

DOCUMENTAZIONE TECNICA PROGETTO TANGO





Progetto Industriale: Tango

Documentazione tecnica del progetto industriale del corso 12.1 di ITS - Accademia Digitale Liguria, svolto in collaborazione con Scuola di Robotica.

Introduzione

Tango è un robot in grado di muoversi, registrare dati dall'ambiente circostante e trasportare carichi. È controllato da remoto tramite una mobile app sviluppata dal nostro team. Questo documento descrive lo scopo, gli obiettivi e le finalità del software di gestione, analizza il codice, fornisce una panoramica riguardo la progettazione dei componenti fisici in ambiente Blender (con l'ausilio di Shapr3D e XTools), descrive il montaggio dei componenti a bordo macchina e offre tavole illustrate sullo schema elettrico e i collegamenti hardware, fornendo anche un manuale d'uso.

Obiettivo del progetto

L'obiettivo del progetto Tango è l'assemblaggio di un robot e del software necessario per gestirlo. Il robot viene controllato tramite una applicazione mobile sviluppata appositamente.

Background

Il corso 12.1 di ITS - Accademia Digitale Liguria deve portare a termine, secondo curriculum, la progettazione e realizzazione di una macchina IoT nello spazio di 84 ore. Abbiamo deciso, insieme al team di Scuola di Robotica, di procedere alla personalizzazione del robot iRobot ATRV jr, dotandolo di sensori ambientali e di un software di controllo remoto.

Dopo una prima analisi, sono stati identificati quattro obiettivi principali:

- Controllo del robot: per poter portare a termine i propri compiti, il robot necessita di un'unità di controllo opportunamente programmata.
- Registrazione dei dati dall'ambiente circostante: attraverso i sensori montati a bordo macchina, il robot deve essere in grado di registrare dati dall'ambiente circostante.
- Movimento del robot: il robot deve essere in grado di muoversi, controllato da remoto, in ogni direzione senza limiti.
- Il robot deve essere in grado di fermarsi immediatamente in caso di emergenza.

Panoramica del documento

Indice

Requisiti

Overview delle funzioni

- Caratteristiche delle funzioni

Overview: stampa e grafica

Overview: hardware

- Tavole illustrate dello schema elettrico

Testing del codice

Appendice - Manuale di utilizzo

Requisiti

Specifiche tecniche

- Robot ATRV-JR
- Arduino Mega 2560 (3 schede)
- App per dispositivi mobile sviluppata in Java

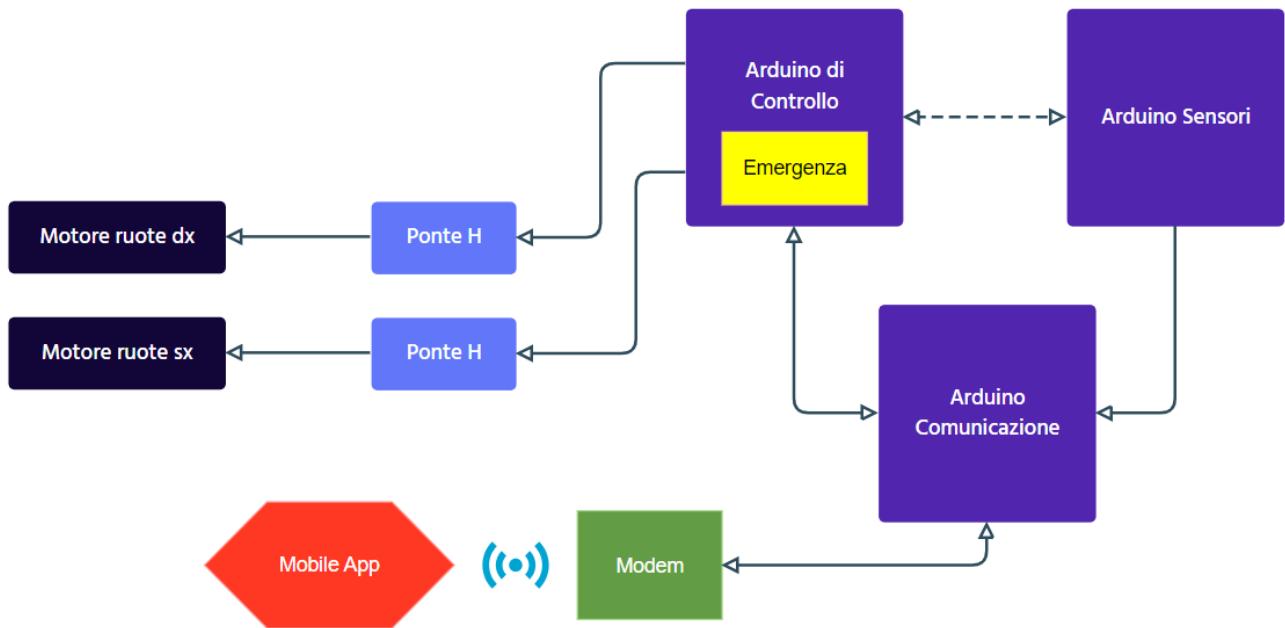
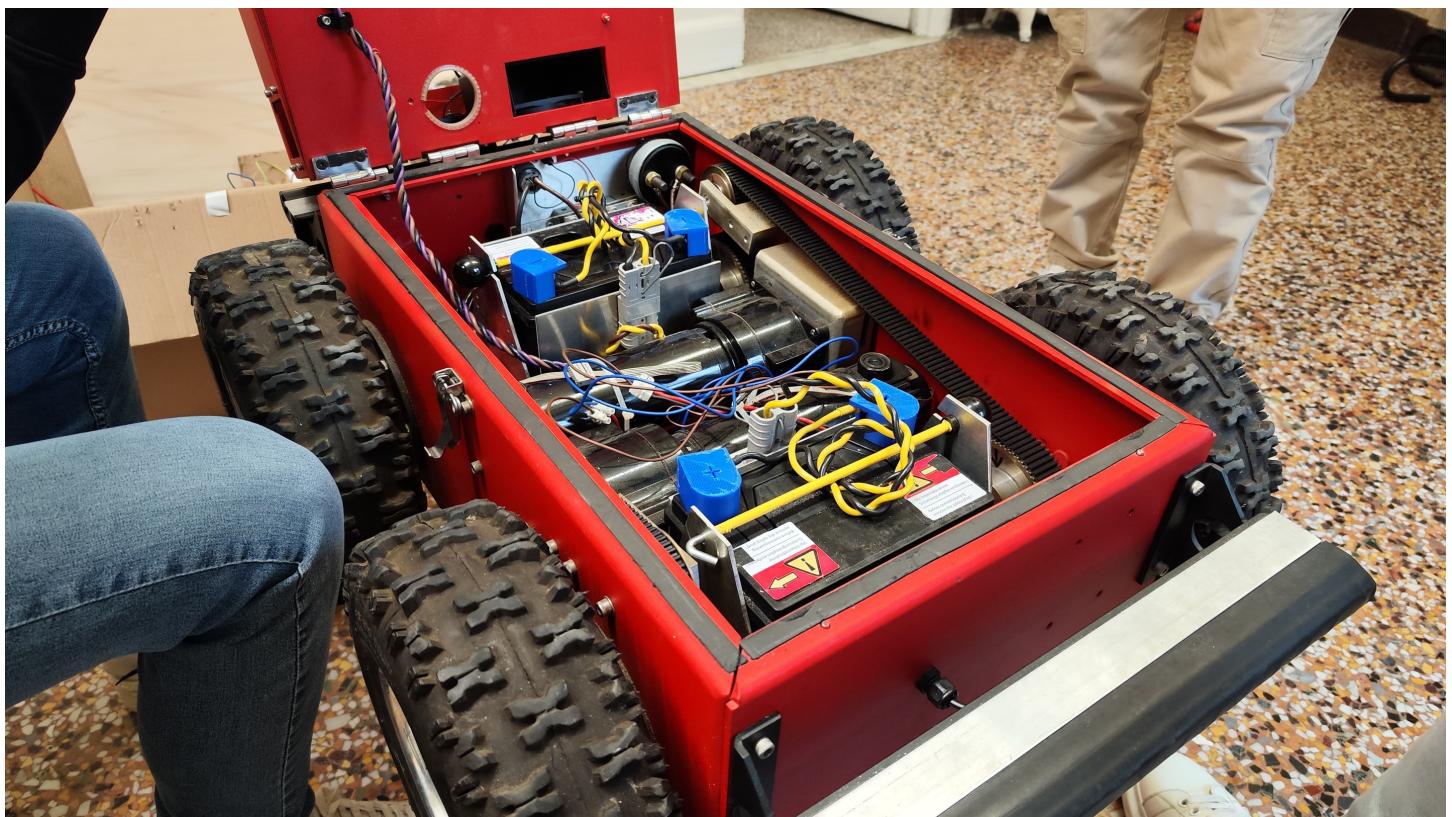


Fig. 1 - schema a blocchi delle funzionalità principali del robot



Fig. 2 - 3 - vista laterale (immagine sopra) e dall'alto (immagine sotto) del robot



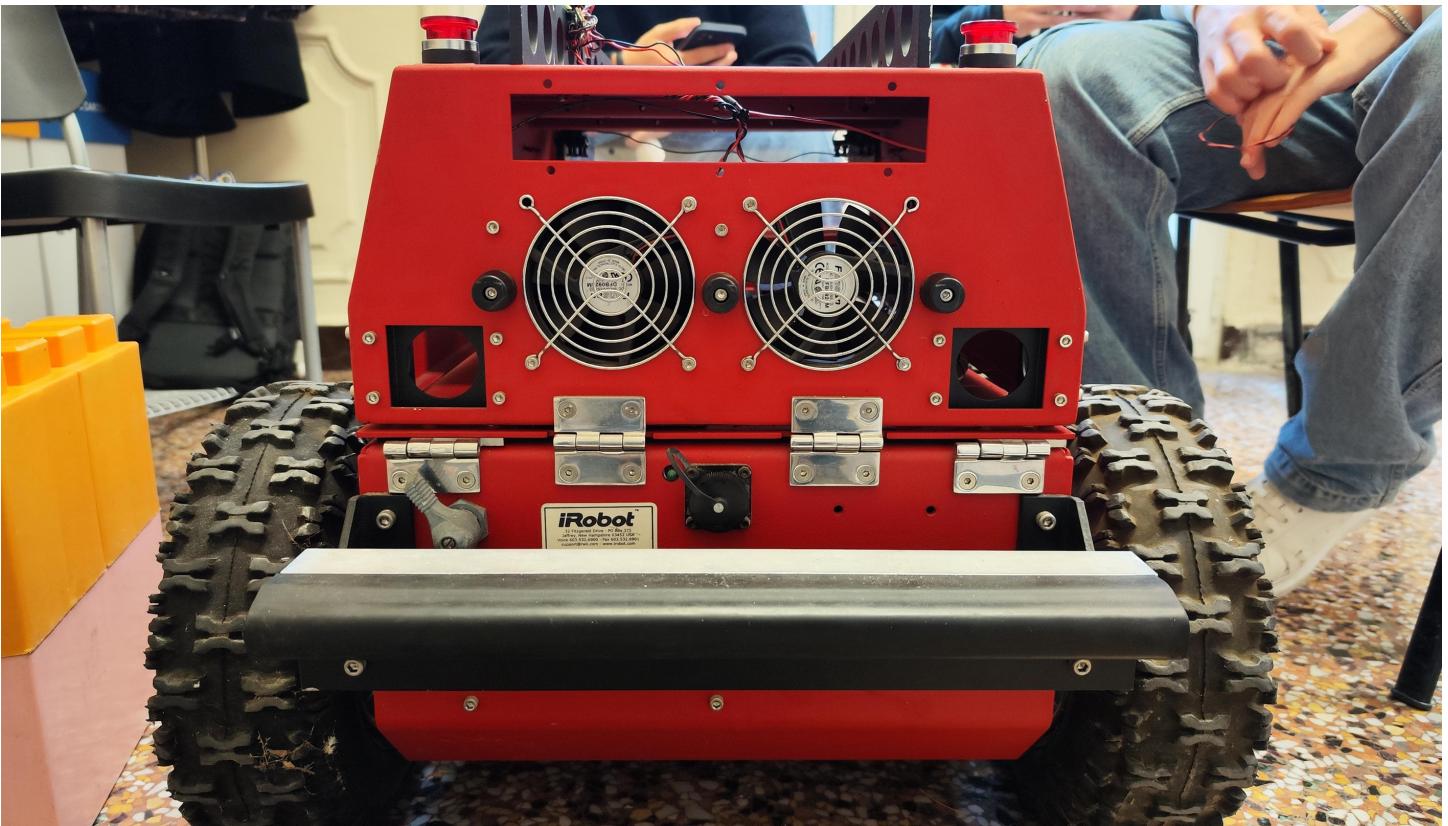


Fig. 4 - vista posteriore del robot

Overview delle funzioni

Il software svolge tre macro-funzioni:

- comunicazione;
- controllo;
- gestione dei sensori.

Per ognuna viene fornito anche un link per consultare il codice sorgente del software presente su GitHub.

Comunicazione

Il sistema di comunicazione bidirezionale collega l'applicazione Android e il sistema di controllo del robot, che è basato su Arduino MEGA, e invierà tutti gli input ricevuti dall'app Android a un altro Arduino, che controlla i due motori del robot.

Applicazione Android

L'interfaccia utente per il controllo di Tango è un'app mobile sviluppata utilizzando Android Studio di JetBrains e Google. Offre un'interfaccia intuitiva per gli utenti finali, consentendo loro di inviare comandi di controllo al robot tramite input touchscreen. Tutti gli input generati dall'applicazione vengono inviati a un server Node.js tramite richieste HTTP POST.

Arduino MEGA

Arduino MEGA è il componente centrale del sistema di controllo del robot. È stato equipaggiato con uno shield Ethernet per consentire la comunicazione con il server Node.js ed eventualmente altri dispositivi di rete. Una parte di codice è stata sviluppata per configurare Arduino MEGA come un server utilizzando la libreria Ethernet.h.

Flusso di comunicazione

Il flusso di comunicazione inizia con l'utente che interagisce con l'applicazione Android tramite input touchscreen. Gli input vengono quindi inviati al server Node.js tramite richieste HTTP POST. Il server Node.js riceve gli input, li elabora e li inoltra alla board Arduino MEGA tramite la connessione Ethernet. Infine, Arduino MEGA interpreta i comandi ricevuti e attua le azioni corrispondenti per controllare i motori del robot.

[Link GitHub](#)

Controllo

Il sistema di controllo è basato su una scheda Arduino MEGA che gestisce la comunicazione e il controllo del veicolo, garantendo la sicurezza e la coerenza del movimento in base ai comandi ricevuti e allo stato dei sensori.

In aggiunta, alla scheda Arduino MEGA di controllo è demandato anche il compito di gestire il sensore a ultrasuoni per la rilevazione degli ostacoli.

Scheda Arduino - Controllo e Comunicazione

Si tratta del firmware per il controllo di Tango tramite un Arduino MEGA, che permette la gestione della comunicazione con altri dispositivi e sensori. Di

seguito, una breve panoramica delle sue funzionalità:

Configurazione e inizializzazione: in questa sezione vengono dichiarate le variabili, configurati i pin di input/output e inizializzate le comunicazioni seriali con altri dispositivi.

Loop principale: il programma esegue ciclicamente un loop in cui controlla la comunicazione seriale, mappa i messaggi ricevuti, gestisce lo stato di emergenza e il movimento del veicolo.

Gestione dell'emergenza: viene definita una funzione per gestire lo stato di emergenza del veicolo, che lo ferma in caso di emergenza e invia segnali agli altri dispositivi.

Comunicazione seriale: una funzione per gestire la comunicazione seriale, sia in entrata che in uscita, con altri dispositivi e sensori.

Gestione della distanza: utilizzando sensori ad ultrasuoni, viene misurata la distanza dagli oggetti circostanti e si gestiscono situazioni di emergenza se la distanza è inferiore a una certa soglia.

Gestione del movimento: la funzione movement() gestisce il movimento del veicolo in base ai comandi ricevuti. Utilizza un'interruzione per rilevare lo stato di emergenza e reagisce di conseguenza.

Controllo della velocità: ci sono funzioni per controllare la velocità del veicolo e garantire che rimanga entro limiti sicuri.

Gestione della Rotazione: viene controllata la rotazione del veicolo e vengono prese misure per evitare movimenti indesiderati o pericolosi.

Funzioni Ausiliarie: le funzioni ausiliarie gestiscono operazioni specifiche, come l'arresto dei motori o la decelerazione graduale.

[Link GitHub](#)

Sensori

Scheda Arduino - Sensori

Sulla scheda dedicata è stato caricato un software che si occupa di misurare la temperatura e l'umidità ambientale con un sensore DHT11 e di monitorare lo stato della batteria tramite un partitore DFR0051.

Le informazioni raccolte vengono quindi trasmesse tramite la comunicazione seriale per essere lette da un dispositivo esterno, come ad esempio un computer, e possono anche essere visualizzate su un display LCD. Nel nostro caso, verranno poi visualizzate tramite l'app dedicata.

Di seguito, una breve panoramica delle sue funzionalità.

Gestione di temperatura e umidità: Viene utilizzato un sensore DHT11 per misurare la temperatura e l'umidità ambiente e questi fornisce le informazioni tramite la comunicazione seriale.

Monitoraggio dello stato della batteria: Lo stato della batteria viene monitorato misurando la tensione tramite un partitore DFR0051. Il dato viene trasmesso tramite la comunicazione seriale.

Loop principale: Nel loop principale del programma, vengono gestite le operazioni di monitoraggio della distanza, aggiornamento del display LCD e monitoraggio periodico della temperatura, umidità e tensione della batteria.

Gestione del tempo: Vengono eseguite determinate azioni ogni tot minuti utilizzando il modulo millis() per calcolare il tempo trascorso.

[Link GitHub](#)

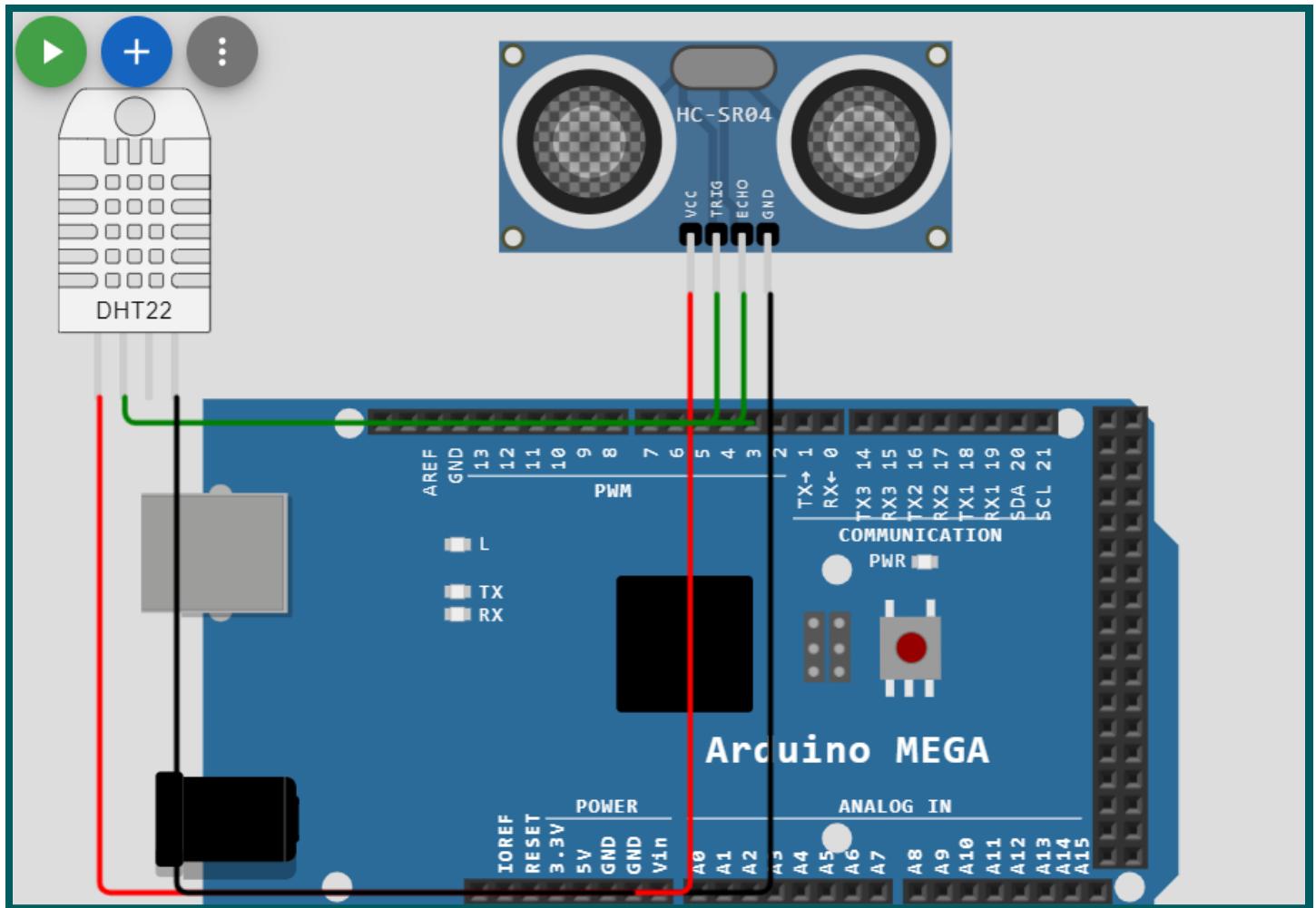


Fig. 5 - simulazione di collegamento sensori su board Arduino MEGA

Sviluppo - STAMPA E GRAFICA

1. Creazione di render per la stampa 3D e con lasercut di componenti
2. Uso di programmi come Xtool Creative Space (XCS) e Shapr3D
3. Montaggio di video di presentazione finale del progetto e sviluppo della brand identity

Sviluppo - HARDWARE

1. Montaggio dei componenti hardware a bordo macchina
2. Design dello schema elettrico

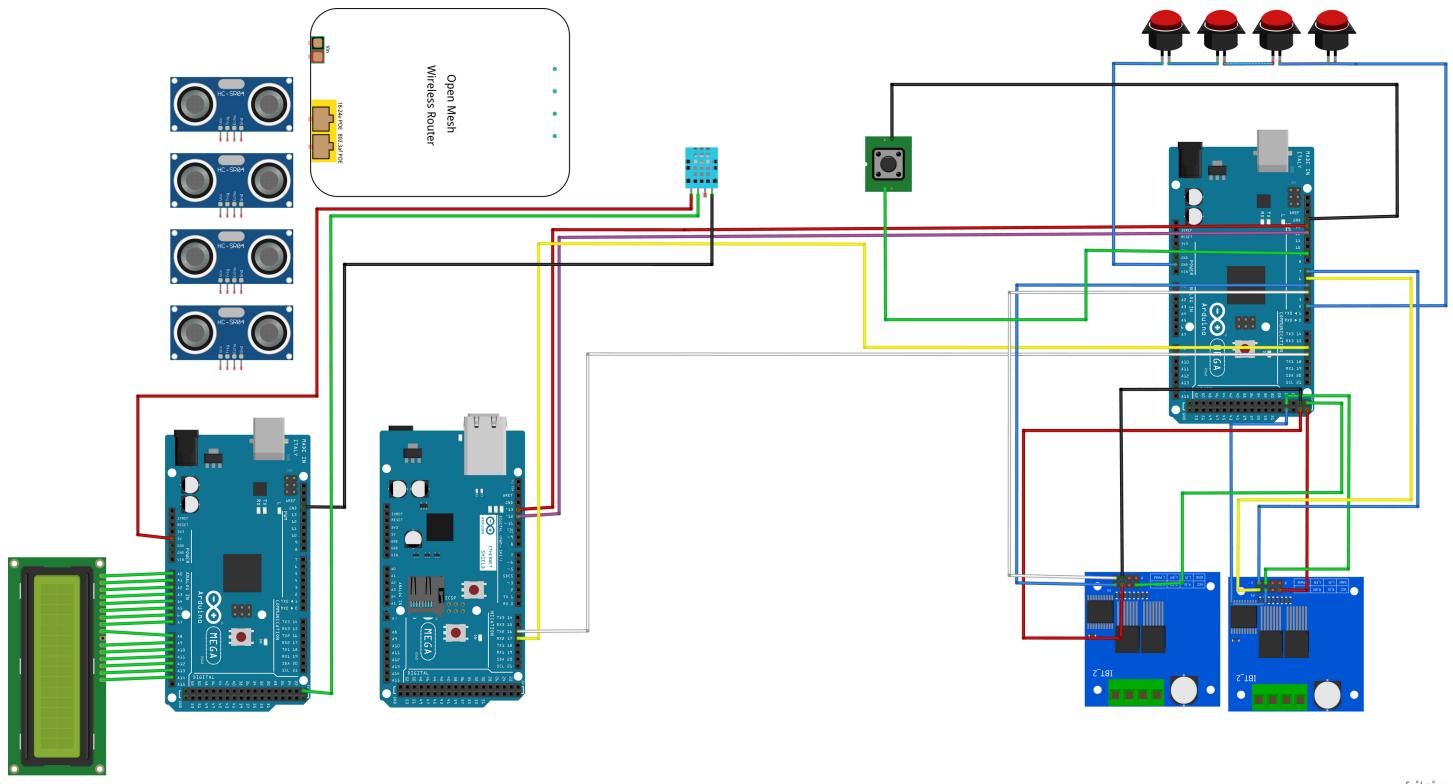


Fig. 6 - simulazione di collegamento dei componenti

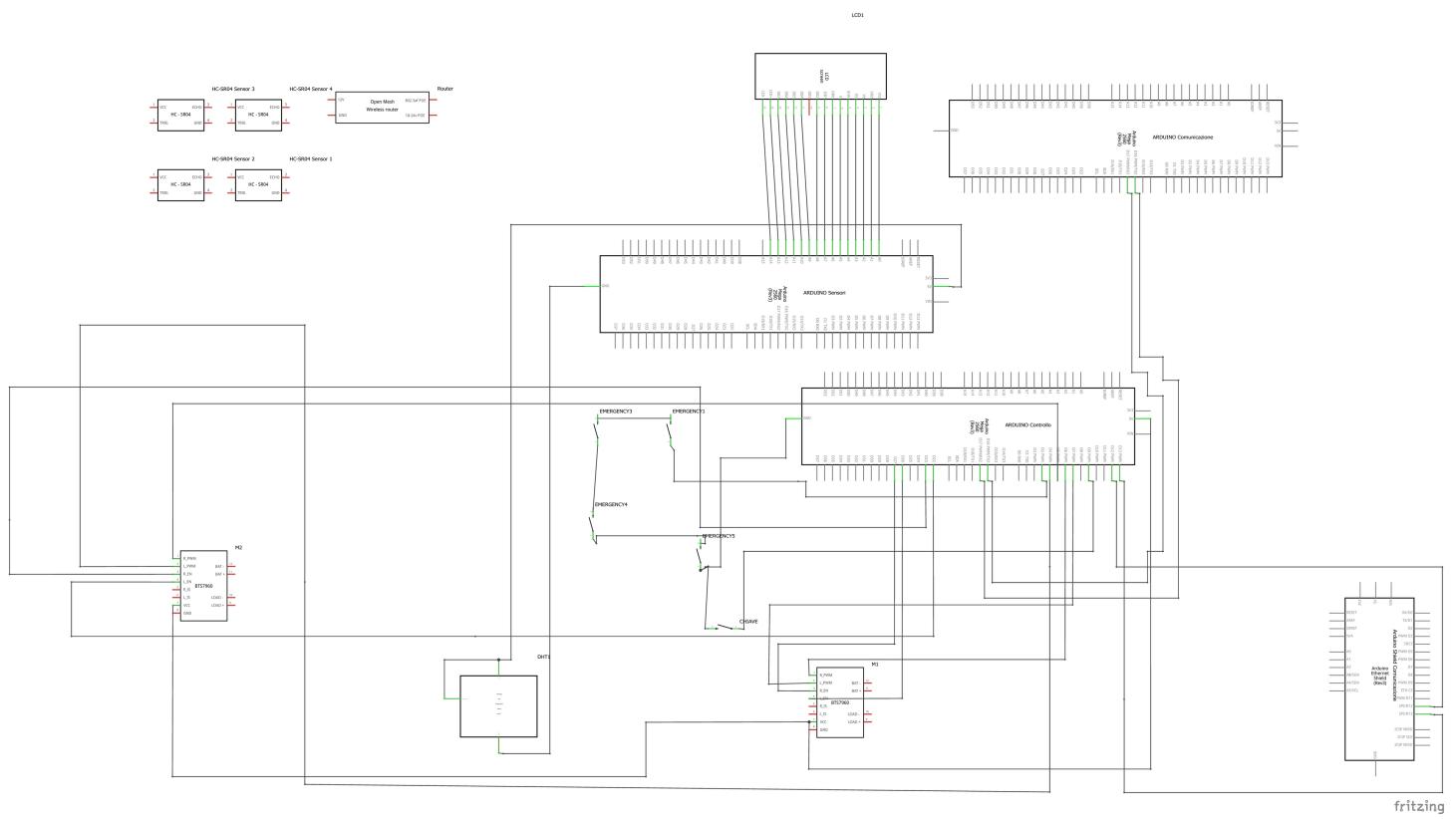


Fig. 7 - tavola illustrativa dello schema elettrico

TESTING

1. Modalità di testing del software

Durante la scrittura del codice, le funzioni sono state testate singolarmente allo scopo di individuare errori prima dell'accorpamento nel listato finale, es. Test di comunicazione per verificare che le schede Arduino del robot comunichino tra loro e con l'app -.

Quando ogni funzione è stata testata, si è potuto verificare e correggere eventuali errori in modo efficiente e preciso. Questa metologia ha permesso di garantire che il codice finale fosse il più possibile privo di difetti, migliorando la qualità complessiva del software sviluppato.

2. Modalità di testing dell'hardware

Dopo aver simulato e disegnato lo schema elettrico, sono stati montati i componenti a bordo macchina seguendo le indicazioni del team di sviluppo del software.

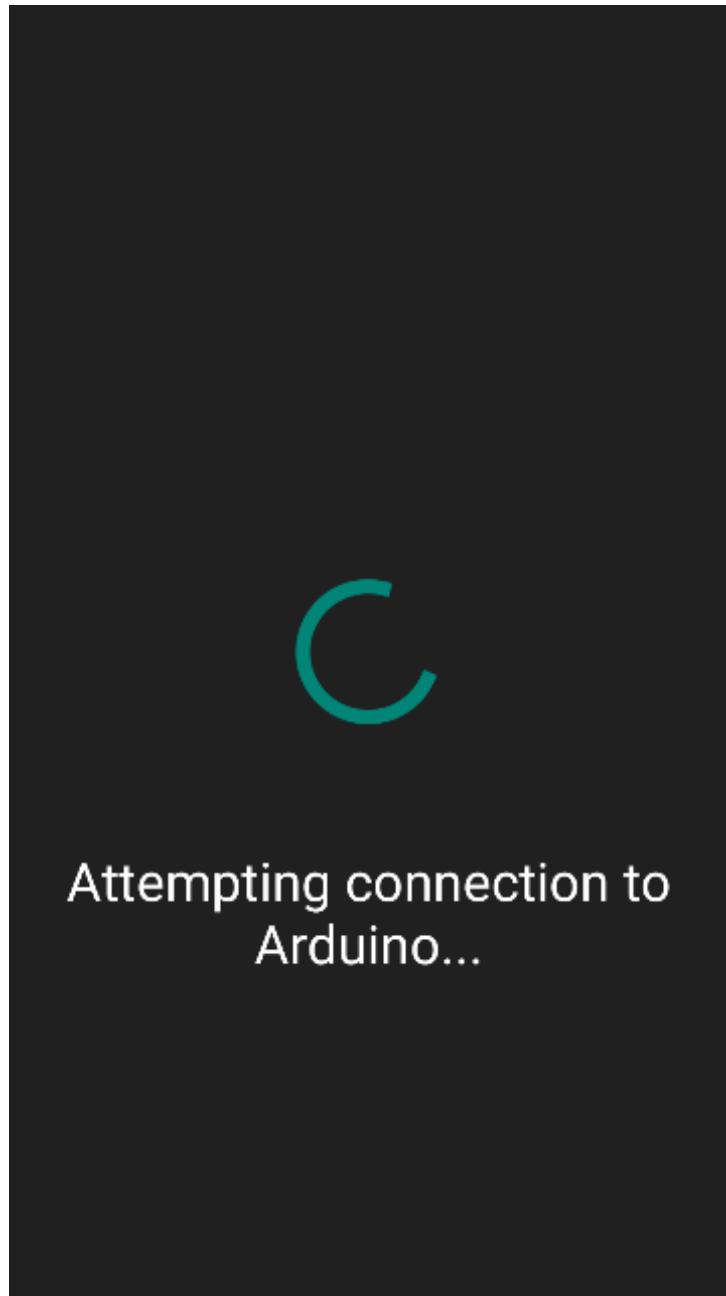
I test sono stati effettuati verificando il corretto funzionamento dei componenti - accensione, movimento delle ruote e spostamento del robot, tensione e perdite delle stesse dalle batterie - ed eventualmente sostituendo parti rovinate e/o non funzionanti. Una volta caricato il software a bordo macchina, si sono effettuati i test definitivi di funzionamento, movimento e registrazione dei dati ambientali in sinergia con il software.

APPENDICE

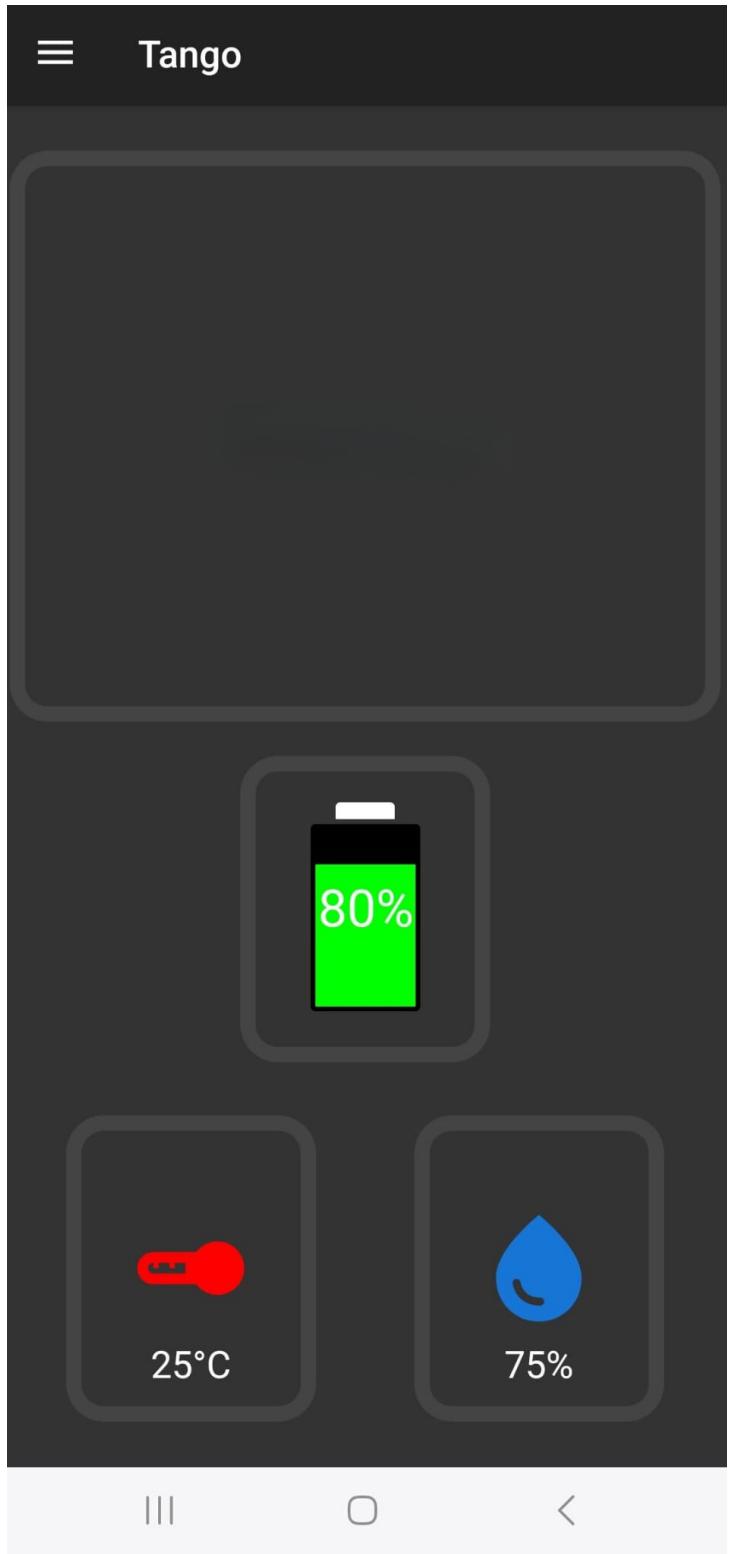
MANUALE DI UTILIZZO

1. AVVIO

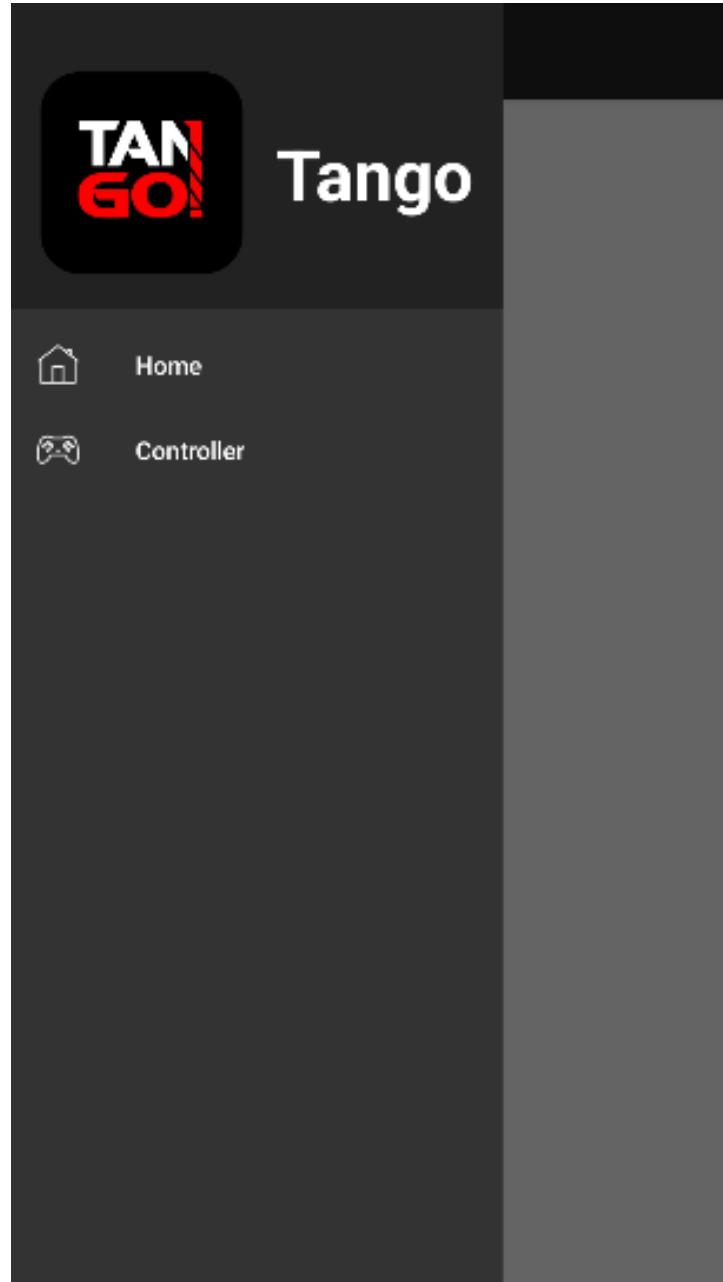
Quando l'app viene lanciata si apre una pagina di loading nella quale tenta di connettersi agli Arduino. Senza un collegamento riuscito, non è possibile accedere all'interfaccia principale.



Una volta connessi, sulla schermo viene visualizzata la schermata di home contenente le statistiche del robot ovvero batteria, temperatura e umidità.

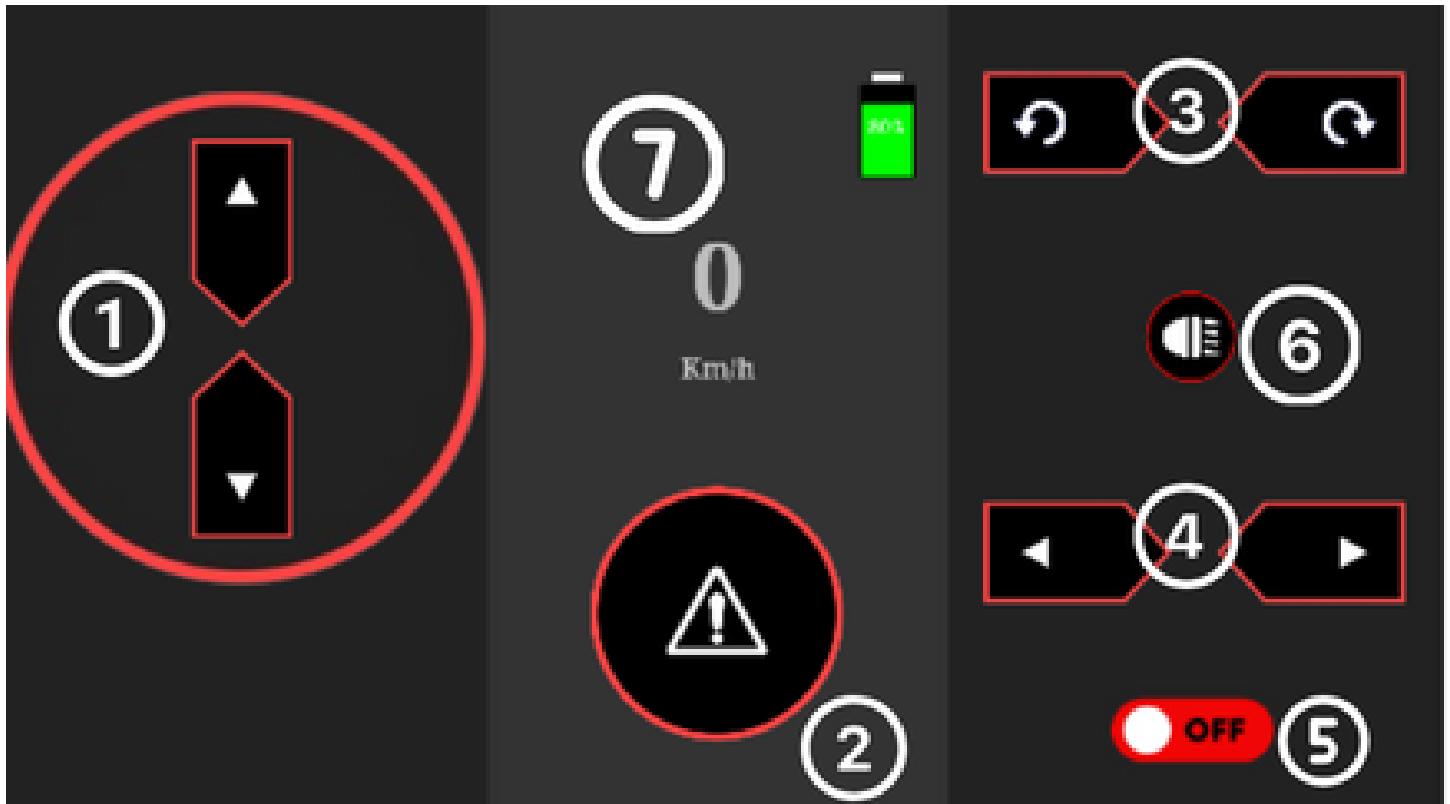


In alto a sinistra si accede a un menu a tendina col quale è possibile scegliere di cambiare schermata tra home e controller.



3. SCHERMATA PRINCIPALE - CONTROLLER

Una volta scelta la schermata del controller, è possibile sfruttare i comandi per far muovere il robot nell'ambiente circostante.



Di seguito, una panoramica delle funzioni indicate dai numeri:

1. TASTI DI MOVIMENTO AVANTI - INDIETRO

Permettono il movimento del robot avanti e indietro

2. TASTO DI EMERGENZA

Il tasto di emergenza implementa le funzioni di emergenza del codice: quando viene premuto, invia un segnale di STOP generale. Questo pulsante sposta lo switch presente al punto 5 su OFF.

3. TASTI DI ROTAZIONE

Premendo questi tasti, è possibile per il robot roteare sul posto verso destra o verso sinistra.

4. TASTI DI MOVIMENTO DESTRA - SINISTRA

Premendo questi tasti insieme ai tasti avanti o indietro, è possibile effettuare una curva dolce per sterzare.

5. SWITCH ON-OFF

Switch di accensione del robot, se su OFF non è possibile interagire né premere pulsanti.

6. LUCI

Pulsante che permette l'accensione delle luci del robot per il movimento in situazioni di scarsa luminosità.

7. TACHIMETRO E BATTERIA

Questa parte permette di tenere sotto controllo la velocità del robot e la batteria disponibile.