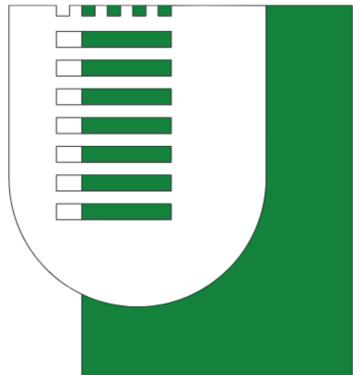


**Meccanica delle vibrazioni:**  
Misure di un robot mobile con Arduino



**TOR VERGATA**  
UNIVERSITY OF ROME  
—  
School of Engineering

Realizzato da:  
Grande L., Iacovacci G., Ortolani G

# Programma del progetto

1 Scelta, sviluppo e setup sperimentale

2 Analisi matematica modello

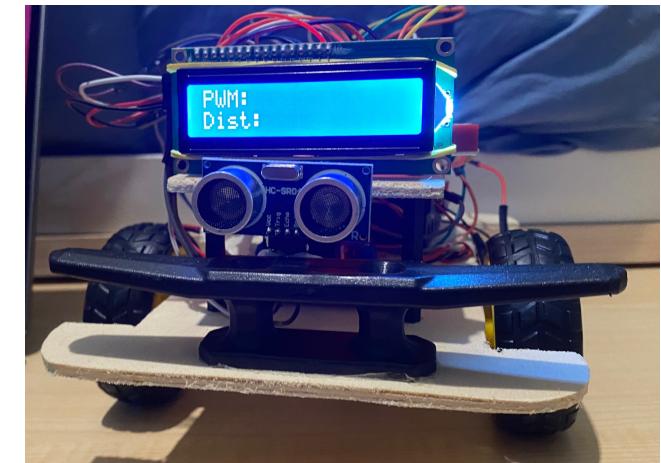
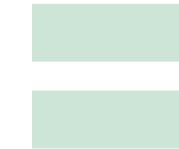
3 Analisi dei dati

4 Conclusione

# Scelta, sviluppo e setup sperimentale

La nostra scelta

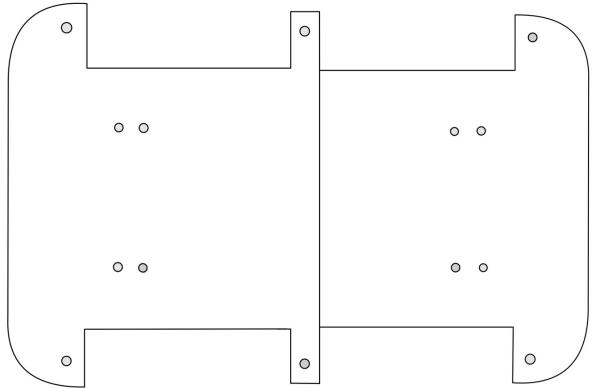
Realizzazione pratica di un uniciclo a tre ruote controllato con Arduino



Con l'obiettivo, dopo la realizzazione, di studiare le vibrazioni in due modalità di frenata: STOP e ABS

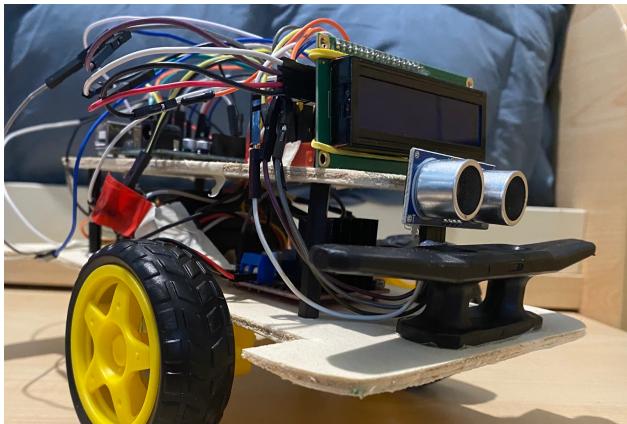
# Scelta, sviluppo e setup sperimentale

## Sviluppo telaio



Per la costruzione del telaio, abbiamo scelto di disegnare una sagoma da poter seguire durante l'intaglio del compensato da 4mm.

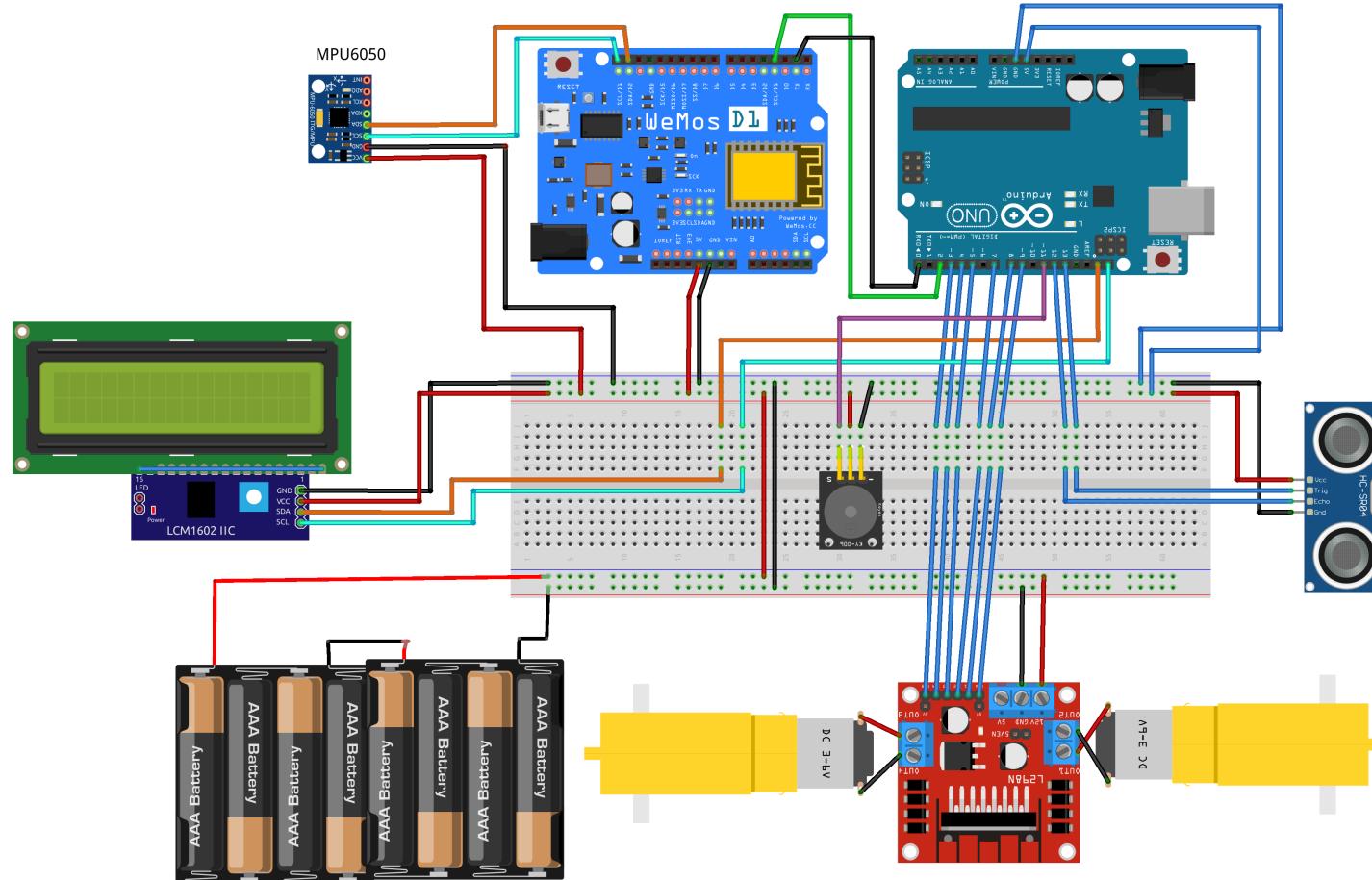
Abbiamo optato per un design molto semplice con paraurti anteriore e posteriore.



A causa dell'ingombro del pacco batterie a 12V, abbiamo scelto di costruire un secondo «piano» per aver più spazio a disposizione

# Scelta, sviluppo e setup sperimentale

## Sviluppo controllo Arduino



fritzing

# Scelta, sviluppo e setup sperimentale

## Sviluppo controllo Arduino

Per lo sviluppo dei codici Arduino sono state utilizzate diverse librerie.

```
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
#include <Wire.h>
#include <MPU6050.h>
```

Abbiamo due diversi codici, uno per la frenata STOP e uno per la frenata ABS

- Nel codice della frenata STOP vengono disattivati, in feedback alla distanza, i PWM dei motori
- Nel caso della frenata ABS, vengono modulati, tramite un PID, i PWM dei motori

L'attivazione dei motori può essere effettuata con una **digitalWrite()**, specificando se la tensione deve uscire su forward o backword di un motore.

Mentre il PWM relativo ad un motore può essere settato tramite **analogWrite()**.

```
digitalWrite(fwD, HIGH);
digitalWrite(bwD, LOW);
digitalWrite(fwS, HIGH);
digitalWrite(bwS, LOW);
```

Motori accesi in forward

```
digitalWrite(fwD, LOW);
digitalWrite(bwD, LOW);
digitalWrite(fwS, LOW);
digitalWrite(bwS, LOW);
```

Motori spenti

```
analogWrite(enD, val2Set);
analogWrite(enS, val2Set);
```

Settaggio PWM

# Scelta, sviluppo e setup sperimentale

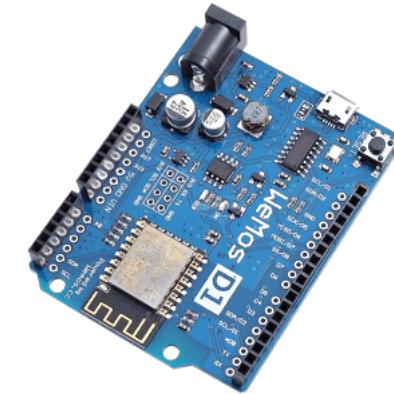
## Sensor fusion



Il sensore MPU6050 è stato utilizzato per campionare i dati delle accelerazioni e del giroscopio lungo i tre assi.



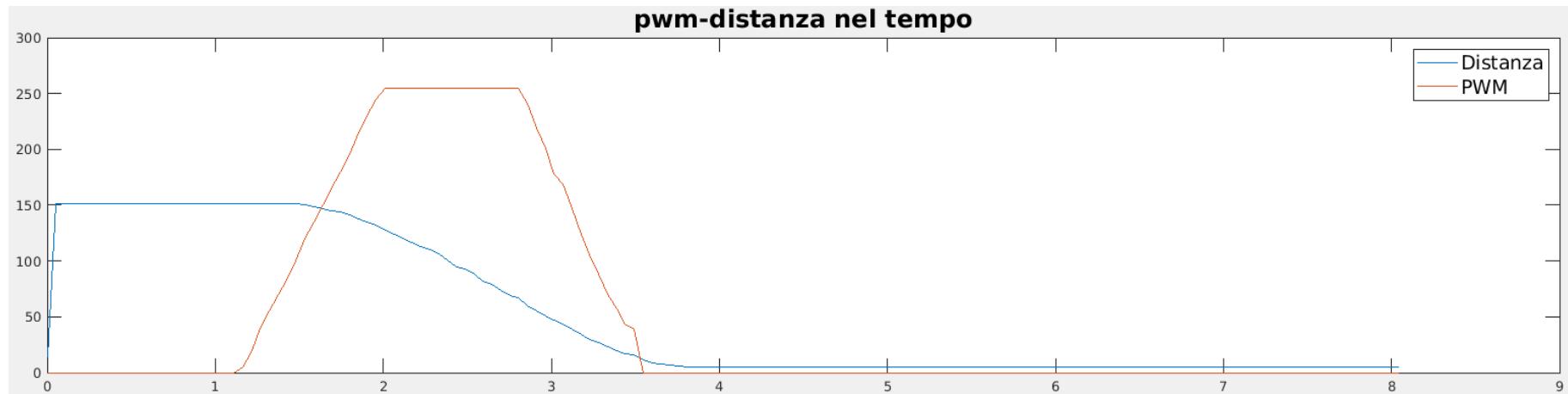
Il sensore HCSR04 è stato utilizzato per il campionamento della distanza da un ostacolo.



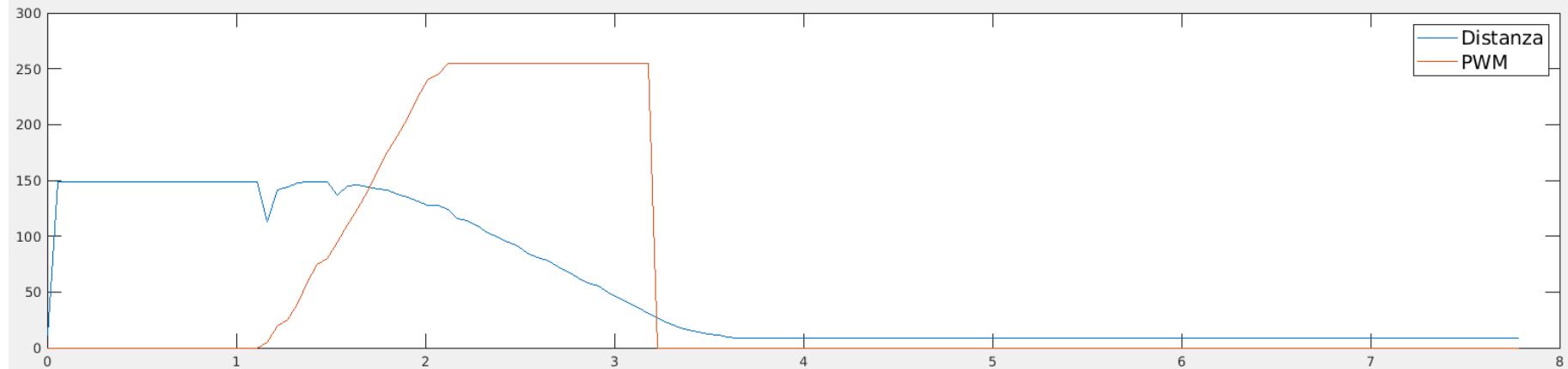
Il sensore ESP8266, ospitato da un Wemos d1 è stato utilizzato per l'inoltro dei dati al PC.

# Scelta, sviluppo e setup sperimentale

## Distanza-PWM



Modalità ABS



Modalità Stop

# Programma del progetto

1 Scelta, sviluppo e setup sperimentale

2 Analisi matematica modello

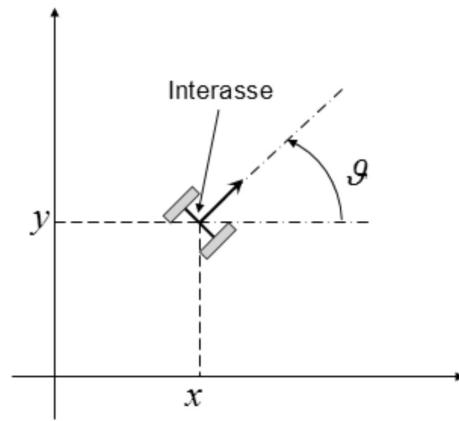
3 Analisi dei dati

4 Conclusione

# Analisi matematica modello

## Cinematica uniciclo

Per poter studiare le vibrazioni dell'uniciclo abbiamo dovuto prima di tutto comprendere le equazioni matematiche che lo descrivono:



**Equazioni uniciclo:**

$$\begin{cases} \dot{x} = v_1 \cos \theta \\ \dot{y} = v_1 \sin \theta \\ \dot{\theta} = v_2 \end{cases}$$

Abbiamo dovuto anche considerare l'impossibilità del modello di avere un moto nella direzione parallela a quella attuale. Ciò si trasforma in un vincolo sulle velocità detto: **vincolo non olonomo**

$$x' \sin \vartheta - y' \cos \vartheta = 0$$

Il passo successivo è stato di trasformare le equazioni matematiche in codice Arduino che permettesse il movimento del nostro uniciclo.

Fondamentale per il nostro studio è l'analisi della stabilità. La stabilità del controllo, infatti, garantisce al sistema proprietà come: **prevedibilità, sicurezza, affidabilità ed efficienza.**

Dato un sistema del tipo:  $a_n p^n + a_{n-1} p^{n-1} + \cdots + a_1 p + a_0$

**Condizioni necessarie stabilità:**

- Nessun coefficiente deve essere nullo
- I coefficienti devono essere tutti positivi o negativi

Tali condizioni sono necessarie, ma non sufficienti



**Valutazione stabilità:**

**1. Valutazione radici del polinomio:**

*«Il sistema è stabile se e solo se le sue radici hanno parte reale minore o uguale a zero. Se sono tutte a parte reale minore il sistema è astintoticamente stabile.»*

1. Metodo Routh-hurwitz
2. Nyquist
3. Luogo delle radici

# Programma del progetto

1 Scelta, sviluppo e setup sperimentale

2 Analisi matematica modello

3 Analisi dei dati

4 Conclusione

# Analisi dei dati

Il modello matematico è stato solo il punto di partenza per il nostro studio, ma solamente tramite un'analisi dei dati siamo riusciti a comprendere effettivamente quale sistema fosse più stabile e quindi sicuro.

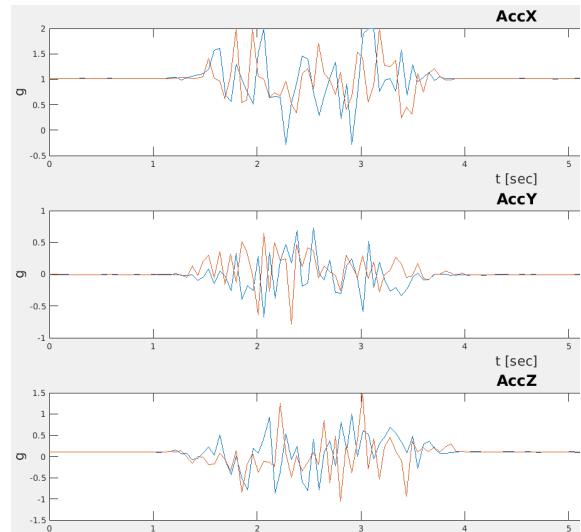
Per tale motivo abbiamo eseguito un'attenta analisi sulla base di misurazioni prese dal giroscopio e dall'accelerometro. Tale analisi è stata effettuata nel **dominio del tempo** e in **frequenza**.

**Dominio del tempo:** ci ha permesso di studiare le variazioni nel tempo delle misurazioni provenienti dai sensori. Variazioni brusche indicano una minore stabilità.

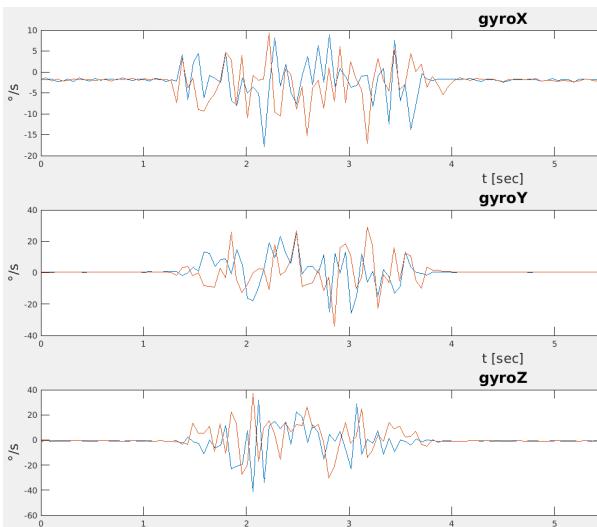
**Analisi in frequenza:** ci ha permesso di valutare le frequenze dei due sistemi e la risposta dinamica in seguito a una frenata. Anch'essa ci dà importanti informazioni sulla stabilità.

# Analisi dei dati

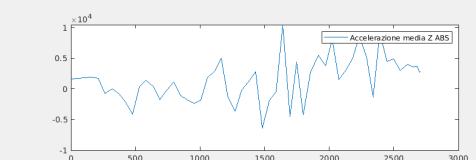
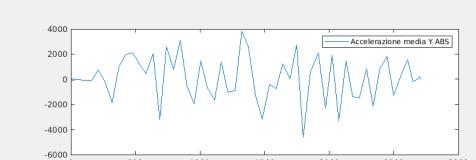
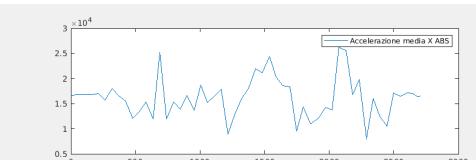
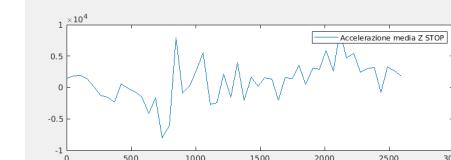
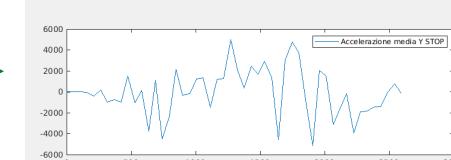
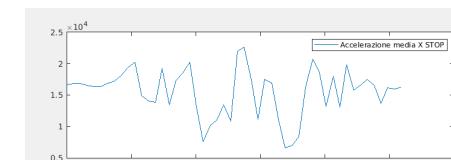
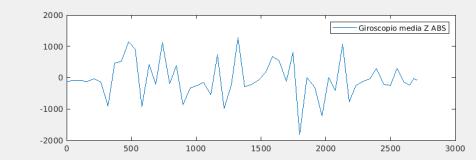
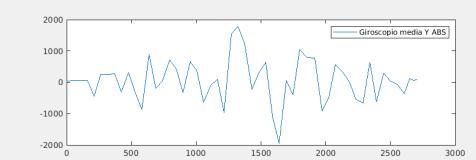
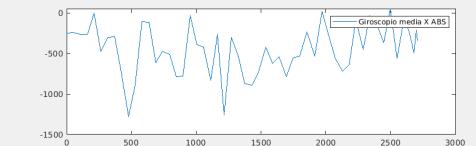
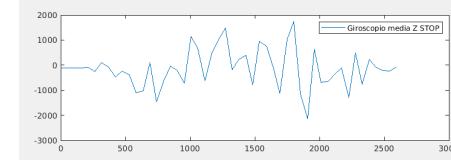
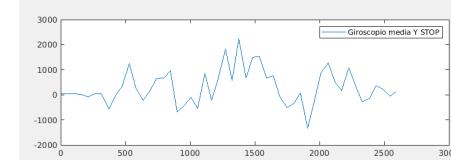
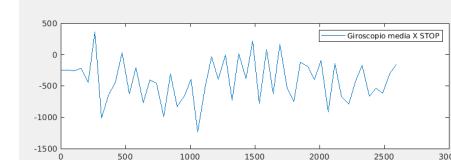
Analisi nel dominio del tempo, confronto tra le due modalità



ABS  
STOP

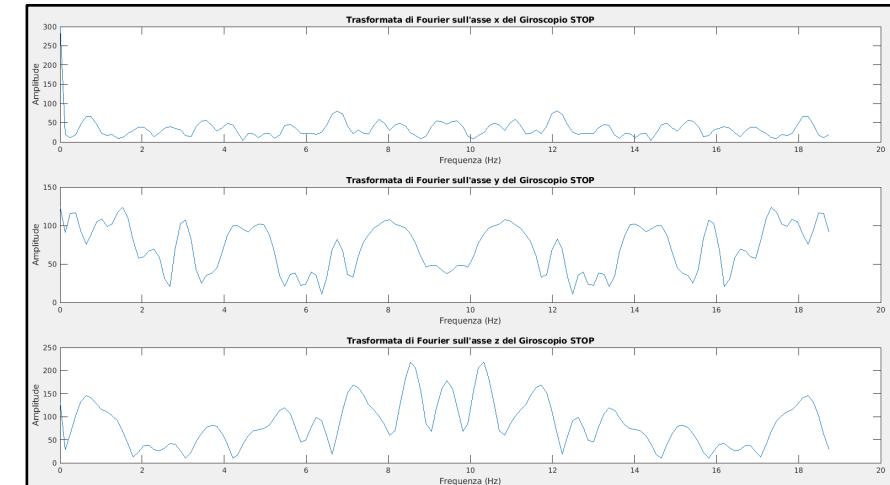


Media delle misure  
ottenute:

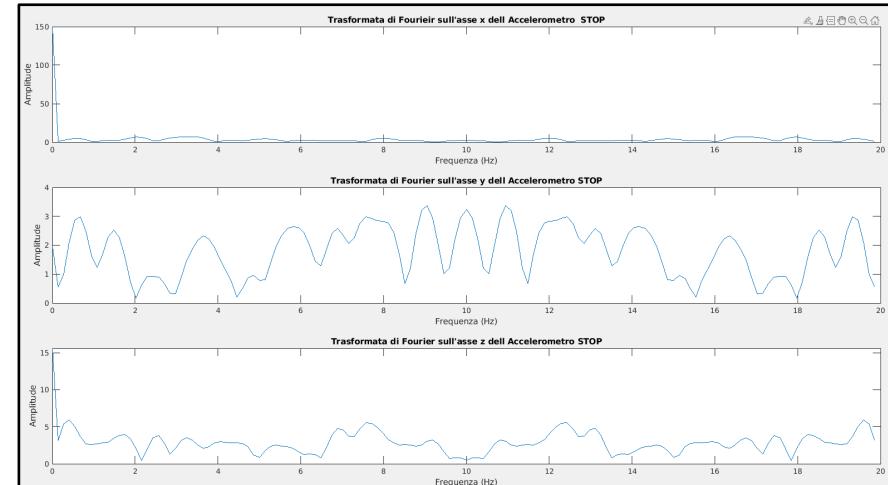


# Analisi dei dati

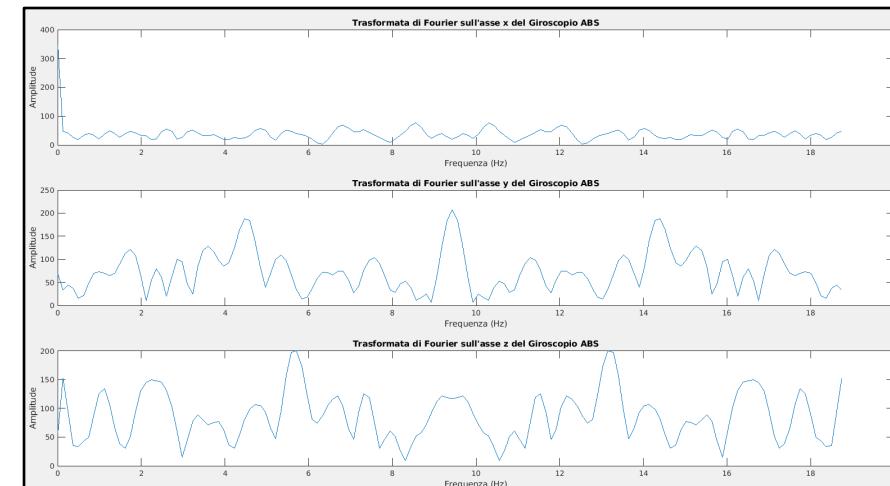
Analisi nel dominio delle frequenze, confronto tra le due modalità



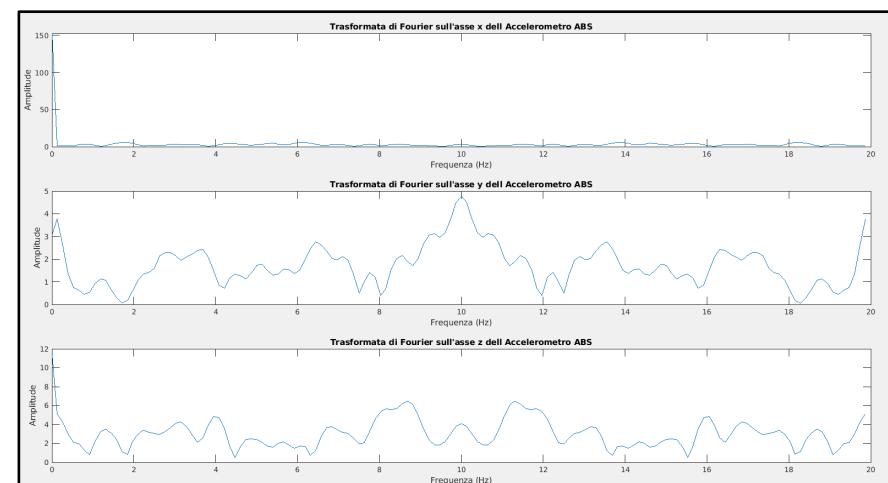
Giroscopio  
modalità  
Stop



Accelerometro  
modalità  
Stop



Giroscopio  
modalità  
ABS



Accelerometro  
modalità  
ABS

# Programma del progetto

1 Scelta, sviluppo e setup sperimentale

2 Analisi matematica modello

4 Analisi dei dati

4 Conclusioni

# Conclusione

## Video

