Indice ed ordine degli argomenti

Introduzione

La realizzazione di robot è la mia passione da tempo, nel 2018, andando al laboratorio, ho iniziato guardando i componenti dei vari team di allora lavorare, totalmente affascinata. Non a caso, passavo lì ore senza rendermene conto, mentre con gli occhi rubavo informazioni. Dal 2019, affiancata dagli altri componenti del team scolastico, lavoro per portare in campo una macchina funzionante; un calciatore, che resista a giorni di gare. Con l'inizio della pandemia, il supporto dei miei compagni è stato vitale per continuare a lavorare e portare risultati, nonostante il sapore dolceamaro, abbiamo lottato fino alla fine.

Ma, un robot, come fa a funzionare?

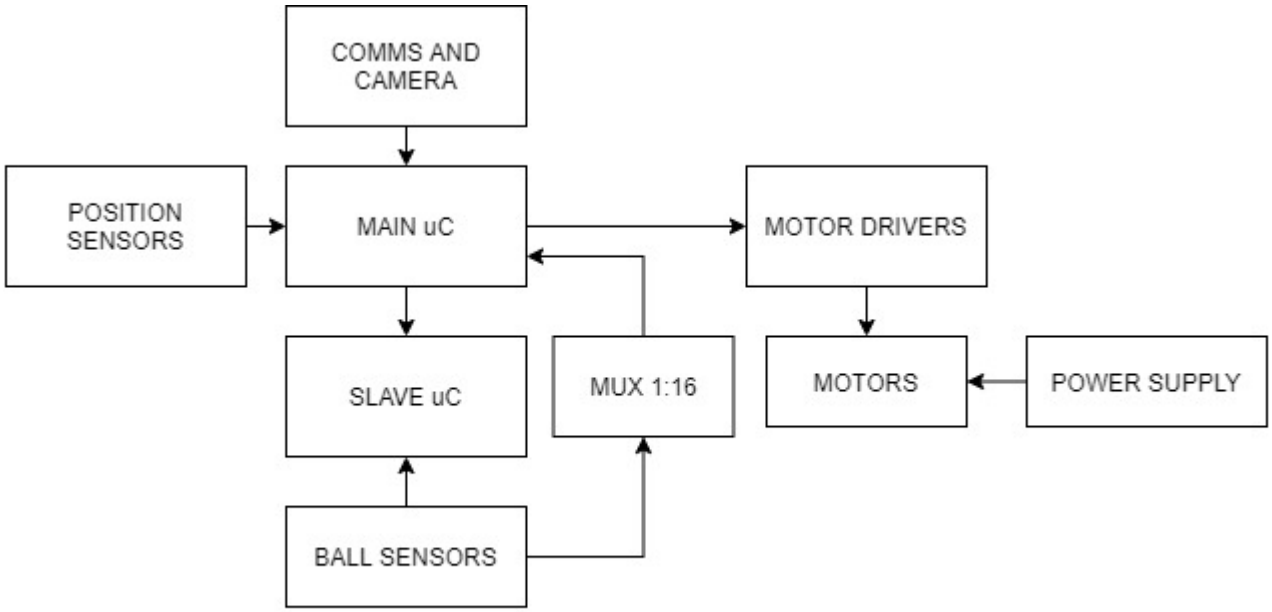
I robot si basano su una scheda elettronica, in gergo chiamata PCB – Printed Circuit Board, progettata ogni anno da un "designer" o "hardware engineer", spedita poi per la fabbricazione da vari "manufacturers". I programmatori o "software engineers", fanno poi muovere quella scheda vuota scrivendo righe di codice. L'ultimo ruolo, quello del meccanico, crea lo scheletro, ciò che lo fa stare in piedi. La cooperazione tra le tre parti è fondamentale ai fini del funzionamento e dell'armonia della macchina.

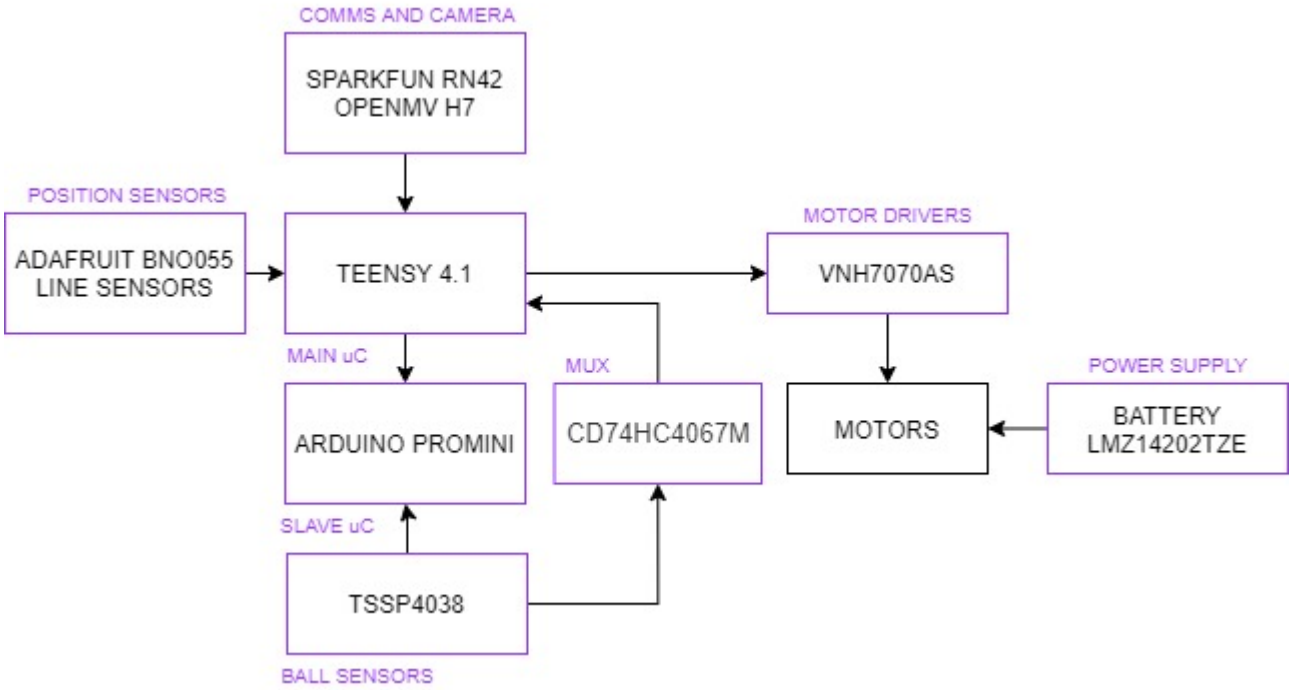
Ciò che non c'è mai stato, però, è un'affidabile piattaforma di test per provare nuovi microcontrollori, nuovi metodi di gioco, nuovo hardware. Ed è così che ho avuto l'idea di progettare una scheda che avesse questo preciso scopo, come una specie di "benchtesting", appositamente creato per il calcio robotico, con motor driver, sensori palla, fotocamera ed alloggiamento per due microcontollori diversi utilizzabili indipendentemente, per far scatenare la creatività dei ragazzi che condividono con me questa passione.

In questo settore costantemente in evoluzione, è importante poter provare ogni piccola modifica senza avere l'obbligo di spendere tante risorse per progettare qualcosa di cui, fino all'arrivo del corriere dalla Cina, fondamentalmente, non si ha la sicurezza di funzionamento; le attese ostacolano lo sviluppo del robot da tutte e tre le parti. La mia esperienza, che racconterò in seguito, è stata vitale per capire cosa realizzare per il mio esame.

Dedico quindi questo mio progetto al team SPQR, a cui auguro di usarlo in modo creativo, per schiodarsi da dogmi esistenti da anni, per poter utilizzare microcontollori nuovi, al fine di ammortizzare costi ed esplorare appieno il mondo della Robotica moderna.

Schema a blocchi





Il "testing" semplificato, con possibilità di espansione

Volevo iniziare descrivendo la mia idea, ossia una scheda di progettazione per i nuovi ragazzi che frequenteranno il corso di robotica, lo definirei quasi un "Arduino creato appositamente per lo scopo del calcio robotico", una scheda flessibile e facile da usare, con una grande possibilità di espansione per la presenza di molti connettori. La realizzazione di schede di "adattamento" era già una grande parte del corso, ma avendo già una base di sviluppo, ogni parte potrà essere sostituita rispetto a quelle presenti. Mi viene in mente un Raspberry Pi Pico, o altri moduli bluetooth. Rimane molto importante l'aspetto "software" del corso di robotica, essendo uno dei due programmatori nella squadra della scuola da ormai tre anni, ha influenzato il mio percorso in questa scuola, nonchè nel mio progetto. Gli argomenti che ho individuato, oltre ai citati prima, sono i seguenti:

- La scheda presenta vari sensori e parti "attive", tra cui 16 sensori IR, i connettori per l'utilizzo di sensori di luce, un Bluetooth (RN42, di classe 2), un 9-axis absolute orientation sensor (BNO055 di Adafruit) e una fotocamera (OpenMV H7). I sensori palla sono 16, in modo da poter utilizzare un multiplexer.
- Gli attuatori, in particolare i motori in continua, poichè la scheda è adibita ad ospitare fino a quattro VNH7070AS, per utilizzare tre o quattro motori in continua (12V).
- Microcontrollori, nella scheda ce ne sono ben due, uno sarà sicuramente un Teensy 4.1, una scheda con un ARM Cortex M7 di potenza di calcolo sbalorditiva, l'altro con molta probabilità sarà un Arduino Pro Mini 3V3, una piccola scheda con un ATMEGA328P che ha tutte le potenzialità di un Arduino ma in dimensioni ridotte.
- Il circuito di alimentazione funziona con una batteria a tre celle LiPo da 12v e un regolatore switching, il LMZ14202TZE.

Nelle parti dedicate ci sarà una spiegazione approfondita di ogni componente.

"What's on board?"

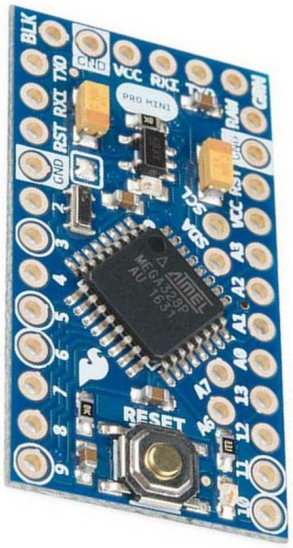
In questa sezione ci sarà una descrizione dettagliata di tutte le parti della scheda e del loro funzionamento, distinte in due macroaree: Microcontrollori e Sensoristica. Ciò che non rientra nelle aree sarà discusso separatamente.

Microcontrollori

- La mente, Teensy 4.1

Un microcontrollore a 3,3 Volt molto leggero e ,soprattutto, molto potente, ideale per evolvere i nostri robot, dato che negli ultimi due anni abbiamo usato un suo fratello più piccolo, il Teensy 3.5, poichè avevamo bisogno di una certa flessibilità nella programmazione, che ormai è un ibrido tra Arduino (è utilizzabile la stessa Arduino IDE, con un plugin "Teensyduino") e C++. Qui inizia la via per creare PCB con un solo microcontrollore, nonostante lo "slave" di supporto ci sia comunque, per facilitare la transizione. Uno dei punti di forza è il suo microprocessore, un ARM Cortex-M7 che porta la velocità della scheda a un minimo di 600 MHz ed ad un massimo di 912 Mhz, assicurando una considerevole potenza di calcolo, con due operazioni per ogni ciclo di Clock, e una velocità notevole con il branch prediction. La sua versatilità si ritrova anche nella comunicazione, avendo un Ethernet Controller ed otto porte seriali UART (ricevitore-trasmettitore asincrono universale) ad alto baudrate, tre di SPI (Serial Peripheral Interface) con tecnologia FIFO, tre di I2C, tre CAN bus (Controlled Area Network-per automazione industriale) e due I2S per l'audio. Per la prima volta abbiamo anche una memoria QSPI espandibile, quattro kilobyte di EEPROM, una SD e della RAM utilizzabili per più prontezza nelle operazioni, un totale di 55 contatti multifunzione con interrupt, sia PTH che SMD, di cui 23 con funzione di PWM (Pulse Width Modulation-per il comando di motori), 18 analogici ed una funzione di ON/OFF. Dall'azienda produttrice sono anche incoraggiati diversi modi d'uso, ma noi abbiamo sempre utilizzato VSCode con PlatformIO IDE, per una maggiore flessibilità.

- Il braccio, Arduino Pro Mini



L'Arduino Pro Mini è una scheda a microcontrollore prodotta da SparkFun, molto piccola nelle dimensioni e grande nelle potenzialità, oltre ad avere il vantaggio di funzionare a 3,3V ed a 8MHz a differenza di molte altre schede microcontrollore sul mercato.

Ha un approccio minimal ed essenziale, utilizzando allo stesso tempo un ATMega 328P, necessità però del suo programmatore, si trova a pochi euro dappertutto.

Tutto questo a un peso minimo, di un paio di grammi, ricordando che anche l'ingombro in una scheda è una parte importante.

Nella scheda è utilizzato come "slave", direttamente collegato al Teensy (master) su una delle porte seriali, a cui affidiamo parti di calcolo che fino ad ora il Teensy 3.5 non riusciva ad effettuare contemporaneamente a tutte le altre operazioni, come la lettura dei dati della palla e la loro processazione. Si programma con Arduino IDE.

In futuro auspico che, anche utilizzando la mia scheda, possa essere integralmente rimosso, poichè è un impegno da mantenere, con codice apposito e spesso abbastanza complesso. Ha un totale di 22 pin I/O, analogici e digitali; 6 di questi possono essere usati come PWM.

- Regole e linguaggi di programmazione

Sensoristica

- Bluetooth, per comunicare

Fino ad ora, abbiamo sempre utilizzato dei moduli bluetooth prodotti da SparkFun, il BlueSMiRF Silver, che spicca per il suo costo ragionevole, le sue dimensioni piccole e per la relativa facilità ed elasticità nel funzionamento. Questa è una schedina per favorire l'utilizzo del modulo RN-42 di Roving Networks, che funziona con comunicazione seriale, con un grande range di baudrate disponibili, da 2400 a 115200 bps, senza alterare il dato. Il connettore è fatto in modo per essere collegato a vari microcontrollori, tra cui il nostro Arduino Pro Mini, per utilizzarlo con gli altri, basta invertire RX e TX, come in ogni comunicazione di tipo UART. Essendo un bluetooth di classe 2, è conforme alle regole del gioco per comunicare a brevi distanze, consumando comunque pochissima corrente, circa 26uA, in "sleep mode", ossia quando non ha connessioni attive.

- Sensori IR, per localizzare la palla

Da anni ormai, per localizzare la palla, ci affidiamo ai sensori di prossimità infrarossi Vishay TSSP4038. Sulla scheda ce ne sono 16, che dividiamo a loro volta in quattro "powerblock", stabilizzati a 3,3V da un filtro RC. Sono molto compatti, montandolo sul lato inferiore della PCB si risparmia spazio senza diminuire la qualità della lettura del segnale, con una frequenza di funzionamento a 38kHz e una distamza fino a due metri. Per ora utilizziamo l'interpolazione per processare i dati da mandare al master.

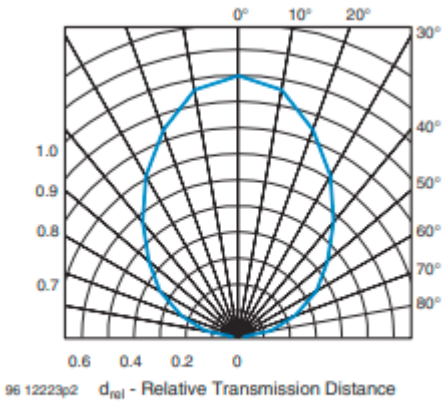


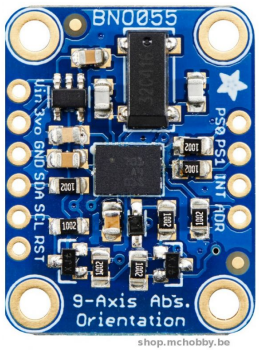
Fig. 8 - Directivity

Grafico del campo visivo del sensore.

- Camera, per vedere

I connettori montati sulla scheda sono stati pensati per l'utilizzo di un OpenMV Cam H7, una fotocamera piccola e potente, adatta per gli algoritmi di "machine vision" e ad altri obbiettivi oltre a quello in dotazione. Si programma in MicroPython, una versione di Python ottimizzata per l'uso con microcontrollori. La utilizziamo per l'operazione di color tracking e la rectangle detection per individuare le porte all'interno del campo, nonostante si usi in campi come il riconoscimento di QR e barcode e l'Eye tracking. Monta un Cortex M7 a 480MHz, quindi molto recettivo. Offre la possibilità di fare foto e memorizzare dati nella MicroSD esterna, con ben quattro metodi diversi per comunicare i dati: SPI, UART, I2C e CAN bus. Ogni pin di I/O offre interrupt e PWM. Il sensore di immagini è l'OV7725, che può scattare foto 640x480 a 8 bit in scala di grigi o a 640x480 a 16 bit RGB con piccole differenze nei frames per second. Si interfaccia a molti microcontrollori e PC con una libreria built in (RPC), funzionante con molti metodi di comunicazione. Qui sotto si potrà vedere in che modo la abbiamo utilizzata nel corso degli anni.

- IMU, per sapere dove ci si trova



Per IMU (Inertial Movement Unit), si intende quella che in gergo chiamiamo "bussola", ossia un sensore o più sensori di posizione a più assi con cui ci orientiamo. Utilizziamo da anni quello di AdaFruit, il BNO055, che ne offre nove. Utilizza un Cortex M0, un accelerometro, un giroscopio e un magnetometro, trovando il suo "nord" automaticamente e comunicando i dati ogni 10ms via I2C o UART, rendendolo adatto al plug-and-play. Di questo sensore di posizione utilizziamo la "Absolute Orientation (Euler Vector)", che dà la posizione su una sfera a 360°, ma ha anche una posizione a quaternioni, un vettore di accelerazione lineare, magnetica e di gravità che vengono calcolati in modo automatico. Sulla scheda si trova sul bottom per ottimizzare gli spazi.

- Multiplexer, un nuovo modo di gestire i dati

La novità più grande di questa scheda è un nuovo modo di leggere i dati della palla, mirata a testare se il secondo microcontrollore sia effettivamente necessario. Essendo presenti 16 sensori, un multiplexer 1:16 dovrebbe aiutarci a risolvere questo problema. La scelta è stato un CD74HC4067M, che rispetto ad altri, ha un propagation delay time molto breve e, soprattutto ha un operating voltage da 2 a 6V DC. I quattro selettori sono collegati a Teensy, permettendoci di risparmiare ben 12 pin per altri usi, e selezionando il sensore che si vuole leggere (vedere tabella di verità riportata in basso). Il package SOIC-24 garantisce anche un peso inferiore al grammo, ed un ingombro minimo sulla PCB.

TRUTH TABLE

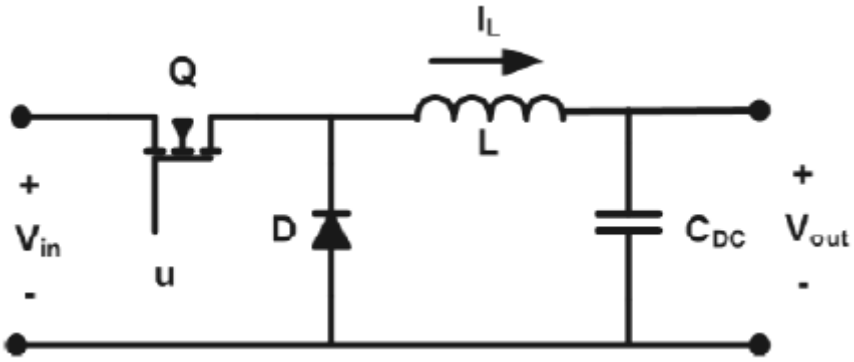
S0	S1	S2	S3	$\overline{E}$	SELECTED CHANNEL
X	X	X	X	1	None
0	0	0	0	0	0
1	0	0	0	0	1
0	1	0	0	0	2
1	1	0	0	0	3
0	0	1	0	0	4
1	0	1	0	0	5
0	1	1	0	0	6
1	1	1	0	0	7
0	0	0	1	0	8
1	0	0	1	0	9
0	1	0	1	0	10
1	1	0	1	0	11
0	0	1	1	0	12
1	0	1	1	0	13
0	1	1	1	0	14
1	1	1	1	0	15

H= High Level
L= Low Level
X= Don't Care

- Circuito di alimentazione

Rispetto agli scorsi anni, il circuito di alimentazione si è semplificato moltissimo, passando da tre regolatori LDO ad uno switching, che garantisce un'efficienza maggiore. Alla base abbiamo sempre una batteria a tre celle LiPo, la sua tensione passa per due condensatori di alta capacità, per stabilizzare la tensione per il funzionamento a 12V. Qui le cose si fanno interessanti, perchè la tensione deve scendere da 12V a 3,3V per attivare tutta la parte "logica della scheda". Utilizzando il regolatore switching LMZ14202TZE, non ci sono problemi anche con alte correnti.
Ma come funziona questo regolatore?
Generalmente, ogni step-down converter ha queste parti fondamentali interne:

- Due "switch", che sono un transistor (qui è usato un MOSFET) e un diodo
- Un induttore, connesso alla fonte dallo switch, che scarica la sua "energia" sul carico
- Un condensatore, per mantenere la tensione in uscita costante



Il calore creato per effetto Joule viene facilmente dissipato dai vari poligoni di massa presenti, nonostante questo particolare regolatore abbia un "thermal protection shutdown", che fa entrare in standby il dispositivo a circa 165°C, non alimentando il MOSFET e scaricando il pin di Soft-Start a massa. Questo contatto particolare serve aggiunge al regolatore la funzione di UVLO (Undervoltage-Lockout), che protegge il dispositivo da cadute improvvise di tensione e tensioni negative. Con un carico così leggero, utilizza la DCM, ossia modalità di funzionamento discontinuo; l'operazione di "switching" inizia con corrente nulla sull'induttore, motivo per cui servono dei condensatori esterni per il funzionamento, perchè la corrente tornerà nulla prima della fine della commutazione. La conversione riinizierà quando la tensione sul pin FeedBack si abbasserà oltre la soglia regolata con il carico esterno. Questa modalità garantisce un'uscita pulita e con una frequenza minore, con un'attenuazione esterna delle perdite.

- Motor Driver

Per far muovere il nostro robot, indipendentemente dalla scelta dei motori, abbiamo bisogno di un piccolo circuito chiamato motor driver. Sulla scheda ce ne sono quattro, ma se ne possono utilizzare anche solo tre, e sono i VNH7070AS, prodotti da ST. Essendo molto piccoli, pesano anche molto poco, e ci hanno permesso di passare da due PCB ad una sola. Questo motor driver sono "full bridge", ossia funzionano con un ponte H, che con l'inversione della corrente di carico, permette l'inversione del senso di rotazione. La regolazione della velocità si fa con la tecnica PWM fino a 20kHz, un segnale digitale periodico con frequenza costante caratterizzato da due stati, di durata variabile, uno in cui il segnale è a livello logico alto (On-time), dove la velocità aumenta progressivamente, e l'altro in cui è a livello logico basso (Off-time), dove il motore tende invece a frenare. A causa dell'elevata costante di tempo meccanica, la velocità di rotazione dl motore non può seguire istantaneamente la tensione di alimentazione impulsiva generata dalla modulazione PWM. La velocità effettiva di rotazione sarà pertanto una media tra il suo andamento durante l'On-time e quello che sia ottiene durante l'Off-time. Questo driver ha di particolare il funzionamento in half-bridge, un controllo di attività e "fault detection" con i pin CurrentSense e SEL0, e la presenza di molti circuiti di protezione da corti, tensioni negative e sbalzi di temperatura.

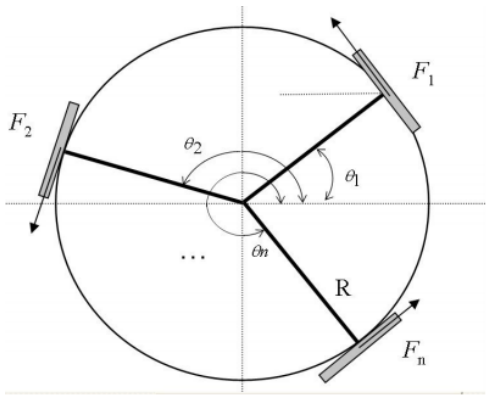
AGGIUNGI FOTO H BRIDGE

- "Eppur si muove!", il moto olonomico

Il moto olonomico è ciò che è alla base di tutti i robot creati dal team fino ad ora, ciò che li ha sempre fatti muovere in modo preciso ed accurato. Si definisce "olonomica" la relazione tra i gradi di libertà totali e controllabili di un determinato sistema, in questo caso, il robot. Se questi sono uguali, il robot si dice olonomo, perchè, grazie a ruote omnidirezionali, può muoversi idealmente in tutte le direzioni; si prenda ad esempio una ruota con intorno delle altre rotelline libere di ruotare, allora sarà possibile muoversi liberamente su X e Y, intorno a un centro di gravità Θ. Parlando di ruote, il fatto che siano onmidirezionali, significa che, muovendosi, creano una forza che si traduce in un movimento rotante e traslazionale. Per questo le ruote utilizzate sono n>=3, disposte ad angoli noti. Per 4, abbiamo utilizzato sia la configurazione perpendicolare che la configurazione incidente (aggiungi gradi). Quando i motori sono attivati, si ottengono le forse trainanti e traslazionali, oltre alla forza e alla coppia rotante. La forza trainante è misurabile con:

Fi = Coppia * r (ruota)

Tramite la forza, è possibile determinare accelerazione (a) e centro di massa (ω) del robot:



$$a = \frac{1}{M}(F_1 + F_2 + \ldots + F_n)$$

$$\dot{\omega} = \frac{R}{I}(f_1 + f_2 + \ldots + f_n)$$

dove M è la massa del robot, R il raggio, fi è la magnitudine di Fi, ed I è il momento di inerzia. Le componenti x e y dell'accelerazionè sono ottenibili moltiplicando l'opposto di fi per, rispettivamente, sinΘx e cosΘx. Usando la geometria euclidea, si possono calcolare le velocità delle ruote e del robot con anche la sua velocità angolare. Mettendo le velocità dei motori in un vettore (v1, v2,..., vn)^T e le velocità di rotazione tangenziale, quelle calcolate sulle distanze euclidee e la velocità angolare in un altro vettore (vx, vy, Rω) ^T , le due parti di movimento orizzontale e verticale sono -sinΘx e cosΘx. Si evince poi che moltiplicando il secondo vettore per una matrice che contiene tutte le componenti del movimento si ottiene la D, o "velocity coupling matrix", che è la rappresenta la relazione fra coppia e velocità. Non essendo questa quadrata, si fa la pseudoinversa, e si ottiene una matrice D+ che avrà le velocità corrette. Per controllare i motori utilizziamo un controllo PID con la ArduinoPIDLibrary, calibrato sugli angoli dei vari motori e nella gestione delle velocità, per evitare oscillazioni.

- Fonti e Datasheets

- https://www.pjrc.com/store/teensy41.html – Teensy 4.1
- https://www.sparkfun.com/products/16771 – Teensy 4.1
- https://www.sparkfun.com/products/11114 – Arduino ProMini
- https://www.sparkfun.com/products/12577 – BlueSMiRF BT module
- https://www.sparkfun.com/products/12574 – RN-42 BT module
- https://www.vishay.com/docs/82458/tssp40.pdf – Sensori IR
- https://openmv.io/products/openmv-cam-h7 – Camera OpenMV H7
- https://www.adafruit.com/product/2472 – AdaFruit BNO055 IMU
- https://www.ti.com/lit/ds/symlink/cd74hc4067.pdf?HQS=dis-mous-null-mousermode-dsf-pf-null-wwe&ts=1621428750713&ref_url=https%253A%252F%252Fwww.mouser.fr%252F – MUX 1:16
- https://www.ti.com/lit/ds/symlink/lmz14202.pdf?HQS=dis-mous-null-mousermode-dsf-pf-null-wwe&ts=1621437144244 – LMZ14202TZE Switching voltage regulator
- https://www.st.com/resource/en/datasheet/vnh7070as.pdf – Automotive Full H-Bridge Motor Driver VNH7070AS
- https://people.idsia.ch//~foerster/2006/1/omnidrive_kiart_preprint.pdf – Holonomic robot movement