

Squadra “Poenta e Osei”



Da sinistra verso destra: Nicola Braceschi, Marco Carollo, prof. Stefano Andriolo, Riccardo Zenere

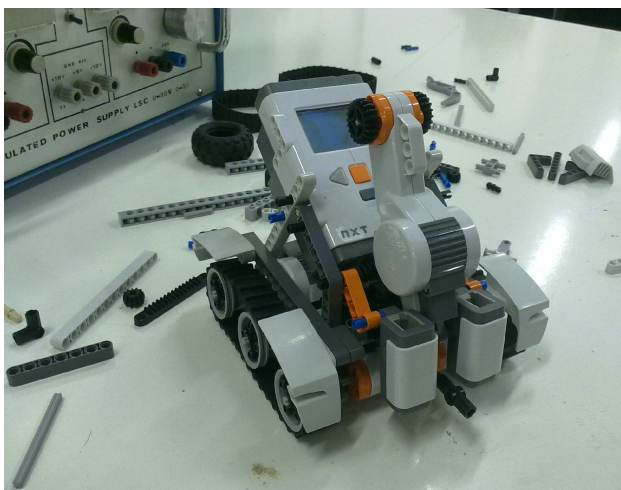
Dati Generali

Membri della Squadra “Poenta e Osei”:

- Braceschi Nicola: studente frequentante la classe 3° BT dell'ITIS Alessandro Rossi, è colui che si è dedicato con maggior impegno alla parte di programmazione. L'idea iniziale dell'uso di soli due sensori d'ultrasuoni, per la ricerca della lattina, al posto di un sensore d'ultrasuoni con un sensore di luce è sua.
- Carollo Marco: studente frequentante la classe 3° AT dell'ITIS Alessandro Rossi, è colui che si è dedicato alla costruzione effettiva del robot e colui che ha avuto l'idea definitiva per il rilascio della lattina nella *Zona Rossa*.
- Zenere Riccardo: studente frequentante la classe 3° AT dell'ITIS Alessandro Rossi, è colui che si è occupato della risoluzione di parte dei problemi sia meccanici che a livello di software. Si è occupato della battitura della documentazione.

Dati di Contesto e Motivazione

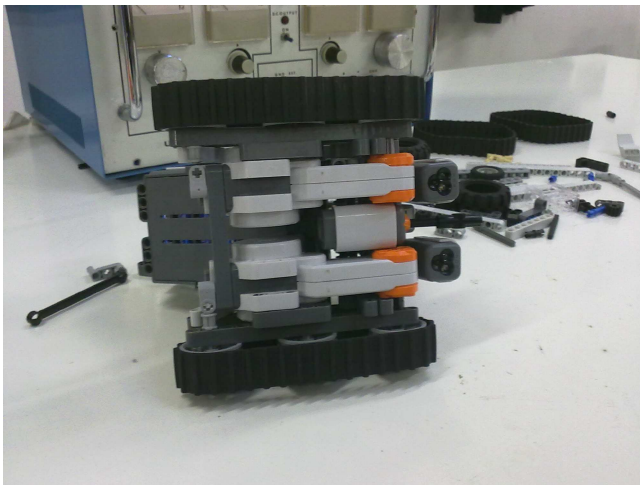
Il progetto di prender parte alla RoboCup Jr iniziò nell'edizione precedente, che venne ospitato nella città di Vicenza. Grazie al fatto che il nostro Istituto in quel momento era a capo della Rete delle scuole che prendevano parte al progetto italiano della RoboCup, abbiamo avuto modo di vedere da vicino in prima persona ciò di cui si trattava, in particolare gli oneri che implicava prendervi parte quali *gareggianti*, oltre che ovviamente anche la soddisfazione di riuscire a *strappare* un buon posto in classifica agli avversari.



Una delle varianti iniziali del robot

L'effettiva idea di prendervi parte tuttavia ci venne lanciata solo nel novembre dello scorso anno quando ci riunimmo al pomeriggio coi docenti, che c'avrebbero in seguito accompagnato in quel di Catania, che ci diedero un'infarinatura bonaria delle varie proposte date per prender parte alla RoboCup {N.B. Il gruppo inizialmente aveva un altro componente al posto di Nicola}.

Gli incontri per lavorare al progetto si svolgevano una sola volta a settimana, il venerdì, dalle 13.30 fino alle 16.30-17 circa.



Fase di sviluppo del robot riportato nella foto precedente

Inizialmente la prendemmo abbastanza alla buona, facendo solo svariate prove col robottino... Ma quando il compagno di squadra che venne poi sostituito da Nicola verso gennaio abbandonò, decidemmo di iniziare seriamente ad *ingranare*. Fino a circa metà febbraio, nonostante chiedessimo insistentemente di poter usare più volte alla settimana i locali scolastici { che c'erano preclusi data la mancanza di docenti disposti a sorvegliarci } continuammo a lavorare solo una volta la settimana anche sapendo che la Scuola avrebbe acconsentito a portare alle gare di Catania solo una squadra delle tre che usavano i Lego Mindstorm NXT. Decidemmo dunque di usare tutti i momenti possibili che ci venivano concessi dall'orario, quali le ore di supplenza ed approfittammo anche di ulteriori ore seguenti agli incontri settimanali del pomeriggio. I lavori sul robottino continuavano, leggermente più a rilento a

livello di software rispetto di quelli “meccanici” ma senza particolari preoccupazioni, salvo il fatto che ogni volta per una motivazione o per un'altra volevamo modificare completamente la forma o qualche particolare del robottino {in particolare la parte della locomozione mediante ausilio di cingoli, che venivano molto a dipendere dalle condizioni della pista}.

All'arrivo di marzo e dunque della conseguenza selezione interna alla scuola che si sarebbe tenuta sabato 26 marzo nell'occasione di una gara amichevole con altre scuole della Regione, tecnicamente ci saremmo dovuti ritrovare ancora solo una volta alla settimana. Peccato per il fatto che siamo arrivati a ritrovarci dai 2 ai 3 pomeriggi alla settimana nei locali scolastici e non con la presenza dei docenti.

Questo fu il periodo nel quale ottenemmo i maggiori e più importanti risultati che dovevamo ottenere per giungere a Catania.

Al 23 marzo eravamo a buon punto, ma non ancora pronti per affrontare la gara. Tra le tre squadre, noi eravamo quelli messi “*così così*”. La squadra della BT {la classe “*concorrente*” alla AT} non era ancora riuscita a far seguire la linea al robot anche se riusciva a farle prendere la lattina ed a metterla in un angolo {sempre e solo un angolo fisso della piattaforma}.

Non eravamo ancora riusciti a far trovare la piattaforma al robot, mentre per trovare la lattina e seguire la linea non v'era problema d'alcun genere: *filava tutto liscio come l'olio*.

Nonostante ciò, non avevamo tempo da perdere dunque optammo per un cambio rapido e necessario dell'idea del solo uso di 2 sensori di ultrasuoni con quella d'usare {come facevano nostro malgrado anche gli altri gruppi} un sensore di ultrasuoni ed uno di luce. Il giorno seguente ci ritrovammo nuovamente, ma non producemmo molto, quindi dovevamo risolvere questo abnorme problema entro venerdì.

Nostro malgrado, nel pomeriggio di venerdì non risolvemmo

tutti i problemi che avevamo nel trovare la lattina con soli due sensori, così ci mettemmo a lavorare durante la nottata tra venerdì e sabato fino alle 2:30 circa, quando il problema era risolto.



Assiduo lavoro notturno per risolvere i problemi che aveva nella presa della lattina

La mattina seguente, strano a crederci, eravamo a scuola pimpanti e pronti per la gara. Fin dalle 7:45, orario d'inizio delle lezioni alle quali non prendemmo parte grazie al fatto che dovevamo partecipare alle gare/selezioni, provammo e riprovammo il robot apportando piccoli accorgimenti al programma in maniera tale che fosse il più funzionante possibile durante la gara.



Foto del Robot prima della gara

Passate da poco le 8, i portatili di Nicola e Marco iniziarono entrambi a fare le bizzes non riconoscendo le chiavette ed i mouse che usavano le porte Usb. Dopo esser riusciti a copiare il programma in un'ulteriore chiavetta, ci spostammo in un laboratorio in disuso e ne assediavamo un PC per poter continuare a lavorare {sì, c'è toccato correre tra il laboratorio e la palestra dove v'erano le piste per la gara per verificare cosa modificare}.

Alla fine, iniziò la gara dopo un breve benvenuto alle squadre da parte dei professori dell'Istituto.

Poco prima della nostra prima gara {per fortuna eravamo in fondo all'elenco} riuscimmo a far ripartire uno dei due portatili.

Come abbiamo avuto modo di constatare coi nostri occhi durante la prima gara, il programma era perfetto salvo che in un punto: la presa della lattina. Quando in circa 1 minuto il robot era giunto alla lattina, nell'azione di sollevarla aveva chiuso troppo tardi la morsa lasciandosela sfuggire.

Riprovammo una sola volta a vedere se il problema si ripeteva, dopodiché decidemmo di non sprecare altri punti: eravamo comunque primi tra le squadre del Rossi per punteggio ottenuto {non per tempo dato che c'era stato come alle altre due squadre il tempo massimo di 8 minuti per non aver completato la pista fino in fondo}.

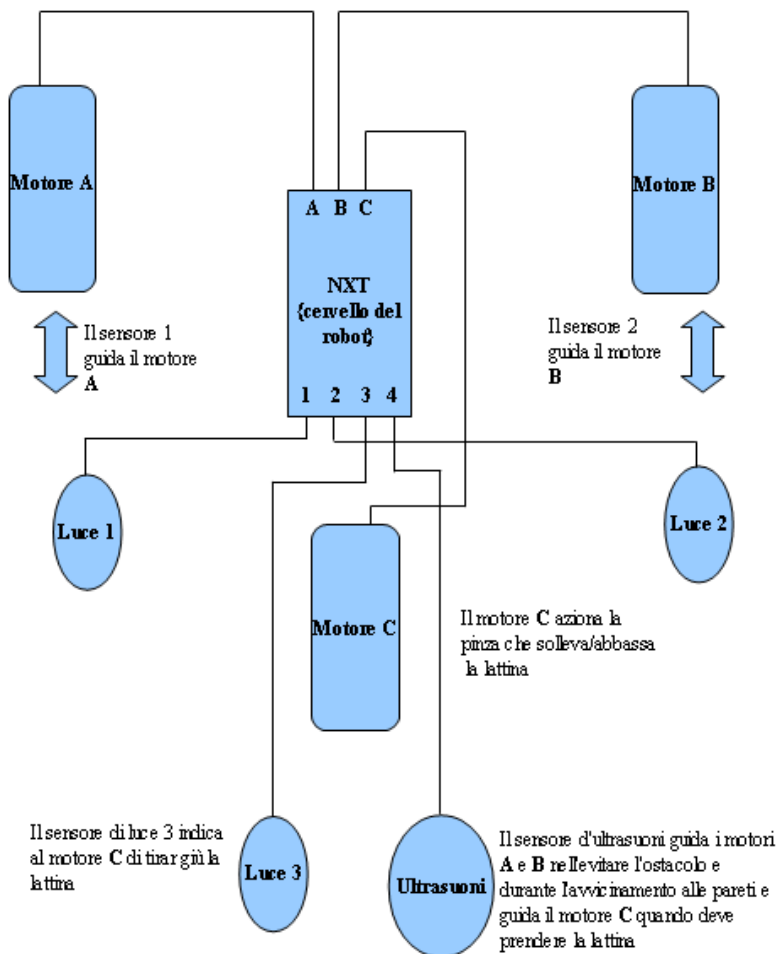
Prima della nostra seconda gara avevamo risolto il problema, facendo sì che le lattine venissero alzate senz'alcun problema. Pareva tutto perfetto durante la seconda gara, quando durante la salita il robot s'è ribaltato sulla schiena percorrendo tutto il tratto appena fatto in discesa. Nonostante questo, arrivò ugualmente in cima alla pista e riuscì a completarla ottenendo quasi il punteggio massimo: eravamo ancora i primi in classifica rispetto al resto dell'Istituto.

Tuttavia, erano arrivate già le 12 e le altre squadre non potevano tardare ancora molto il rientro, dunque si decise che per loro le gare si sarebbero concluse con sole due prove. Mentre si congedavano, a livello d'Istituto decidemmo di completare la 3° pista mancante per essere equi con chi aveva fatto una pista rispetto all'altra {ma, soprattutto, non potevamo permetterci di attendere oltre per decidere quale fosse la squadra che adoperava un robot Lego da mandare a Catania dato che non avrebbe avuto tempo e modo di completare la documentazione}. Nonostante le previsioni nefaste {ci mancava la pista più difficile tra le 3 che erano state usate durante le gare}, ottenemmo un rispettoso punteggio di 195 qualificandoci quindi per andare alle gare di Catania.

Nome e Struttura del Robot

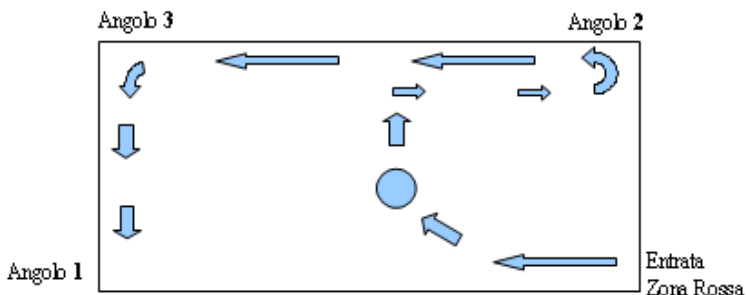
Il robot, che prende il nome di *Fascette* per l'elevato contenuto delle stesce tra le strutture al solo fine di solidificare la struttura, funziona con questi componenti/sensori: 1 “cervello” Nxt, 3 sensori di luce ed un sensore di ultrasuoni.

Schema di funzionamento:



Procedimento “teorico”:

I sensori di luce 1 e 2 fanno seguire ai motori A e B la linea, mentre in contemporanea il sensore d'ultrasuoni rileva eventuali ostacoli davanti. Quando e se li dovesse rilevare, il sensore provvederà a far girare inizialmente verso destra il robot che provvederà ad effettuare una rilevazione dello spazio disponibile sempre mediante il sensore di ultrasuoni. Nel caso in cui la rilevazione desse esito negativo {spazio inferiore a quello necessario che abbiamo impostato} il robot girerà su se stesso di 180° in maniera tale da esser direzionato in maniera opposta rispetto a prima {in questo caso il robot prosegue ugualmente secondo la curva impostata}. Sono sempre i sensori di luce 1 e 2 ad indicare quando iniziare la parte di programma della *Zona Rossa*. Mediante l'ausilio del sensore d'ultrasuoni, il robot procede nel rilevare la lattina proseguendo nel suo percorso per poi rilevare 4 volte in totale se alla sua destra v'è la lattina. Dopo averla trovata e sollevata con l'apposita pinza, percorrerà sempre il percorso che va da quello che abbiamo definito come angolo 2 per poi girare verso l'angolo 3 ed infine tornare verso l'angolo 1. La lattina viene rilasciata solamente quando il sensore di luce 3, posto davanti, rileverà il nero della piattaforma. Dopodiché, il programma termina col robot che indietreggia così da lasciar cadere meglio la lattina nel caso in cui ciò non fosse avvenuto.



Meccanica

Per la realizzazione del robot sono stati adoperati solamente pezzi della linea Lego Technic, opportunamente sistemati con qualche taglio lì dove serviva oppure fissati mediante l'uso di fascette, così che rimanessero più uniti tra di loro i pezzi.

Per la locomozione sono stati usati dei cingoli sempre appartenenti alla linea Lego Technic, che sono apparsi i più idonei, veloci nella maggior parte del circuito e meno soggetti a sobbalzi dovuti a ostacoli presenti nella pista.



Dettaglio di uno dei cingoli

Per migliorare la presa della pinza sono stati usati dei pezzi semi-gommosi della linea Lego Technic che hanno molto attrito con la lattina, così da riuscire a sollevarla con facilità. Sulla parte superiore della pinza, sono presenti due pezzi leggermente ricurvi ai quali si aggiunge un'asta che blocca i movimenti della lattina quando questa è stata sollevata. Sempre al fine di mantenere nella posizione verticale la lattina, sono stati aggiunti due “corni” nella parte superiore del robottino.



Dettaglio della pinza senza l'asta superiore

Nota: tutti e tre i motori usati sono montati in senso contrario rispetto a quello che prevede il programma. Per questo, nel programma ritroveremo principalmente la voce “OnRev” {comando per far andare indietro il motore} al posto di “OnFwd” {comando per far andare avanti il motore}. E' stato inoltre usato un contrappeso composto da una batteria da 9V posta sulla parte posteriore del robot, al fine di equilibrarla.

Unità di Controllo

Per comandare il robot s'è scelto d'adoperare il sistema NXT, dotato di 3 uscite per i motori, 4 per i sensori ed un uscita Usb {per interfacciarsi con un PC} e di Bluetooth, usato per interfacciare il robot con altri dispositivi NXT o con altri dispositivi dotati di sistema Bluetooth. E' alimentato da una batteria ricaricabile già in dotazione con l'NXT da 7,4V oppure con 6 batterie AA da 1,5V ciascuna. L'Nxt usato adopera Firmware 1.28, rilasciato direttamente dalla Lego. E' possibile inserire il programma creato e precedentemente controllato collegando l'Nxt al PC e inserendogli il programma mediante l'ausilio del programma Bricx Command Center.

Sensori

Come già spiegato in precedenza nel punto riguardante il funzionamento del robot, quest'ultimo usa tre sensori di luce ed uno d'ultrasuoni prodotti sempre dalla Lego.

I sensori di luce 1 e 2 svolgono solamente il compito di far seguire la linea al robot, mentre il sensore di luce 3 serve per rilevare il colore scuro della piattaforma d'arrivo per far lasciare cadere la pinza con la lattina stretta in precedenza tra le sue morsa.

Il sensore d'ultrasuoni si occupa invece di evitare gli ostacoli e di localizzare nella Zona Rossa la lattina.

Ambiente di Sviluppo

Per programmare l'NXT, al posto d'usare il programma Lego allegato, s'è scelto d'usare il linguaggio di scrittura Bricx, corretto con apposito programma Bricx Command Center {versione 3.3 (Build 3.3.8.8)} che risultava ben più libero rispetto a quanto poteva offrire il normale programma Lego in dotazione.

La scelta deriva anche e principalmente dal fatto che già in precedenza c'era stata data occasione di provare il programma ufficiale Lego in ambito di un breve progetto mattutino, in cui ci trovammo molto svantaggiati rispetto a come ci siamo trovati con questo linguaggio.

Il Programma Software

Quello che segue è il programma, nell'ultima versione creata fino al 30 marzo.

In corsivo a lato sono riportate le spiegazioni solamente delle parti ritenuti più rilevanti.

N.B. Il robot inizierà con il braccio-pinza alzato.

Programma	Commenti
#define luce 46	<i>Definizione variabile "luce"</i>
task main()	<i>Task principale</i>
{SetSensorLight(IN_3); SetSensorLight(IN_1); SetSensorLight(IN_2); SetSensorLowspeed(IN_4); int ciclo=0;	<i>Dichiarazione dei sensori ed inserimento della variabile "ciclo"</i>
do{ OnRev(OUT_AB,100);	<i>Ciclo base per seguire la linea</i>
if(Sensor(IN_2) < luce) { Off(OUT_AB); OnFwd(OUT_B,100); OnRev(OUT_A,100); Wait(150); Off(OUT_AB); }	<i>Struttura per fare in modo che il robot ritorni a seguire la linea nel caso in cui il sensore 2 dovesse rilevare una luminosità inferiore a quella definita inizialmente dalla variabile "luce"</i>
if(Sensor(IN_1)< luce) { Off(OUT_AB); OnFwd(OUT_A,100); OnRev(OUT_B,100); Wait(150); Off(OUT_AB);	<i>Struttura per fare in modo che il robot ritorni a seguire la linea nel caso in cui il sensore 1 dovesse rilevare una luminosità inferiore a quella definita inizialmente dalla variabile "luce"</i>

<pre> } if(SensorUS(IN_4)<7) { PlayTone(4000,50); Off(OUT_AB); OnFwd(OUT_AB,100); Wait(600); Off(OUT_AB); OnFwd(OUT_B,100); OnRev(OUT_A,100); Wait(800); Off(OUT_AB); if(SensorUS(IN_4)>35) { Off(OUT_AB); OnRev(OUT_AB,100); Wait(300); Off(OUT_AB); OnRev(OUT_A,35); OnRev(OUT_B,100); Wait(5000); do{ OnRev(OUT_AB,100); }while(Sensor(IN_2)> luce); Off(OUT_AB); OnFwd(OUT_B,100); OnRev(OUT_A,100); Wait(700); } else { Off(OUT_AB); OnFwd(OUT_A,100); </pre>	<hr/> <p><i>{Spiegazione del processo col quale dovrà evitare l'ostacolo e riprendere a seguire la linea. Il programma si estende dalla linea sovrastante fino alla prossima linea di demarcazione}</i></p> <p><i>Struttura per far evitare l'ostacolo nel caso in cui il sensore 4 rilevi una distanza inferiore ai 7cm. Il robot procederà a tornare indietro ed a ruotare su se stesso prima verso destra e successivamente verso sinistra per rilevare un'eventuale distanza superiore ai 35cm che gli permetterebbero di eseguire la manovra di superamento dell'ostacolo. Nel caso in cui dovesse rilevare la linea {solo col sensore 2} mentre sta eseguendo la manovra, procederà a seguirla nuovamente.</i></p>
---	--

```
OnRev(OUT_B,100);
Wait(1850);
Off(OUT_AB);
OnRev(OUT_B,35);
OnRev(OUT_A,100);
Wait(5000);
do{
OnRev(OUT_AB,100);
}while(Sensor(IN_1)> luce);
Off(OUT_AB);
OnFwd(OUT_A,100);
OnRev(OUT_B,100);
Wait(700);
}
}
if(Sensor(IN_1)>60)
{ciclo=1;
Off(OUT_AB);}
} while(ciclo==0);
```

```
if(Sensor(IN_1)>60)
{PlayTone(4000,50);}
Wait(500);
```

```
OnFwd(OUT_AB,100);
Wait(300);
Off(OUT_AB);
```

```
OnFwd(OUT_AB,100);
Wait(700);
Off(OUT_AB);
```

```
OnRev(OUT_AB,100);
Wait(1200);
```

*Rilevamento della striscia
argentata da parte del solo
sensore 1 alla fine della salita.*

*Entrambi i motori manderanno
indietro il robot in maniera tale
che vada ad allinearsi con la
parete dietro stante per poi
entrare nella Zona Rossa, tirare
giù il braccio {che per tutto il
percorso sarà rimasto alzato} ed
iniziare a cercare la lattina.*

```
Off(OUT_AB);
```

```
OnRev(OUT_C,100);
```

```
Wait(1000);
```

```
Off(OUT_C);
```

```
int scatto=0;
```

```
int posizione=0;
```

```
int aaa=0;
```

```
int blocco=0;
```

```
/*prime 2 rotazioni*/
```

```
repeat(2)
```

```
{
```

```
if(scatto<3)
```

```
{
```

```
OnRev(OUT_AB,100);
```

```
Wait(600);
```

```
repeat(18)
```

```
{
```

```
if(scatto<3)
```

```
{
```

```
if(SensorUS(IN_4)<65)
```

```
{scatto=scatto+1;posizione=1;
```

```
PlayTone(4000,50);}
```

```
else{ scatto=0; }
```

```
OnRev(OUT_A,100);
```

```
OnFwd(OUT_B,100);
```

```
Wait(50);
```

```
Off(OUT_AB);
```

```
}
```

```
}
```

```
if(scatto<3)
```

```
{
```

Abbassamento del braccio.

*Dichiarazione nuove variabili
per la ricerca della lattina.*

*Primi due tentativi per
individuare la lattina {rotazione
in senso orario e prosecuzione in
avanti dopo esser tornati alla
posizione "originale"}.*


```
OnRev(OUT_B,100);
OnFwd(OUT_A,100);
Wait(800);
Off(OUT_AB);
}
}
}
```

```
/*3^rotazione*/
if(scatto<3)
{
  repeat(1)
  {
    OnRev(OUT_AB,100);
    Wait(600);

  repeat(18)
  {
    if(scatto<3)
    {
      if(SensorUS(IN_4)<60)
      { scatto=scatto+1;posizione=2;
      PlayTone(4000,50);}
      else{scatto=0;}

    OnRev(OUT_A,100);
    OnFwd(OUT_B,100);
    Wait(50);
    Off(OUT_AB);
    }
    }
  if(scatto<3)
  {
    OnRev(OUT_B,100);
    OnFwd(OUT_A,100);
    Wait(800);
```

Terzo tentativo per trovare la lattina seguendo la medesima procedura usata in precedenza.

```

Off(OUT_AB);
}
}
}

/*4^rotazione*/
if(scatto<3)
{
repeat(1)
{
OnRev(OUT_AB,100);
Wait(600);

repeat(9)
{
if(scatto<3)
{
if(SensorUS(IN_4)<50)
{scatto=scatto+1;posizione=3;Pl
ayTone(4000,50);}
else{scatto=0;}

OnRev(OUT_A,100);
OnFwd(OUT_B,100);
Wait(50);
Off(OUT_AB);
}
}
}
if(scatto<3)
{
repeat(6)
{
if(scatto<3)
{
if(SensorUS(IN_4)<70)
{scatto=scatto+1;posizione=4;
PlayTone(4000,50);}
}
}
}
}

```

Quarto ed ultimo tentativo di individuare la lattina.

```

else{scatto=0;}

OnRev(OUT_A,100);
OnFwd(OUT_B,100);
Wait(50);
Off(OUT_AB);
}}}}

/*****/
if(posizione==1)
{
while(scatto<4){
if(SensorUS(IN_4)<65)
{
PlayTone(4000,50);
scatto=scatto+1;
aaa=aaa+50;}
else
{scatto=0;}
if (scatto<4)
{
OnRev(OUT_A,100);
OnFwd(OUT_B,100);
Wait(50);
Off(OUT_AB);
Wait(50);
}}}

if(posizione==2)
{
while(scatto<6){
if(SensorUS(IN_4)<60)
{
PlayTone(4000,50);
scatto=scatto+1;
aaa=aaa+50;}
else

```

Procedimento per avvicinarsi alla piattaforma d'arrivo.

```
{ scatto=0;}
if (scatto<6)
{
OnRev(OUT_A,100);
OnFwd(OUT_B,100);
Wait(50);
Off(OUT_AB);
Wait(50);
}}

if(posizione==3)
{
while(scatto<4){
if(SensorUS(IN_4)<50)
{
PlayTone(4000,50);
scatto=scatto+1;
aaa=aaa+50;}
else
{ scatto=0;}
if (scatto<4)
{
OnRev(OUT_A,100);
OnFwd(OUT_B,100);
Wait(50);
Off(OUT_AB);
Wait(50);
}}}

if(posizione==4)
{
while(scatto<4){
if(SensorUS(IN_4)<60)
{
PlayTone(4000,50);
scatto=scatto+1;
aaa=aaa+50;}
```

```

else
{scatto=0;}
if (scatto<4)
{
OnRev(OUT_A,100);
OnFwd(OUT_B,100);
Wait(50);
Off(OUT_AB);
Wait(50);
}}

/*****/
do{
OnRev(OUT_AB,100);
Wait(50);
if(SensorUS(IN_4)<7)
{blocco=1;}
}while(blocco==0);

Off(OUT_AB);
OnRev(OUT_AB,100);
Wait(500);
Off(OUT_AB);
OnFwdReg(OUT_C,100,OUT_R
EGMODE_SPEED);
Wait(1500);
Off(OUT_AB);
OnFwd(OUT_C,15);
Wait(1000);
/*****/

OnRev(OUT_A,100);
OnFwd(OUT_B,100);
Wait(350);

do{
OnRev(OUT_AB,100);

```

```
Wait(50);  
}while(SensorUS(IN_4)>7);
```

```
OnRev(OUT_AB,100);  
Wait(1000);
```

```
OnFwd(OUT_AB,100);  
Wait(500);
```

```
OnRev(OUT_A,100);  
OnFwd(OUT_B,100);  
Wait(800);
```

```
OnRev(OUT_AB,100);  
Wait(9000);
```

```
if( Sensor(IN_3)< 30  
{
```

```
OnRev(OUT_C,50);  
Wait(500);  
OnFwd(OUT_AB,100);  
Wait(1000);
```

```
}  
else  
{
```

```
OnFwd(OUT_AB,100);  
Wait(800);
```

```
OnRev(OUT_B,100);  
OnFwd(OUT_A,100);  
Wait(1500);
```

```
OnRev(OUT_AB,100);  
Wait(9050);
```

*Primo rilevamento della
piattaforma {angolo 2 della
figura sopra illustrata}.*

```
if( Sensor(IN_3)< 30)
{
```

```
    OnRev(OUT_C,50);
    Wait(500);
    OnFwd(OUT_AB,100);
    Wait(1000);
}
else
{
```

```
    OnFwd(OUT_AB,100);
    Wait(500);
```

```
    OnRev(OUT_B,100);
    OnFwd(OUT_A,100);
    Wait(850);
```

```
    OnRev(OUT_AB,100);
    Wait(5000);
```

```
if( Sensor(IN_3)< 30)
{
```

```
    OnRev(OUT_C,50);
    Wait(500);
    OnFwd(OUT_AB,100);
    Wait(1000);
}
```

```
}
}
}
```

Secondo tentativo per la rilevazione della piattaforma.

Terzo ed ultimo tentativo per la rilevazione della piattaforma {in ogni caso, per esclusione, andrebbe a rilasciare la lattina}.