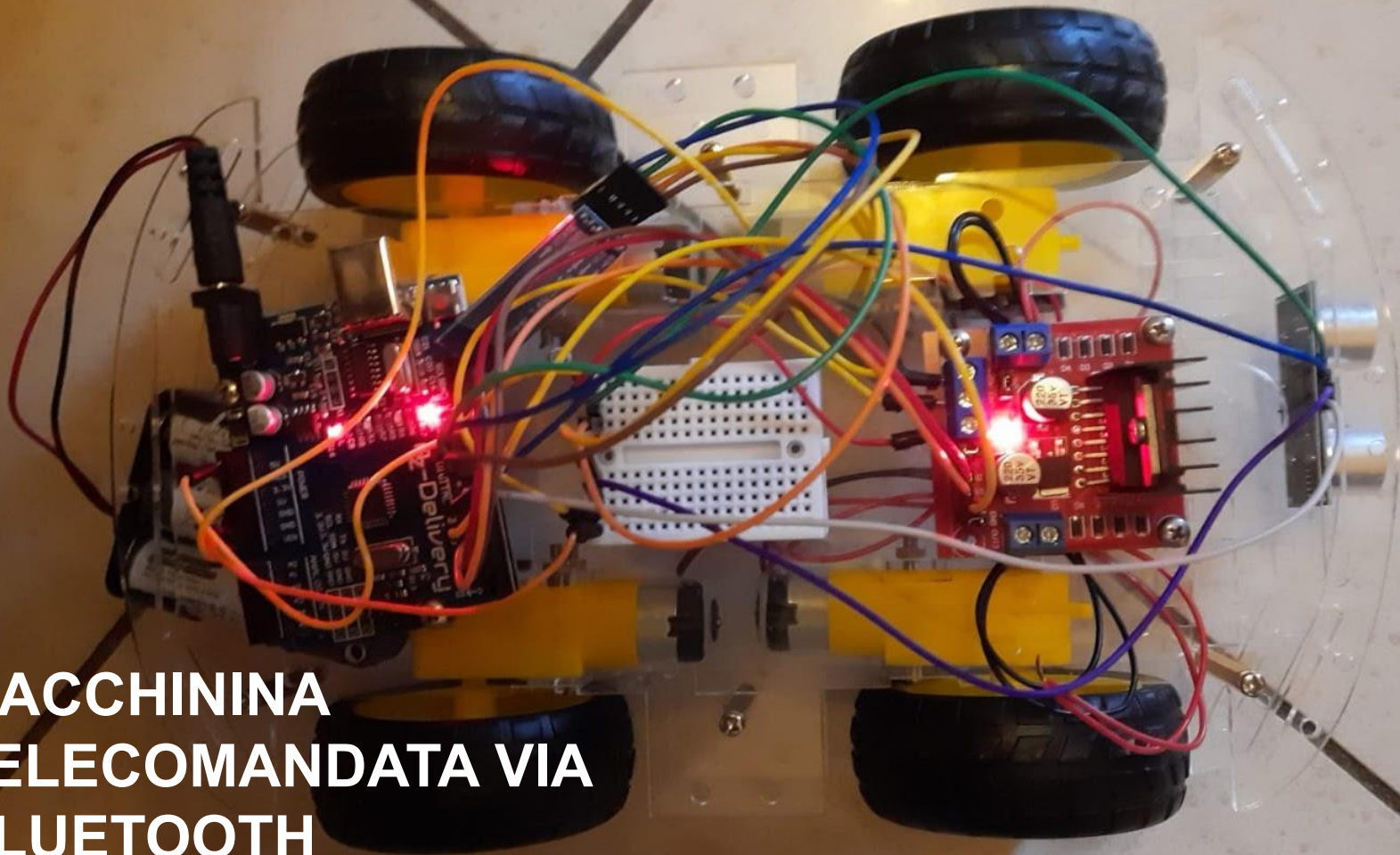


MACCHININA TELECOMANDA VIA BLUETOOTH



Assemblaggio macchinina

Abbiamo saldato i cavi ai motori che poi abbiamo posizionato e fissato sulla base inferiore.

Abbiamo poi fissato la parte superiore, posizionato l'arduino nella parte posteriore e il vano batterie per alimentarlo, nello spazio tra i motori e le due basi. Mentre la breadboard l'abbiamo posizionata al centro.

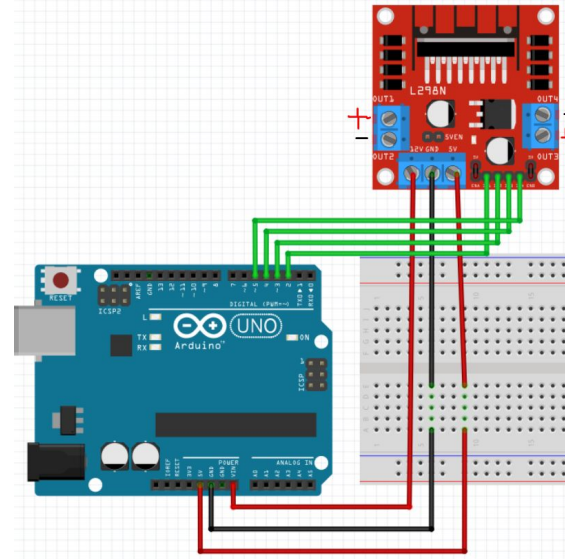


Driver L298N

E' un componente che permette di comandare motori a corrente continua che abbiamo posizionato nella parte anteriore. Alle porte OUT abbiamo collegato i fili dei motori che avevamo già accoppiato per colore distinguendo quelli di destra da quelli di sinistra.

Poi abbiamo lo abbiamo alimentato tramite il collegamento tra Vin e 12V, 5V e 5V poi abbiamo collegato il GND del driver a quello dell'arduino.

Abbiamo collegato poi i piedini dal IN1 all'IN4 alle porte dell'arduino da 2 a 5 che ci serviranno per comandare i motori.

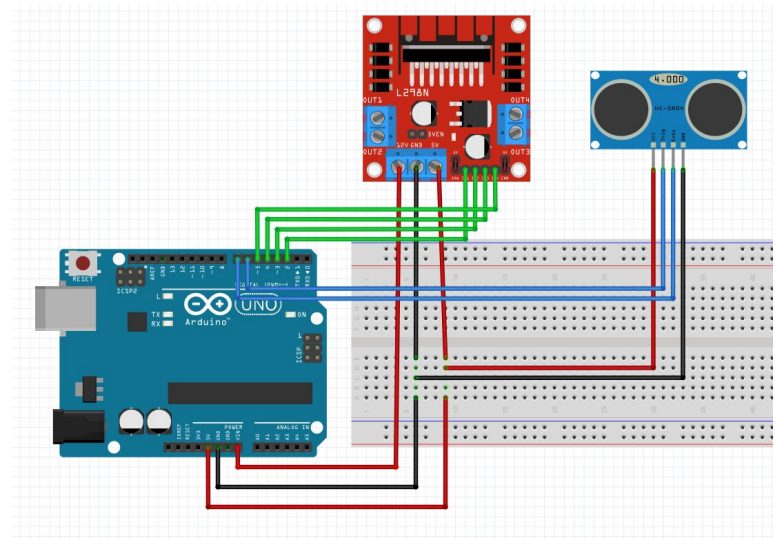
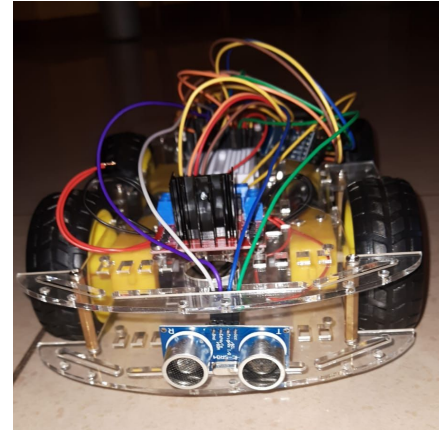


Sensore ad ultrasuoni

Per rilevare gli ostacoli abbiamo inserito un sensore ad ultrasuoni che abbiamo incastrato nella parte frontale della macchinina tra le due basi.

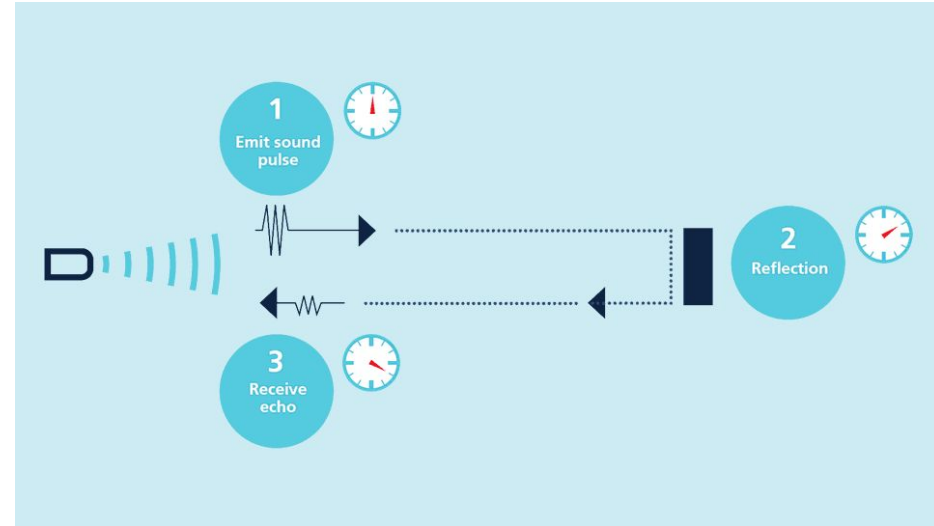
Abbiamo quindi collegato Vcc e GND in parallelo ai 5V e al GND del Driver. Infine Trig al pin 6 e echo al pin 7 dell'Arduino.

Codice
ultrasuoni



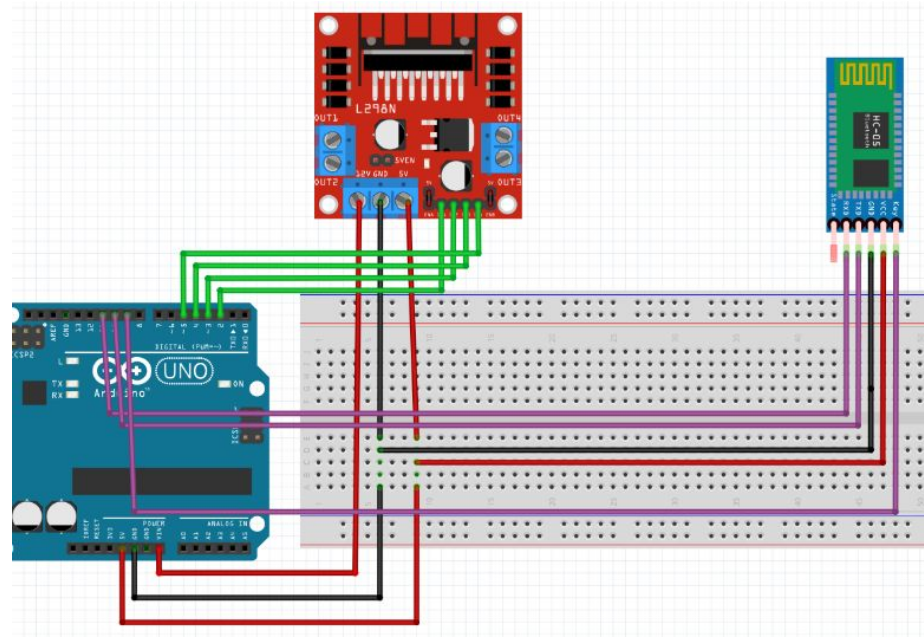
Funzionamento sensore ultrasuoni

Il sensore ultrasuoni emette ciclicamente un breve impulso acustico di alta frequenza. Quest'ultimo si propaga nell'aria alla velocità del suono. Quando incontra un oggetto, esso viene riflesso e raggiunge il sensore ultrasonico in forma di eco. Il sensore a ultrasuoni calcola quindi internamente la distanza dell'oggetto riflettente in base al margine di tempo compreso tra la trasmissione dell'impulso acustico e la ricezione del segnale d'eco.



Modulo Bluetooth

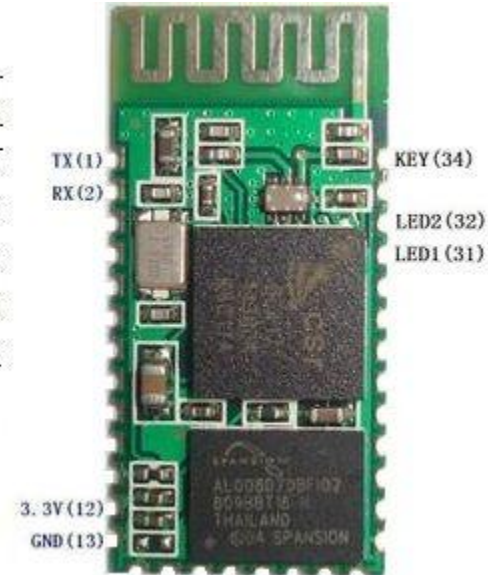
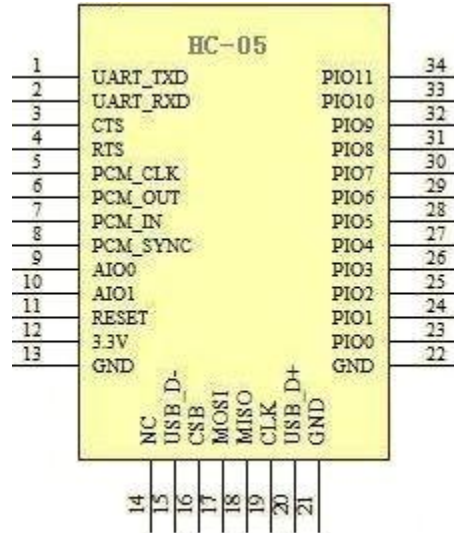
Abbiamo inserito un modulo bluetooth che ci permette di far muovere la macchinina tramite un'app. Per farlo abbiamo collegato RDX e pin 11, TXD con il pin 10 e infine Key con il pin 9. Poi abbiamo collegato, similmente a come avevamo fatto prima per il sensore ad ultrasuoni, il GND e Vcc.



Codice

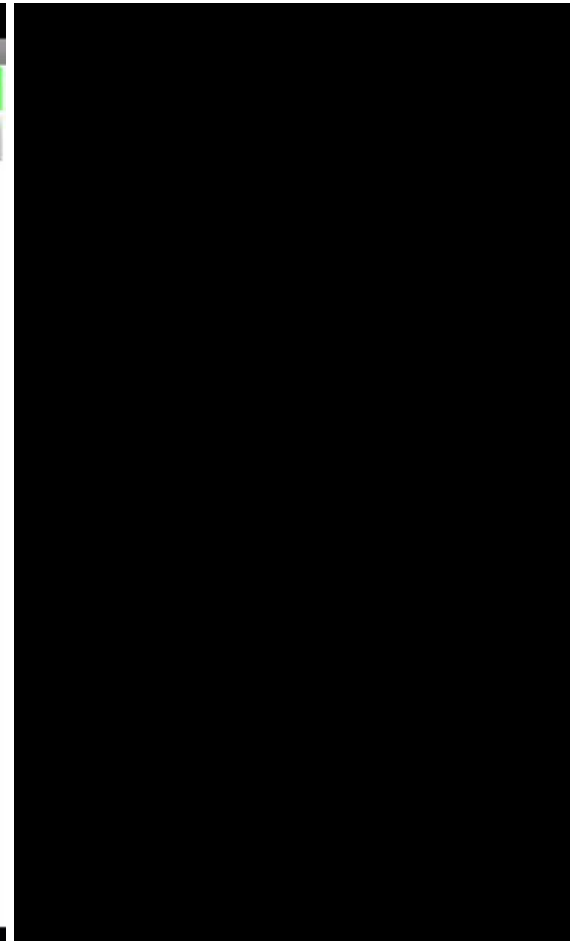
Funzionamento Modulo Bluetooth

Il Bluetooth® sfrutta onde radio, precisamente sulla frequenza che oscilla tra i 2.4 ed i 2.485 GHz, tramite la presenza di un chip installato sul tuo dispositivo. Questo ti consente la creazione di una piccola rete, le cui dimensioni possono variare dai 10 ai 30 m in termini di raggio d'azione della stessa. Quest'area si chiama PAN, acronimo di Personal Area Network, e ti permette di connettere il tuo dispositivo, e quindi scambiare dati, con uno o più dispositivi dotati della presenza di un chip affine. Il passaggio dei file dal tuo apparecchio all'altro avviene in totale sicurezza e l'impiego di energia è a oggi estremamente ridotto, grazie ai miglioramenti apportati.



App

Grazie a MIT app inventor possiamo creare la nostra app. In cui possiamo verificare se è collegato il telefono alla macchinina, e nel caso non lo fosse collegarci, spostarci avanti e indietro oppure a destra, sinistra e fermarci.



Codice a blocchi APP

The screenshot displays the MIT App Inventor web interface. At the top, the MIT App Inventor logo is visible, along with navigation links: Progetti, Connetti, Compila, Settings, and Aiuto. The project name 'macchinina' is shown in the top bar, with options to 'Pubblica nella Galleria', 'Screen1', 'Aggiungi Schermo...', and 'Rimuovi Schermo'.

The left sidebar contains a 'Blocchi' (Blocks) section with categories: Incorporato (Controllo, Logica, Matematica, Testo, Liste, Dictionaries, Colori, Variabili, Procedure), Screen1, and Multimediale (Carica File...). The 'Screen1' section is expanded, showing a vertical stack of components: DisposizioneVerticale1, SelettoreLista1, Pulsante1, DisposizioneOrizzzo, CasellaDiTesto1, and avanti.

The main area is the 'Visualizzatore' (Visualizer) showing the code blocks for the application. The code is organized into several 'per sempre quando' (when green flag clicked) blocks:

- Block 1:** When 'SelettoreLista1' is clicked, execute 'imposta SelettoreLista1 Elementi a ClientBluetooth1 IndirizziNomi'.
- Block 2:** When 'SelettoreLista1' is clicked, execute 'imposta SelettoreLista1 Selezione a', 'imposta SelettoreLista1 ColoreSfondo a', and 'imposta SelettoreLista1 Testo a CONNESSO'.
- Block 3:** When 'Pulsante1' is clicked, execute 'esegui ClientBluetooth1 Disconnetti', 'imposta SelettoreLista1 ColoreSfondo a', and 'imposta SelettoreLista1 Testo a NON CONNESSO'.
- Block 4:** When 'avanti' is clicked, execute 'esegui ClientBluetooth1 InviaTesto' and 'testo'.
- Block 5:** When 'dx' is clicked, execute 'esegui ClientBluetooth1 InviaTesto' and 'testo'.
- Block 6:** When 'sx' is clicked, execute 'esegui ClientBluetooth1 InviaTesto' and 'testo'.
- Block 7:** When 'dietro' is clicked, execute 'esegui ClientBluetooth1 InviaTesto' and 'testo'.
- Block 8:** When 'stop' is clicked, execute 'esegui ClientBluetooth1 InviaTesto' and 'testo'.

At the bottom left, there are two warning icons (yellow triangle and red X) and a button labeled 'Mostra Avvertimenti'.

Guanto

(Non l'abbiamo realizzato perché il costo sarebbe stato troppo alto).

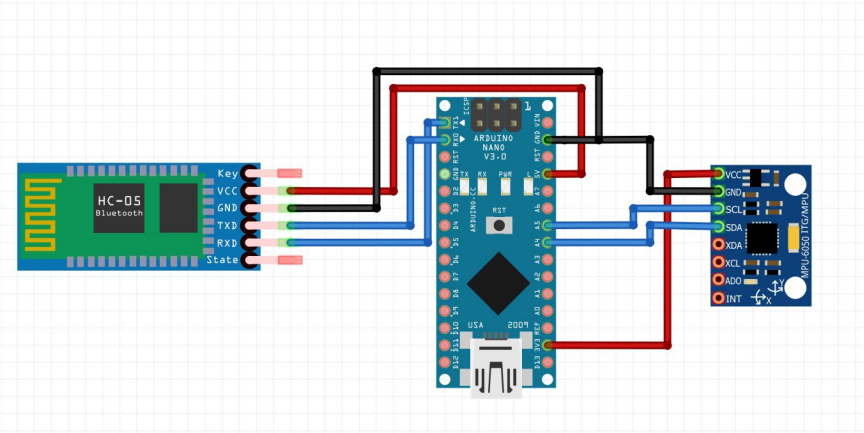
Fissiamo su un guanto un arduino nano, un accelerometro e un modulo bluetooth.

Sulla macchinina dovremmo rimuovere Key e invece collegare TXD e RXD alle porte 1 e 0.

Mentre sul guanto colleghiamo il modulo bluetooth a TXD e RXD alle rispettive porte sull'arduino nano, Vcc a 5V e il GND alla suo pin.

Per l'accelerometro colleghiamo VCC a EVE, che sarebbe 3.3V dell'arduino, SCL ad A5 e SDA a A4.

Codice Guanto



Volevamo calcolare lo spostamento lungo gli assi e controllare entro quale range di valori fosse ogni movimento della mano, in base a quello veniva passata all'arduino una lettera che poi tramite lo switch faceva muovere in maniera opportuna la macchinina.