

Dipartimento di Ingegneria Enzo Ferrari  
Corso di Laurea triennale in Ingegneria Informatica  
*Università degli studi di Modena e Reggio Emilia*

# ***Sistemi e algoritmi di navigazione autonoma di robot indoor e sviluppo di un prototipo Arduino ©***

Laureando:

**Luca Munarini**

Relatore:

**Prof. Nicola Bicocchi**

**Anno accademico 2016-2017**

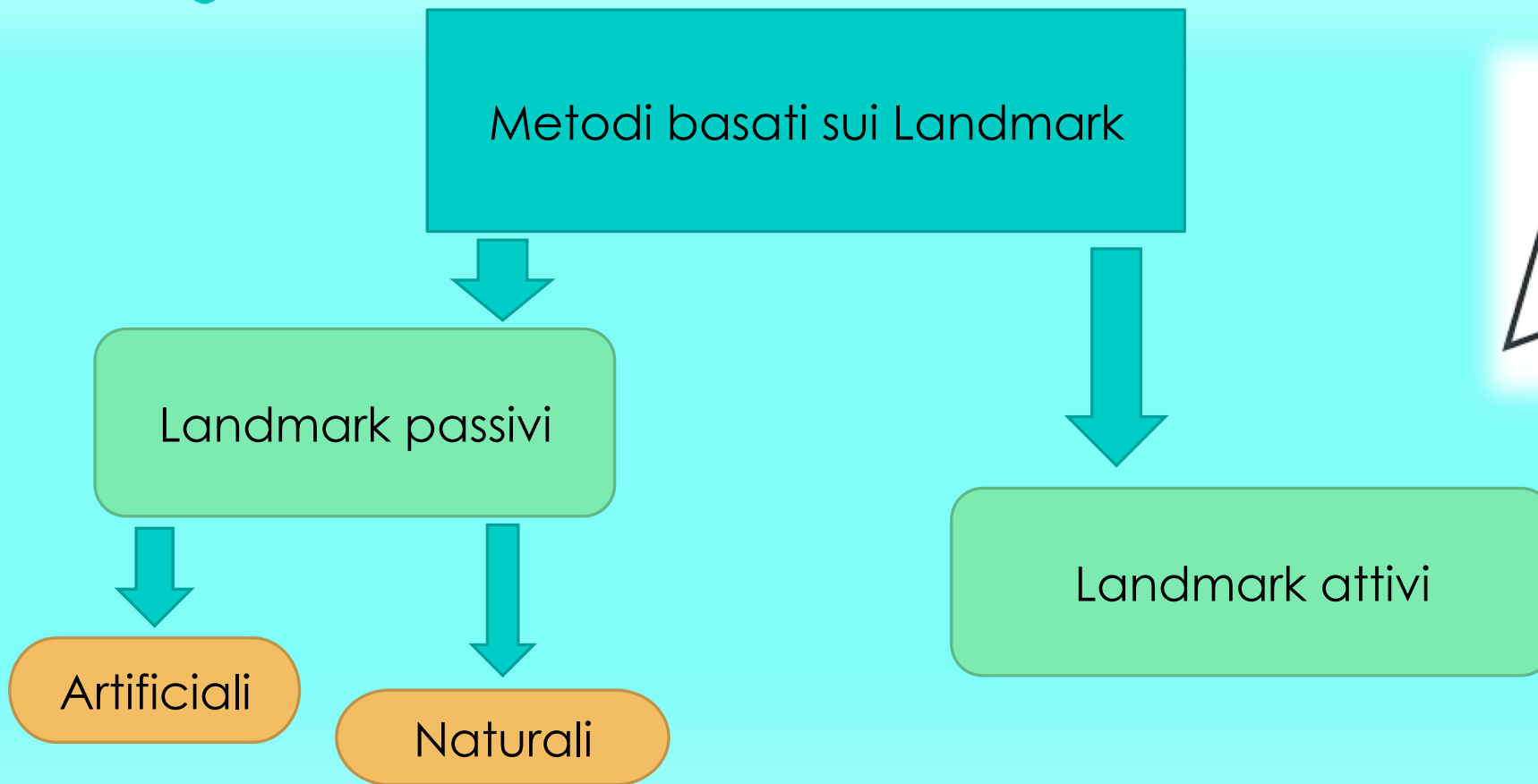
# Introduzione

- ❑ Per poter navigare un robot deve avere l'abilità di stabilire la propria posizione e pianificare un percorso.
- ❑ Una volta localizzato, la navigazione rende il robot capace di **muoversi autonomamente** nell'ambiente in cui si trova. Esistono due categorie di metodi di navigazione:
  - **Navigazione relativa** → le misure sono indipendenti una dall'altra
    - Navigazione odometrica
    - Navigazione inerziale
  - **Navigazione assoluta** → le misure si basano sui dati precedenti
    - Metodi basati su Landmark
    - Metodi basati su mappe

# Sistemi di navigazione relativa

- *Navigazione inerziale*: si utilizzano degli encoder montati sulle ruote del veicolo
- *Navigazione odometrica*: vengono utilizzati alcuni sensori come accelerometri e giroscopi per misurare la velocità di rotazione e l'accelerazione del robot.

# Sistemi di navigazione assoluta



# Sistemi di navigazione assoluta

Metodi basati su mappe

Mappe topologiche

Contengono la **descrizione** dell'ambiente in cui si trova il robot, come limiti e ostacoli, in un sistema globale.

Mappe geometriche

L'ambiente è rappresentato tramite interconnessione di **nodi** (i luoghi dell'ambiente) e **archi** (le azioni intraprendere per spostarsi da un all'altro).

# Un algoritmo di navigazione: Virtual Force Field

Il metodo si basa su due concetti:

- Griglie di certezza nella rappresentazione degli ostacoli
  - Suddivisione dello spazio in una griglia di celle
  - Ogni cella ha un valore: maggiore è questo valore e più alta è la possibilità di occupazione della cella
- Campi potenziali per la navigazione
  - Una cella occupata esercita una forza repulsiva
  - Il target esercita una forza attrattiva

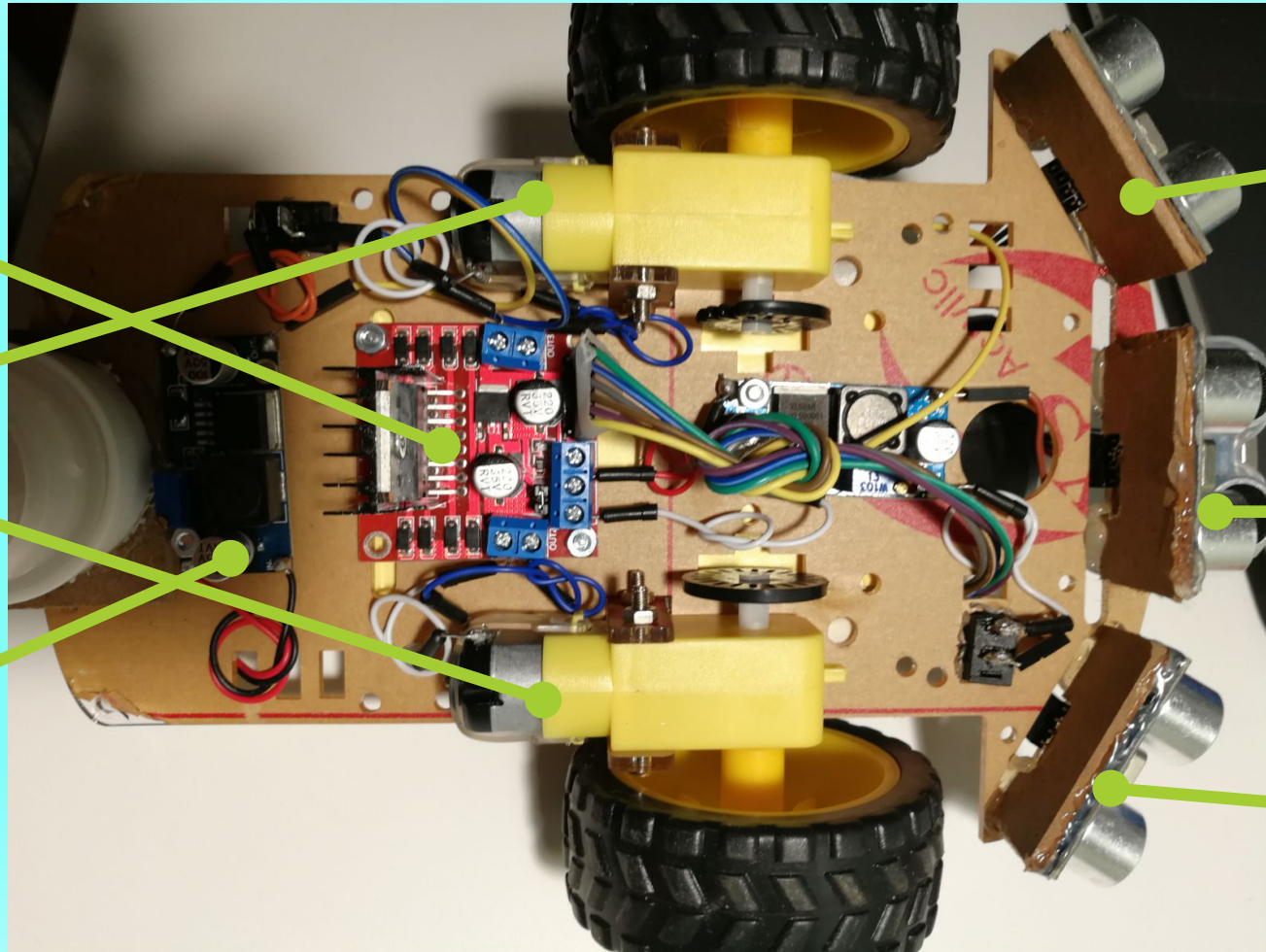


# Prototipo di robot autonomo

Driver L298D

Motori DC

Convertitore  
buck



HC-SR04

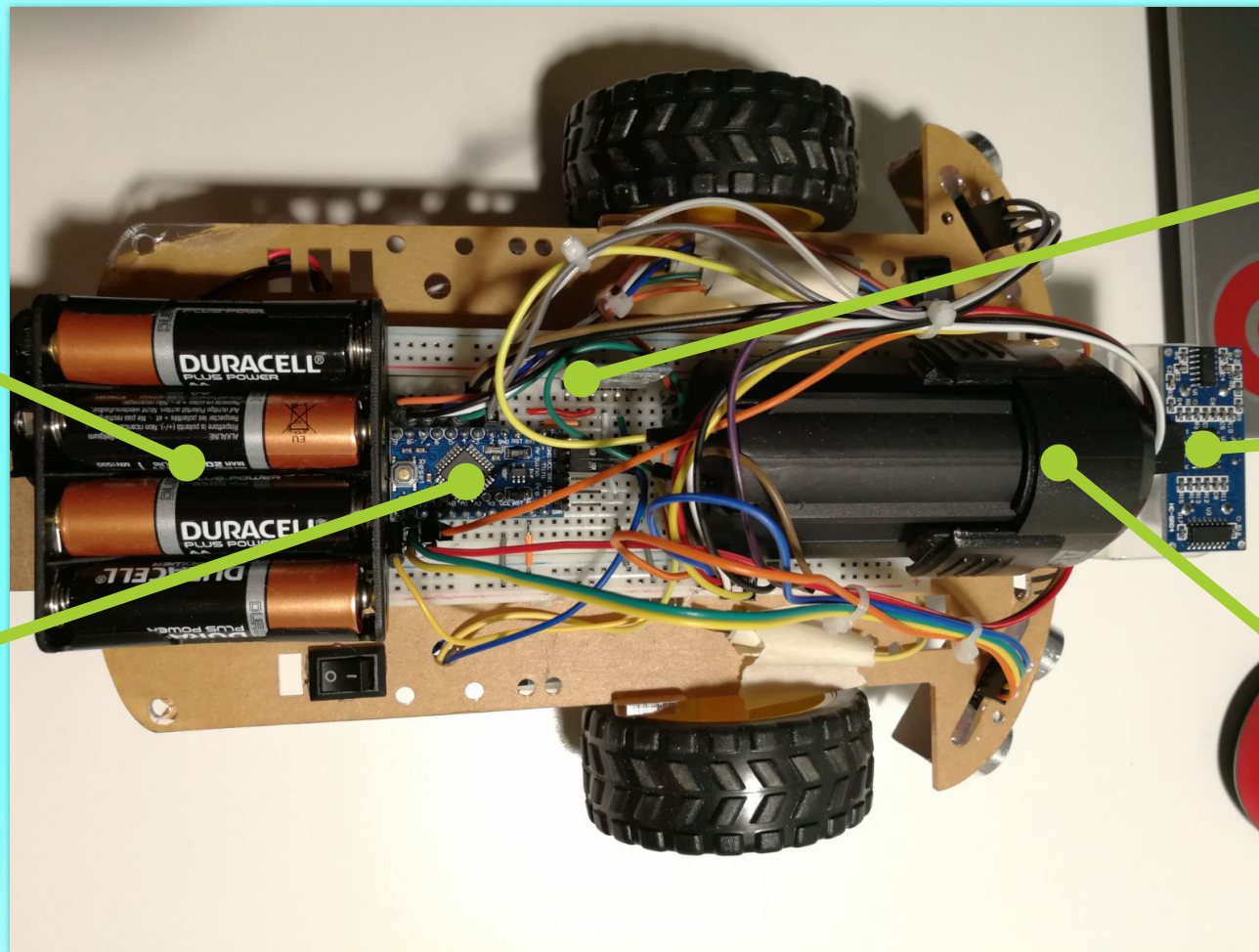
HC-SR04

HC-SR04

# Prototipo di robot autonomo

4x AA per  
alimentazione  
elettronica

Arduino  
Pro Mini



Modulo  
Bluetooth  
HC-05

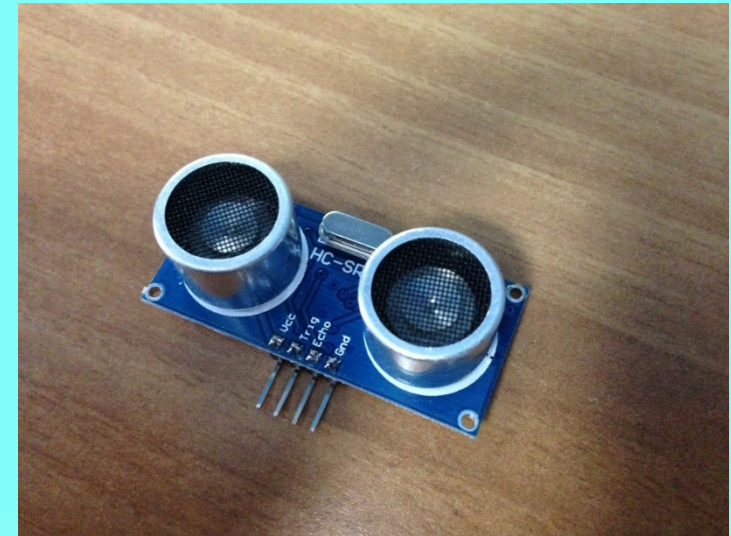
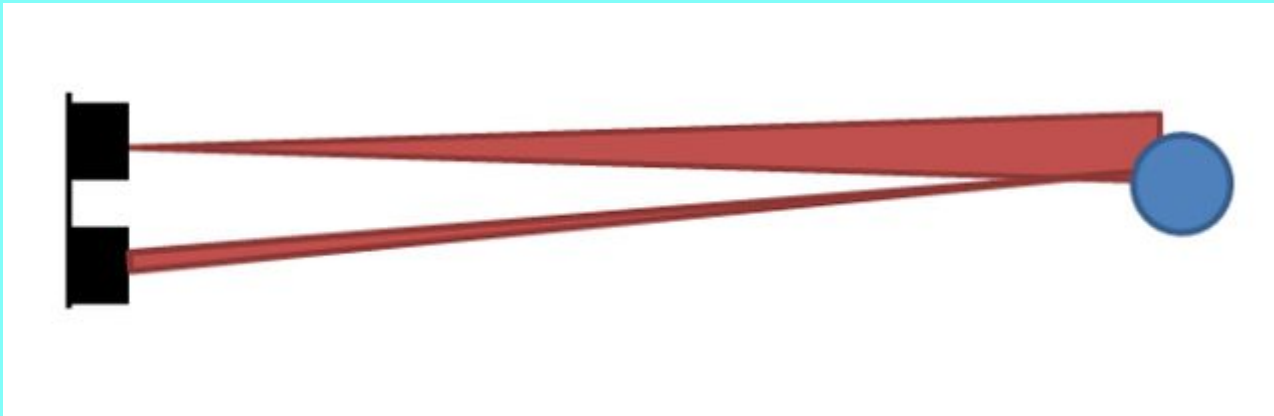
HC-SR-04 per  
rilevazione di  
vuoto

Batteria 12V  
per motori



# La navigazione tramite sensori ad ultrasuoni: modulo HC-SR-04

Un sensore a ultrasuoni come il **HC-SR-04** misura il tempo impiegato dalle onde sonore emesse da una sorgente a ritornare dopo aver incontrato un ostacolo che le riflette.



Per fare il calcolo della distanza la formula base è:

$$\text{Distanza} = \text{tempo} * \text{velocità del suono} / 2 = \text{tempo} * 3,4 * 10^{-2} / 2$$

# Algoritmo di Obstacle Avoidance

## Robot

- speedFW:float  
- speedBW:float

//pin dedicati al driver dei motori

- EN\_DX:int  
- in1,in2:int

- obstacleSX:boolean = "false"  
- obstacleDX:boolean = "false"  
- obstacleDOWN:boolean = "false"  
- distance:int

+ Robot()  
+ setPin():void  
+ forward():void  
+ backward():void  
+ left():void  
+ right():void  
+ stop\_robot():void  
+ auto\_mode():void

```
if ((!obstacleSX) && (!obstacleDX)) {  
    forward();  
}  
  
if (sonarSX.ping_cm() <= 15) {  
    obstacleSX = true;  
    obstacleDX = false;  
}  
else {  
    obstacleSX = false;  
}  
if (sonarDX.ping_cm() <= 15) {  
    obstacleSX = false;  
    obstacleDX = true;  
}  
else {  
    obstacleDX = false;  
}  
if (sonar.ping_cm() <= 20) {  
    if ((sonarDX.ping_cm()) <= (sonarSX.ping_cm())) {  
        obstacleDX = true;  
        obstacleSX = false;  
    }  
    else {  
        obstacleSX = true;  
        obstacleDX = false;  
    }  
}  
  
if (obstacleSX) {  
    right();  
}  
if (obstacleDX) {  
    left();  
}
```

# Conclusione

*Quello che è stato realizzato è solo a scopo dimostrativo, poiché l'hardware utilizzato, se pur funzionante ed efficiente, non è consono ad uno sviluppo sperimentale e il software si basa su un semplice algoritmo che permette al veicolo di evitare gli ostacoli. Questo progetto è, d'altra parte, utile poiché rappresenta il punto di partenza per la sperimentazione dei metodi di navigazione e degli algoritmi affrontati nei precedenti capitoli, con lo scopo futuro di realizzare un algoritmo combinato di localizzazione e navigazione autonoma, basandosi sugli studi e sui dati raccolti dal comportamento del robot con vari metodi già esistenti.*