



## Robot Beginner KIT 4WD - ARDUINO UNO

*Oggi la tecnologia permette la realizzazione di un Robot in maniera molto semplice. Con il nostro kit è possibile realizzare, in maniera molto elementare e divertente, un vero robot in poco tempo. Anche se le istruzioni dichiarano che il dispositivo in oggetto non costituisce un dispositivo ludico, potete montarlo, programmarlo e collaudarlo assieme a vostro figlio, facendogli assaporare gli aspetti dell'elettronica, della robotica e della programmazione. Il tutto con grande passione e divertimento. Occorre procedere per gradi al fine di assemblare un gioiello dell'elettronica in poco tempo per regalargli una vita propria.*

### Parliamo di questo Robot

Un robot è stato sempre un sogno di grandi e piccini e oggi questo sogno può essere facilmente realizzato. Con esso, è possibile riprodurre fedelmente la volontà dell'uomo, programmandolo e insegnandogli la strada da percorrere e il comportamento da intraprendere in determinate condizioni. Il Robot Beginner KIT 4WD è un kit di

montaggio, da assemblare e da montare in modo semplice ed in poco tempo. Il suo funzionamento è garantito in circa un'ora di lavoro. Seguendo le istruzioni riportate nel presente tutorial le operazioni di assemblaggio risulteranno estremamente semplici. Il kit contiene tutto ciò che serve per portare a termine l'intera installazione. E' stato concepito, soprattutto, per chi

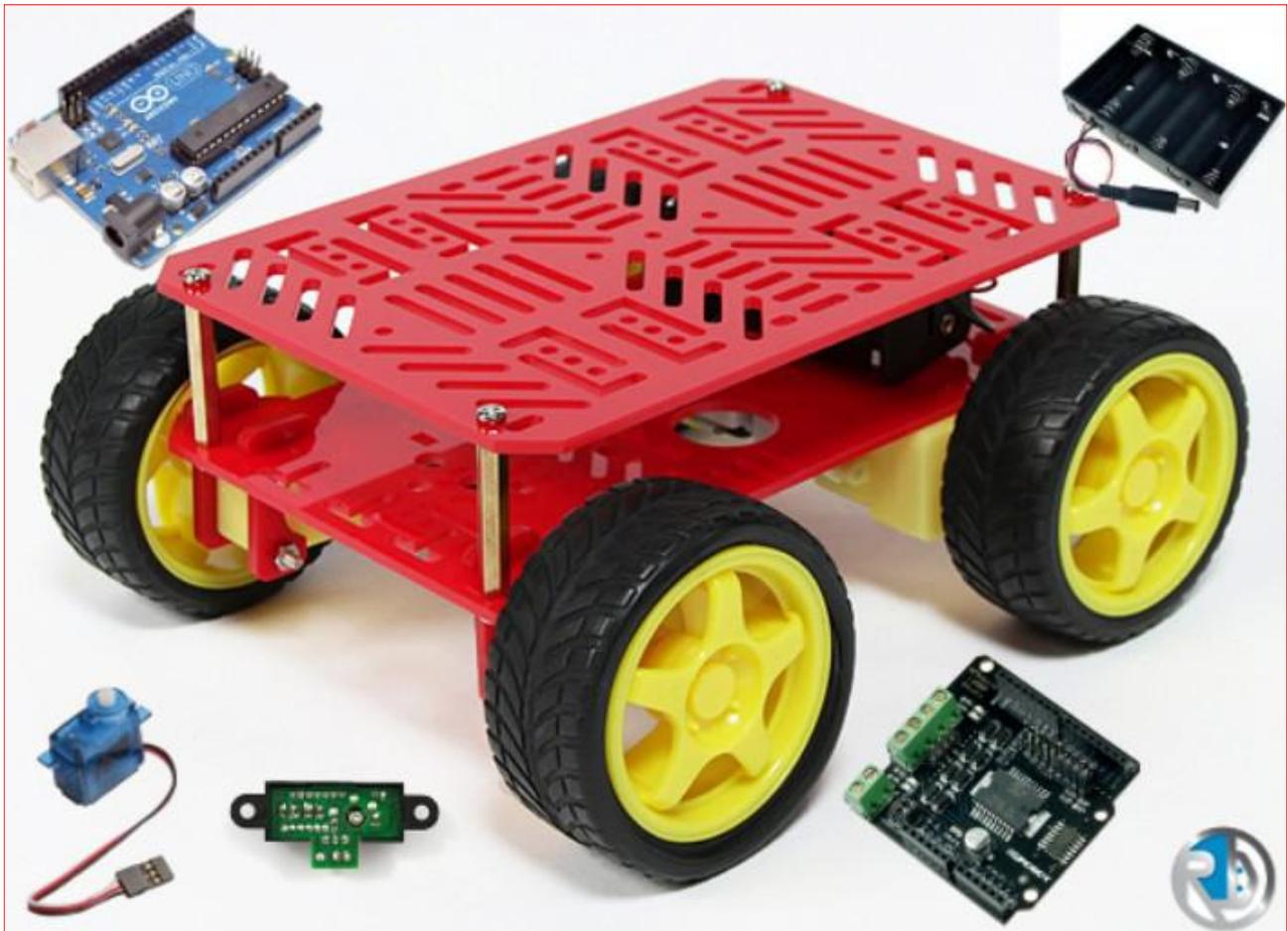


FIGURA 1: ROBOT BEGINNER KIT 4WD - ARDUINO UNO



desidera iniziare a operare subito con questa tipologia di dispositivi. La facilità di programmazione di Arduino, infatti, assieme alla duttilità del robot, consentono di dargli subito una vita propria, impartendogli i comandi più disparati e consentendogli di conoscere il mondo che lo circonda, grazie alla sensibilità del suo sensore a infrarossi.

### **Contenuto della confezione**

Al ricevimento del prodotto e all'apertura della confezione, troverete alcuni componenti sigillati e racchiusi negli appositi blister. E' indispensabile conoscere a fondo le parti che compongono il kit. Dedicate, quindi, qualche minuto per inventariare le varie componenti. Questo tempo speso in questa fase risulterà prezioso per il proseguo del corretto montaggio del robot.

### **Robot Jumper Wire Kit**

Si tratta di un insieme di utilissimi jumper finalizzati a connettere, tra loro, le varie parti elettroniche e i dispositivi. Essi evitano la noiosa e poco professionale preparazione casalinga di cavi di collegamento. Le rigide terminazioni permettono connessioni elettriche molto sicure e affidabili. Essi possono essere collocate o trasferite in pochi istanti. La fornitura comprende jumper di tipo maschio-maschio e jumper di tipo maschio-femmina. La loro lunghezza è di 6 pollici (figura 2).

### **Sensore di prossimità Sharp GP2Y0A21YK**

Si tratta di un sensore di prossimità all'infrarosso (vedi figura 3). Riesce a misurare, con sufficiente precisione, la distanza di un oggetto. Esso fornisce un'uscita analogica che aumenta al



*FIGURA 2: ROBOT JUMPER WIRE KIT*

diminuire della distanza dell'ostacolo. Il suo intervallo di funzionamento è compreso tra 3,1V a 10cm e 0,4V a 80cm. La tensione di uscita scende all'incirca a 0V quando non è presente alcun ostacolo di fronte ad esso. La sua adozione costituisce, sicuramente, una delle migliori soluzioni, a basso costo.

### **Telaio Dagu 4WD Magician Chassis**

Si tratta del telaio del robot, ossia la struttura portante del dispositivo (figura 4). Le due piastre sono dotate di numerosi fori e scanalature che consentono la più alta libertà di montaggio e di disposizione. Ovviamente è fornito in soluzione da



*FIGURA 3: SENSORE INFRAROSSO DI PROSSIMITÀ SHARP GP2Y0A21YK*



FIGURA 4: IL TELAIO DAGU 4WD MAGICIAN CHASSIS

assemblare completamente e ciò costituisce, sicuramente, un'operazione molto istruttiva e divertente. E', infatti, composto di esigue parti che si montano in pochi minuti. Il supporto di dettagliate istruzioni per il montaggio consentono un sicuro e preciso lavoro di preparazione e di assemblaggio. E' corredato da una staffa a "L" per permetterne l'alloggiamento e il fissaggio sullo chassis.

### Sub Micro Servo 9g

Si tratta di un piccolissimo servo (figura 5) dal peso di soli 9 grammi. Può essere alimentato con una tensione continua compresa tra 4,8V e 6V. La sua forza è pari a 1,5kg/cm (coppia di torsione). Il servo è corredato da alcuni supporti che ne agevolano il montaggio.



FIGURA 5: IL SUB MICRO SERVO 9G



FIGURA 6: 2A MOTOR SHIELD FOR ARDUINO

### 2A Motor Shield per Arduino

E' una scheda che costituisce il modulo di potenza, pilotando direttamente i motori del robot, per un massimo di 2 Ampere di carico. Essa utilizza l'integrato L298P, un potente driver Dual Full Bridge per motori (figura 6). Può essere direttamente montata su Arduino.

### Portabatteria per 6 pile AA

Questo porta batterie (figura 7), ospita fino a sei batterie stilo (AA). È dotato di un cavetto jack di tipo DC2 che ne consente la connessione direttamente ad Arduino.



FIGURA 7: PORTABATTERIE



FIGURA 8: DISTANZIATORI

## Distanziali

Per ultimo, sono a corredo della confezione alcuni utilissimi distanziali esagonali M3x30 (figura 8). Essi permettono il montaggio dello chassis o altra tipologia di assemblaggio meccanico. La confezione comprende:

- 10 distanziali esagonali maschio/femmina M3x30;
- 10 viti M3x6mm;

- 10 dadi M3.

## Montaggio del telaio

Procediamo, adesso, al corretto montaggio dello chassis. Esso è abbastanza semplice e si conclude in circa un'ora di lavoro. Tuttavia consigliamo di seguire le istruzioni riportate nel foglio illustrativo incluso. Tutte le dotazioni meccaniche hanno una ben precisa funzione e le varie minuterie (viti, dadi, bulloni, rondelle, ecc.) non possono essere scambiate tra loro. Approntiamo, pertanto, un inventario delle minuterie che compongono il kit del telaio (figura 9), seguendo il foglio incluso e adagiandole ordinatamente sul tavolo. Seguendo l'inventario riportato in figura, la confezione dello chassis è composta dal seguente materiale:

- N. 1 Parte superiore chassis (1);
- N. 1 Parte inferiore chassis (2);
- N. 8 Supporti per motore (3);
- N. 4 Motori (4);

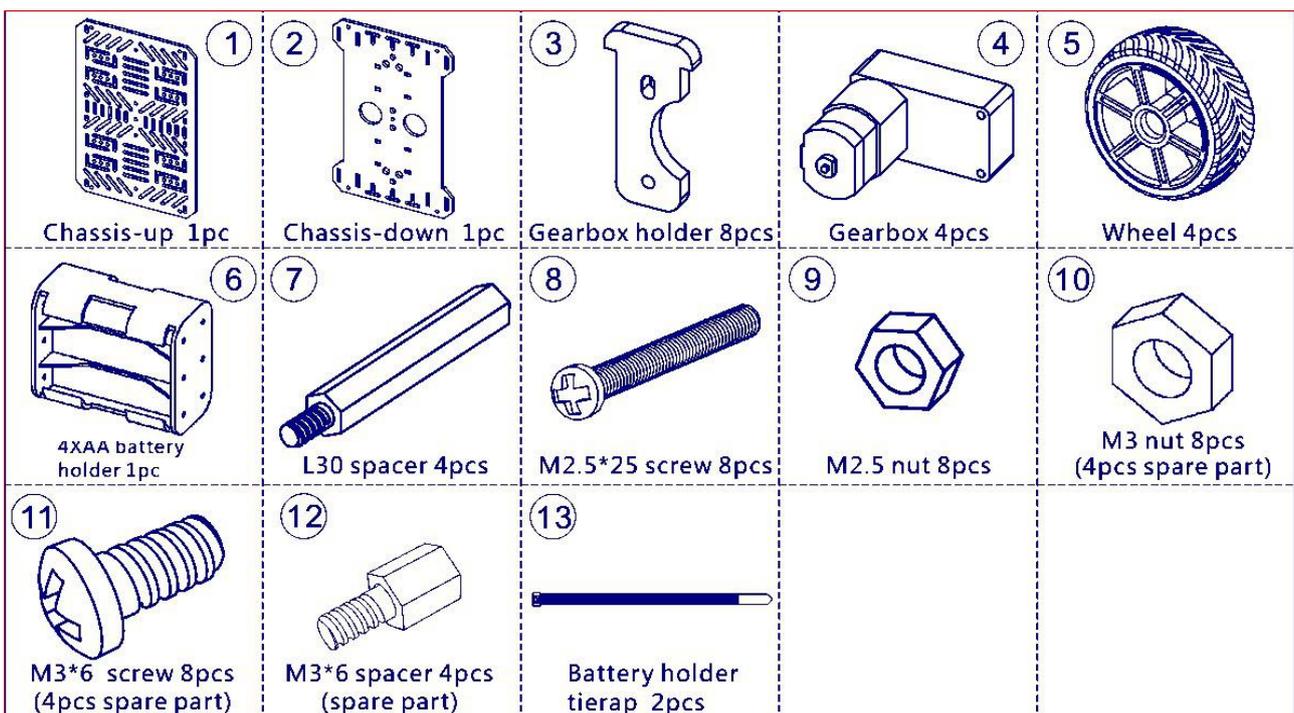


FIGURA 9: LE PARTI DELLO CHASSIS

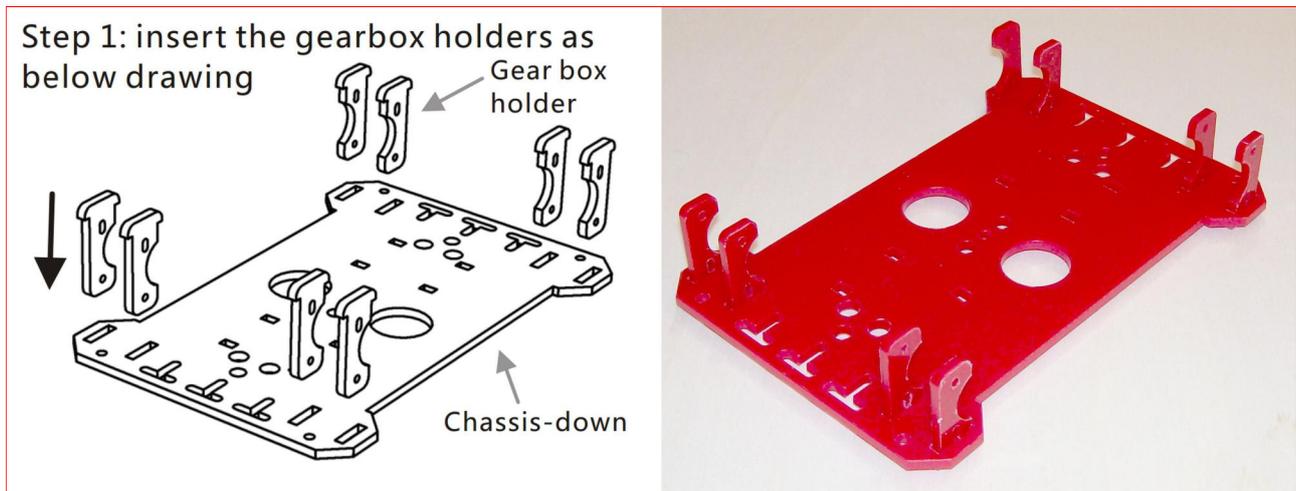


FIGURA 10: INSERIMENTO DEI SUPPORTI DEL MOTORE NELLO CHASSIS

- N. 4 Ruote (5);
- N. 1 Porta batterie (6);
- N. 4 Distanziatori L30 (7);
- N. 8 Viti M2,5x25 (8);
- N. 8 Dadi M2,5 (9);
- N. 8 Dadi M3 (4 di ricambio) (10);
- N. 8 Viti M3x6 (4 di ricambio) (11);
- N. 4 Distanziatori M3x6 (ricambio) (7);
- N. 2 Fascette per porta batterie.

Controllate con attenzione che tale lista combaci con quanto riportato nel foglio illustrativo. Procediamo, quindi, al montaggio vero e proprio, ricordando che la fretta è sempre causa di errori, a volte irreparabili. Seguiamo con attenzione e perizia, e soprattutto senza commettere errori, le istruzioni, passo per passo.

### **Passo 1: Inserimento dei supporti del motore**

Sulla parte inferiore dello chassis si devono innestare gli otto supporti dei motori (vedi figura 10). Seguendo il foglio allegato nella confezione e le nostre illustrazioni, l'operazione dovrebbe risultare molto semplice. Inizialmente, tali supporti non sono solidali alla base, fino a fissaggio completo dei motori con le viti. In effetti, più che inserimento definitivo, tale

operazione si può considerare come una preparazione dei supporti per il secondo successivo passo.

### **Passo 2: Collocazione dei motori**

In questa fase i quattro motori devono essere alloggiati, come mostrato in figura 11, in modo che i relativi fori corrispondano a quelli dei supporti.

### **Passo 3: Fissaggio dei motori**

Il terzo passo consiste nel fissaggio definitivo dei motori ai supporti dello chassis (figura 12). A disposizione avvenuta, si può procedere al bloccaggio definitivo con le viti M2,5x25 e relativi dadi. L'intera struttura è stata progettata con estrema cura e i fori, assieme agli alloggiamenti, corrispondono perfettamente. Non bisogna esercitare, in ogni caso, un'eccessiva forza nell'avvitamento. Si tratta pur sempre di supporti di plastica.

### **Passo 4: Assemblaggio delle ruote**

Il montaggio delle quattro ruote è semplice: è sufficiente incastrarle nei

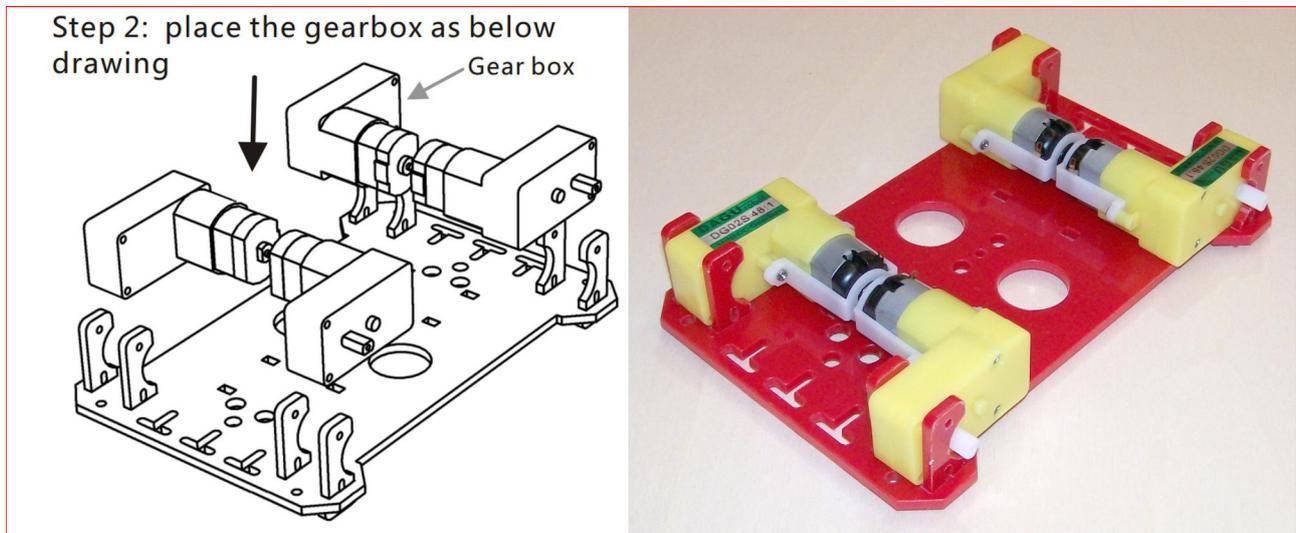


FIGURA 11: COLLOCAZIONE DEI MOTORI

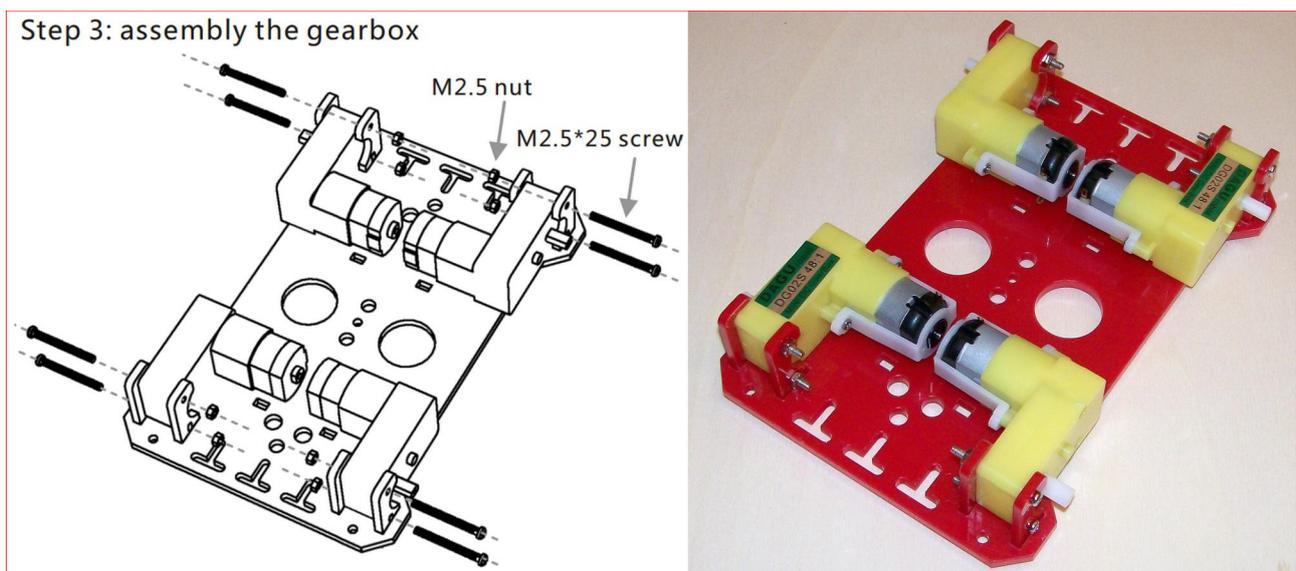


FIGURA 12: L'AVVITAMENTO DEFINITIVO DEI MOTORI SULLO CHASSIS

rispettivi alberi del motore (figura 13). Non servono viti e il fissaggio è molto sicuro per via del particolare profilo dell'innesto. Le ruote possono essere, comunque, successivamente rimosse in maniera semplice e veloce.

### Passo 5: Montaggio dello chassis superiore

Questa fase prevede il montaggio della piastra superiore dello chassis. Si deve semplicemente bloccare, con delle viti M3x6, la parte superiore dello chassis (vedi figura 14).

### Passo 6: Collocazione del vano batterie

Si procede, dunque, alla collocazione del portapile (figura 15). Esso può essere sistemato con una certa libertà in qualsiasi posizione della struttura. L'operatore ha piena autonomia, in questo senso.

### Lo chassis è pronto

Dopo le fasi preliminari sopra descritte, in circa un'ora la struttura portante del

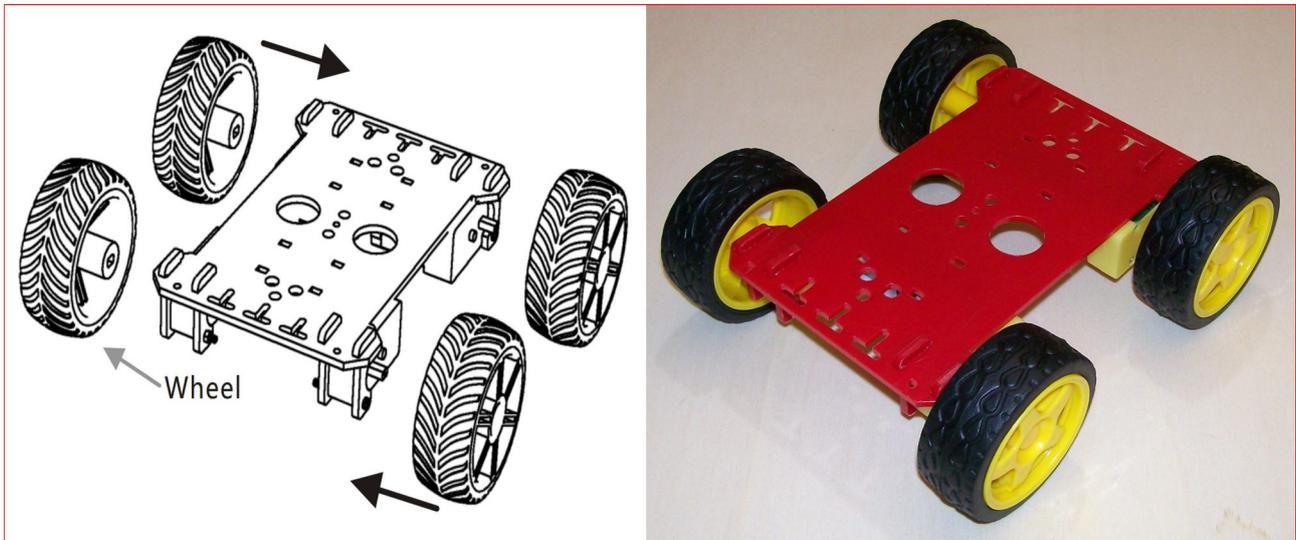


FIGURA 13: IL MONTAGGIO DELLE RUOTE

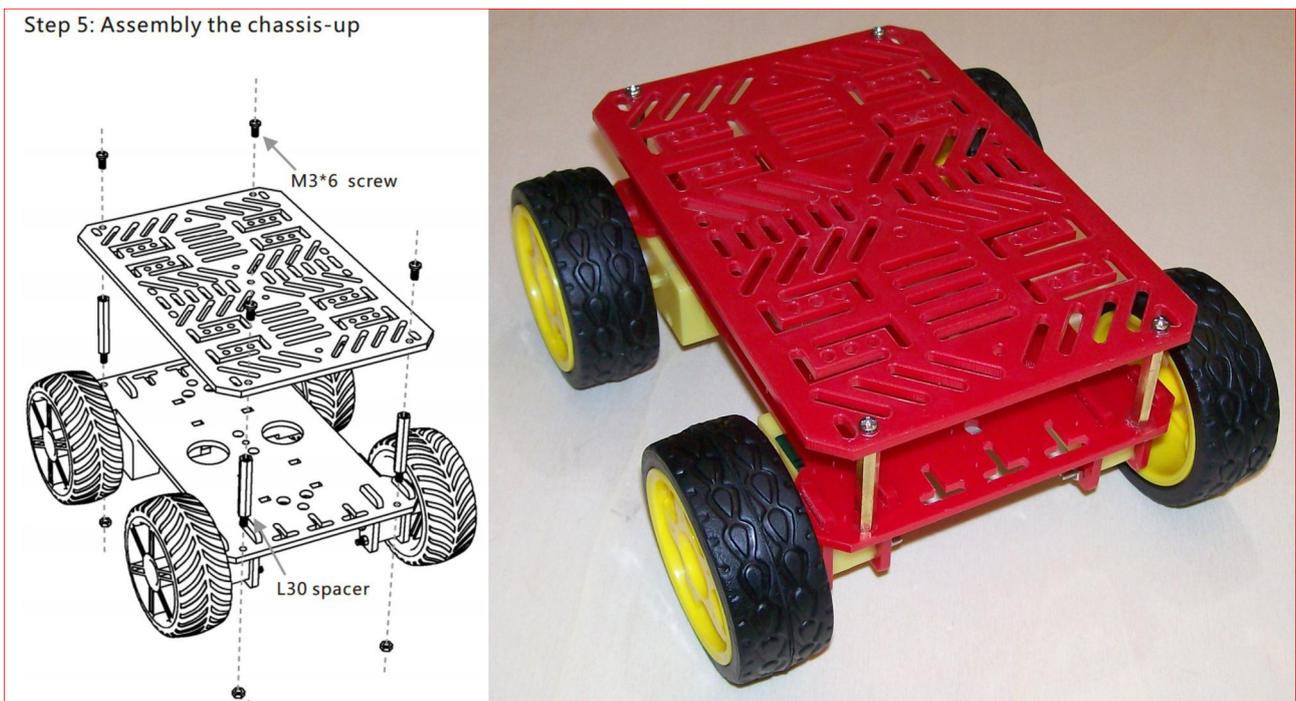


FIGURA 14: ASSEMBLAGGIO FINALE DELLO CHASSIS

robot è pronta ed è visibile in figura 16.

## Cablaggio dei quattro motori

Una volta che la struttura meccanica è pronta e lo chassis è interamente montato, occorre procedere con la saldatura dei motori. Vi sarete accorti, infatti, che essi sono sprovvisti di fili elettrici uscenti e sono presenti

solamente due contatti elettrici per ogni motore. Per tale scopo occorre munirsi di due cavetti dotati di morsetti a coccodrillo, al fine di testare la polarità dei motori (figura 17). Una volta stabilita la polarità e il senso di rotazione, con l'ausilio di una batteria di 4,5V, in pratica occorre collegare in parallelo due motori alla volta appartenenti, rispettivamente, al lato destro e sinistro dello chassis, come



Step 6: put the battery holder in place

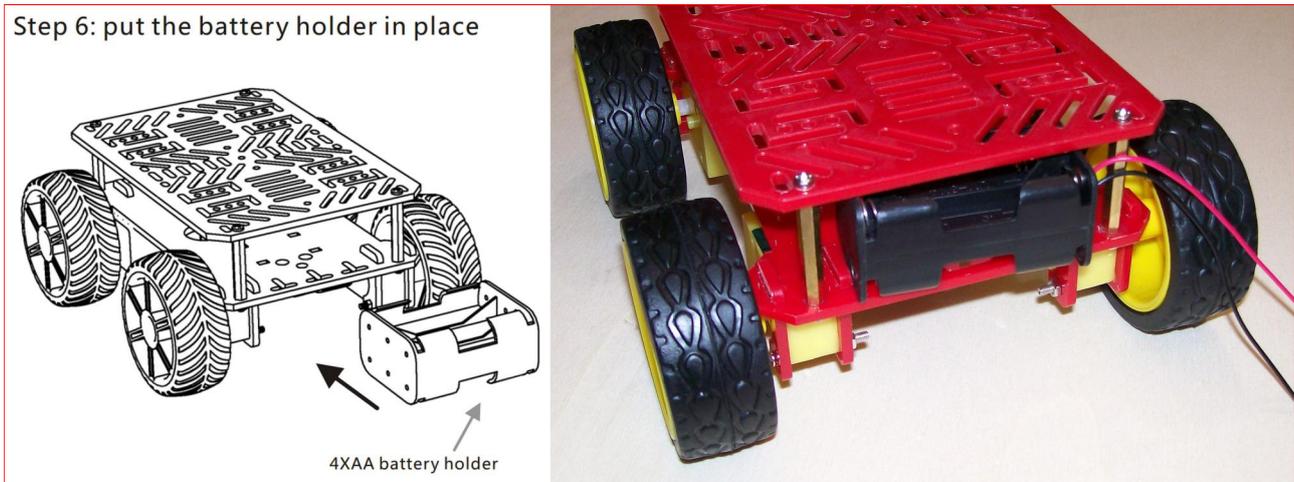


FIGURA 15: INSTALLAZIONE DEL VANO BATTERIE

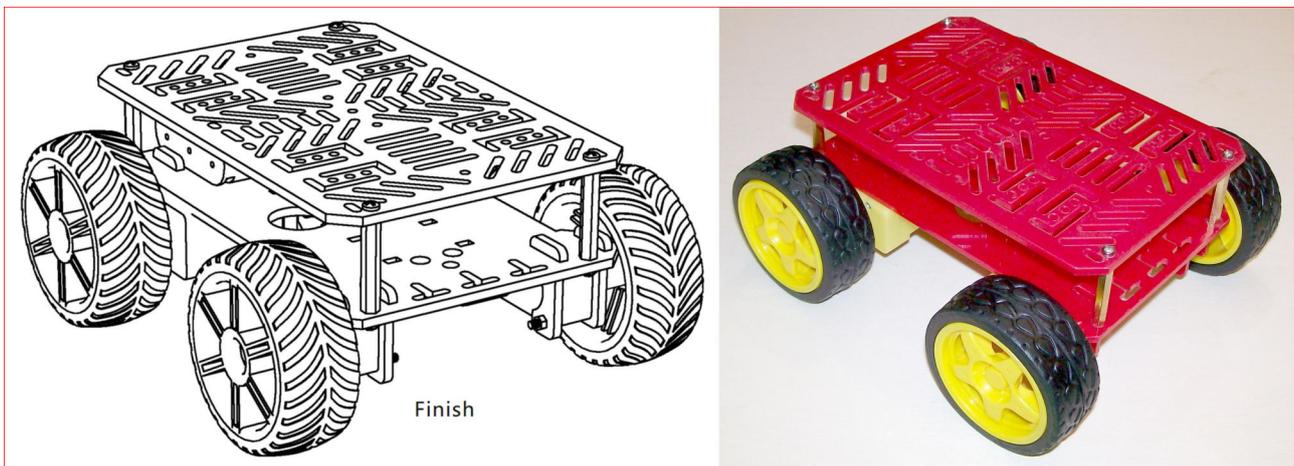


FIGURA 16: LA STRUTTURA COMPLETA

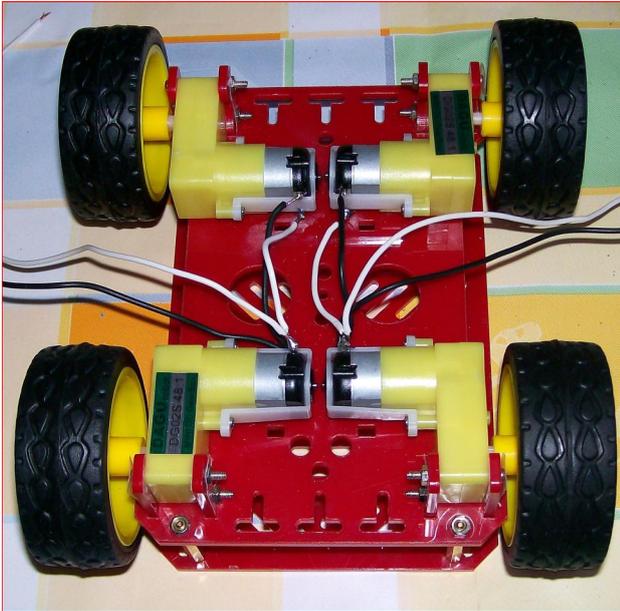
mostrato dalle illustrazioni sottostanti. L'utilizzo di filo colorato agevola, ovviamente, le operazioni d'identificazione dei contatti. Occorre, allo scopo, prestagnare i contatti dei motori (figura 18), per facilitare la brasatura dei fili di collegamento. Con un pennarello è possibile anche contrassegnare, con un puntino, il polo positivo. Si consiglia di dimensionare, almeno per adesso, una lunghezza di filo maggiore del necessario. Essa sarà, poi, ridotta successivamente all'occorrenza.

Con i distanziatori in dotazione, anche tale operazione è molto semplice. I fori presenti sullo chassis consentono una libera scelta del montaggio. Per queste operazioni non è necessario utilizzare

## Installazione di Arduino e Impilazione del Motor Shield su Arduino



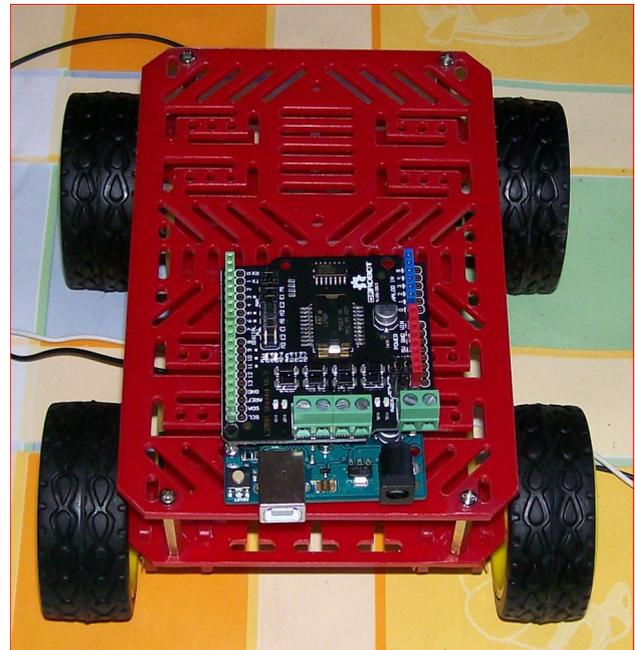
FIGURA 17: I CONTATTI DEI MOTORI



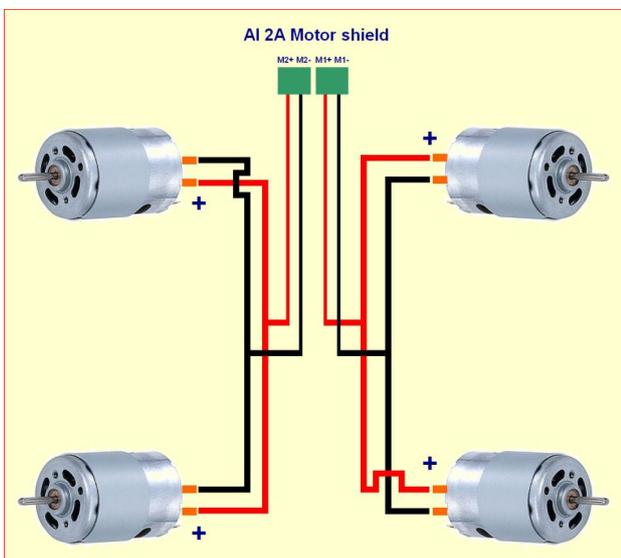
**FIGURA 18: SALDATURA DEI MOTORI**

alcun saldatore a stagno. Tutte le connessioni avvengono per avvitemento e incastro. Dopo che Arduino è stato fissato saldamente sullo chassis, si può innestare su esso il Motor Shield, impilandolo sul sistema (figura 20). La connessione pin-to-pin è molto semplice, in quanto tutti i terminali sono numerati progressivamente. In ogni caso assicuratevi di orientare i morsetti di colore verde del c.s. dello shield dei

motori verso le prese di alimentazione e USB di Arduino. Inoltre, i pettini d'innesto sono di lunghezza variabile e non esiste possibilità alcuna di errore. Questo è anche il momento buono per far passare i fili dei motori attraverso le feritoie dello chassis, in modo da farli convogliare verso la scheda relativa. Adesso è possibile accorciare un po' i fili uscenti dai motori, dal momento che all'atto del loro montaggio era stata prevista una lunghezza maggiore. La

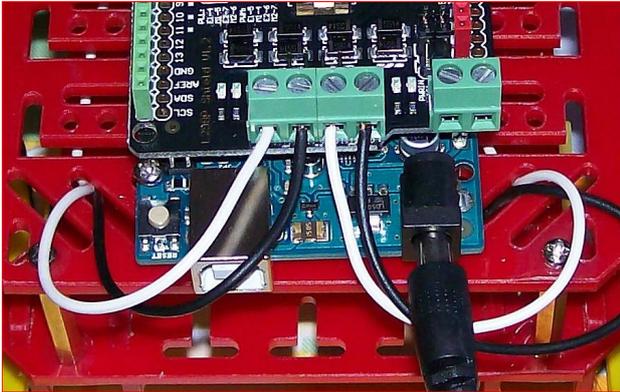


**FIGURA 20 : INSTALLAZIONE DI ARDUINO E MOTOR SHIELD SULLO CHASSIS**



**FIGURA 19: SCHEMA ELETTRICO PER LE SALDATURE DEI MOTORI**

riduzione del filo consente anche un miglioramento dell'estetica generale del robot, come si può evincere dalla figura 21. Dopo averli denudati, si possono inserire nel morsetto del motor shield, rispettando i piedini M1+, M1-, M2+ e M2-. Prima di proseguire con il montaggio è opportuno provare i motori e il loro verso di marcia, nonché la corretta rotazione delle quattro ruote. Allo scopo si può provare questo semplice sketch di prova. Esso non utilizza alcun sensore di ostacoli ma si limita a far girare le ruote in avanti per tre secondi e poi indietro per altri tre



**FIGURA 21 : COLLEGAMENTO DEI MOTORI AL MOTOR SHIELD**

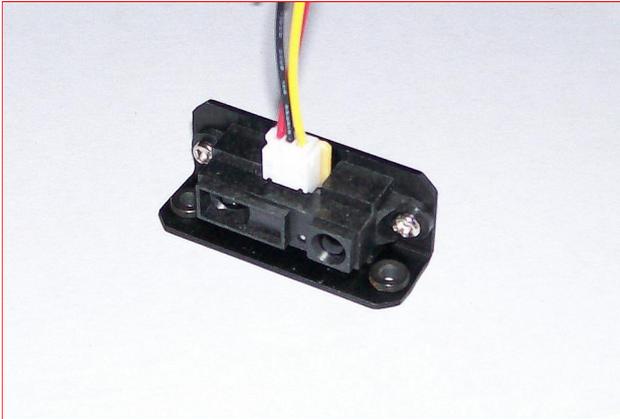
secondi, in un loop infinito, in modo da far “danzare” il robot. Ovviamente occorre evitare gli ostacoli perché, come detto, il listato non controlla la presenza di oggetti sul percorso del robot. Il valore del PWM è impostato 255, cosicché i motori girano al massimo della loro velocità, essendo il duty cycle stabilito al 100%. Il listato evidenzia che, per far avanzare il motore occorre agire sulle variabili M1 e M2, mentre per fermare i motori o regolarne la velocità si devono usare le variabili E1 e E2. Nel proseguo del tutorial vedremo come programmare Arduino con gli opportuni sistemi di sviluppo. Non serve mettere il robot a

terra, basta tenerlo in mano e verificare che le ruote girino correttamente. Si noti come la presenza di quattro motori conferisca al dispositivo una grande forza. Si consiglia di alimentare il robot con il portatile da 6 posti, fornito nella dotazione. Il motore si pilota agendo su alcuni valori logici delle porte, nonché sul duty cycle per controllarne la velocità.

## Installazione del sensore a infrarossi

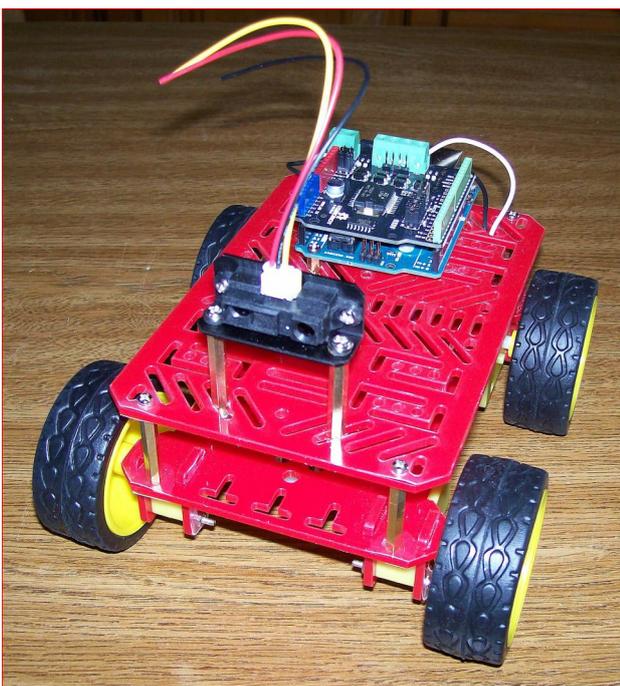
Il sensore di prossimità a infrarossi Sharp GP2Y0A21YK, come detto in precedenza, funge da “occhio” del robot, riuscendo a verificare e misurare la distanza di eventuali oggetti dinanzi ad esso. Le possibilità di montaggio sono le più disparate e grazie all’ampio ventaglio di scelte disponibili, possiamo effettuare il montaggio come meglio crediamo. Nel nostro esempio si è deciso di rendere il sensore fisso e solidale allo chassis. Altre scelte potranno prevederne l’installazione sul servo, in modo da renderlo orientabile al contesto ambientale incontrato. La

```
/*----Il Robot procede avanti e indietro----*/
int E1 = 5;
int M1 = 4;
int E2 = 6;
int M2 = 7;
void setup()
{
  pinMode(M1, OUTPUT);
  pinMode(M2, OUTPUT);
}
void loop()
{
  digitalWrite(M1,HIGH);
  digitalWrite(M2,HIGH);
  analogWrite(E1, 255); //PWM Speed Control
  analogWrite(E2, 255); //PWM Speed Control
  delay(3000);
  digitalWrite(M1,LOW);
  digitalWrite(M2,LOW);
  analogWrite(E1, 255); //PWM Speed Control
  analogWrite(E2, 255); //PWM Speed Control
  delay(3000);
}
```

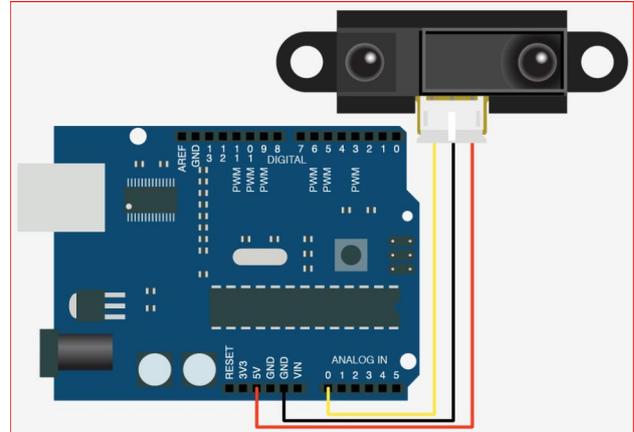


**FIGURA 22 : FISSAGGIO DEL SENSORE DI PROSSIMITÀ ALLA STAFFA**

prima operazione da compiere è quella del fissaggio del sensore alla rispettiva staffa. Due piccole viti e relativi dadi potranno assicurarne la perfetta aderenza e stabilità, come mostrato in figura 22. Utilizzando gli appositi distanziali forniti in dotazione si può, pertanto, montare il sensore sullo chassis (figura 22). Anche in questo caso si è pienamente autonomi e la posizione scelta del modulo a infrarossi può variare secondo le esigenze delle finalità del progetto finale previsto. Le soluzioni, a riguardo, sono molteplici.



**FIGURA 23 : MONTAGGIO DEL SENSORE DI PROSSIMITÀ ALLO CHASSIS**

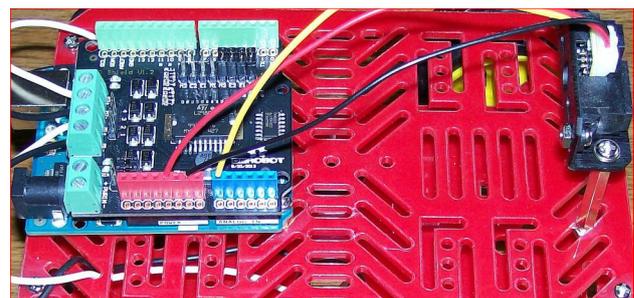


**FIGURA 24 : SCHEMA DI MONTAGGIO DEL SENSORE AD INFRAROSSI AD ARDUINO**

La connessione elettrica del sensore al sistema è estremamente semplice. Si devono collegare i fili di colore rosso e nero, rispettivamente, ai pin +5V e GND di Arduino e il filo giallo all'ingresso analogico 0, come mostrato nello schema di cui alla figura 24. In figura 25, invece, possiamo osservare il relativo il collegamento fisico. Se, per caso, i fili del sensore non dovessero arrivare sulla board, si possono usare utilmente i jumper a mo' di prolunghe, rispettando i colori per una maggiore chiarezza.

## Programma di prova per il sensore

Per testare la bontà di installazione del sensore scriviamo un piccolo sketch di prova. Lo scopo del programma è quello di far camminare il robot ad una certa velocità. Man mano che esso si



**FIGURA 25 : COLLEGAMENTO ELETTRICO DEL SENSORE AD ARDUINO**



avvicina a un ostacolo, la velocità diminuisce fino ad azzerarsi del tutto non appena il robot lo raggiunge. Il listato è molto semplice. Dopo il setup di rito, in un ciclo senza fine viene letta la porta analogica (0) che contiene il valore trasmesso dal sensore. Una serie di condizioni "if" verificano tale valore e decidono per il cambiamento di velocità dei motori. Il programma è molto generico ed ha solamente un carattere puramente illustrativo. L'utente può modificare le condizioni imposte e il numero di scaglionamenti delle distanze. Anche la tensione di alimentazione, infatti, influisce sulla potenza e velocità sprigionata dai motori. Può anche ricorrere a un'unica formula matematica adottando l'interpolazione lineare. Occorre ricordare che il sensore di distanza fornisce una misura analogica in volt, ma tale segnale è processato, da Arduino, come un numero intero a 10 bit, compreso tra 0 (0 Volt) e 1023 (5

Volt). Il listato mostra chiaramente questa prerogativa della gestione A/D.

## Programmazione di Arduino

Si può adesso programmare ufficialmente Arduino con il software predisposto a far compiere i primi passi al robot. Trattandosi, appunto, di un firmware, si può realizzare qualsiasi applicazione pratica, ludica o professionale. Occorre, allo scopo, avere dimestichezza con linguaggio di programmazione di Arduino. Dopo aver scaricato da Internet e installato l'Arduino Software (IDE), si può creare lo sketch del robot e copiare, almeno per le prime volte, quelli presenti su questo tutorial. Si presuppone che l'utente abbia anche installato il driver per il riconoscimento di Arduino, come dispositivo seriale COM, anche se connesso alla porta USB. Lo sketch può essere caricato su Arduino

```
/*----Il Robot rallenta in prossimità di un ostacolo----*/
int E1 = 5;
int M1 = 4;
int E2 = 6;
int M2 = 7;
int analogico;
int potenza;
void setup()
{
  pinMode(M1, OUTPUT);
  pinMode(M2, OUTPUT);
}
void loop()
{
  analogico = analogRead(0);
  if(analogico<400)
    potenza=255;
  if(analogico>=400 && analogico<600)
    potenza=100;
  if(analogico>=600 && analogico<800)
    potenza=50;
  if(analogico>=800)
    potenza=0;
  digitalWrite(M1, LOW);
  digitalWrite(M2, LOW);
  analogWrite(E1, potenza); //PWM Speed Control
  analogWrite(E2, potenza); //PWM Speed Control
  delay(10);
}
```



```
sketch_mar16a | Arduino 1.6.7
File Modifica Sketch Strumenti Aiuto
sketch_mar16a
int E1 = 5;
int M1 = 4;
int E2 = 6;
int M2 = 7;
int analogico;
int potenza;
void setup()
{
  pinMode(M1, OUTPUT);
  pinMode(M2, OUTPUT);
}
void loop()
{
  analogico = analogRead(0);
  if(analogico<400)
    potenza=255;
  if(analogico>=400 && analogico<600)
    potenza=100;
  if(analogico>=600 && analogico<800)
    potenza=50;
  if(analogico>=800)
    potenza=0;
}
```

Salvataggio completato

```
avrdude done. Thank you.
```

14 Arduino/Genuino Uno on COM12

**FIGURA 26: LO SKETCH DEL ROBOT PRONTO PER ESSERE CARICATO**

premendo l'apposito tasto dell'IDE (un tondino con la freccia rivolta a destra) oppure attraverso il menù "Sketch" e la voce "Carica". Se l'operazione va a buon fine, il robot deve funzionare immediatamente. E' possibile utilizzare anche la combinazione veloce dei tasti <CTRL><U>. E' ovvio che il listato deve essere esente da errori di sintassi.

## Installazione del servo

Dopo aver opzionalmente provato i listati, si può adesso montare il servo (il suo nome completo è servomotore o servocomando). Esso è un dispositivo, molto usato nel modellismo, ed è stato creato per posizionare, con elevata precisione, un braccetto o un

ingranaggio ad una certa angolazione. Esso, dunque, non ha finalità di rotazione continua ma limitata ad un movimento angolare di precisione. Il servo può essere comandato con un segnale PWM. Lo chassis offre, come per tutti gli altri dispositivi, un ampio ventaglio di soluzioni per il montaggio. Nell'esempio abbiamo preferito installarlo sulla parte anteriore del robot, tramite un opportuno distanziatore, come si evince in figura 27.

## Collegamento del servo ad Arduino

Tutti i servi sono dotati di tre fili: il rosso va collegato all'alimentazione (+5V), il

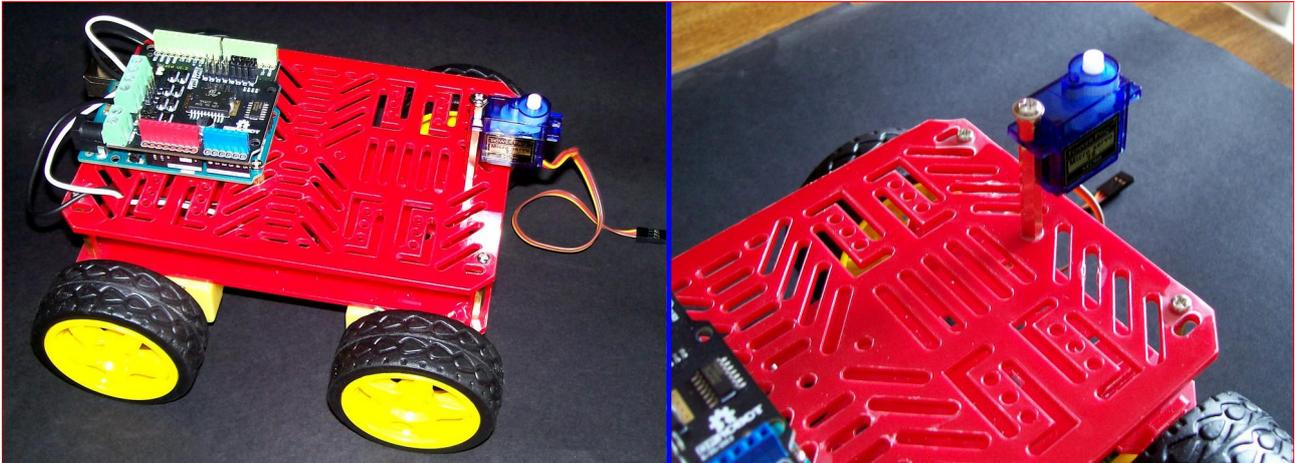


FIGURA 27: MONTAGGIO DEL SERVO SULLO CHASSIS

nero o marrone alla massa comune e il giallo, trasportante il segnale di comando, a una porta digitale di Arduino. Per l'esempio successivo si è scelto di effettuare la connessione verso la porta numero 2. La figura 28 mostra la connessione del servo ad Arduino, attraverso i jumper maschio/maschio in dotazione.

## Programma dimostrativo per il servo

Come esempio di utilizzo del servo proponiamo uno sketch, estremamente didattico, che apre le porte a qualsiasi tipologia di applicazioni con tale dispositivo mobile meccanico. Con gli opportuni adattatori e staffe in dotazione, abbiamo montato sul servo

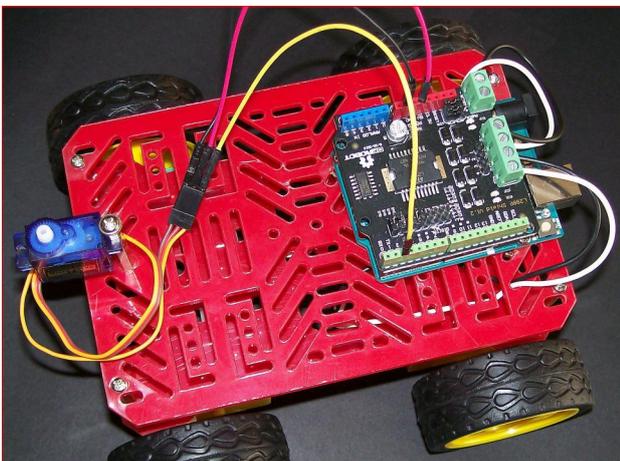
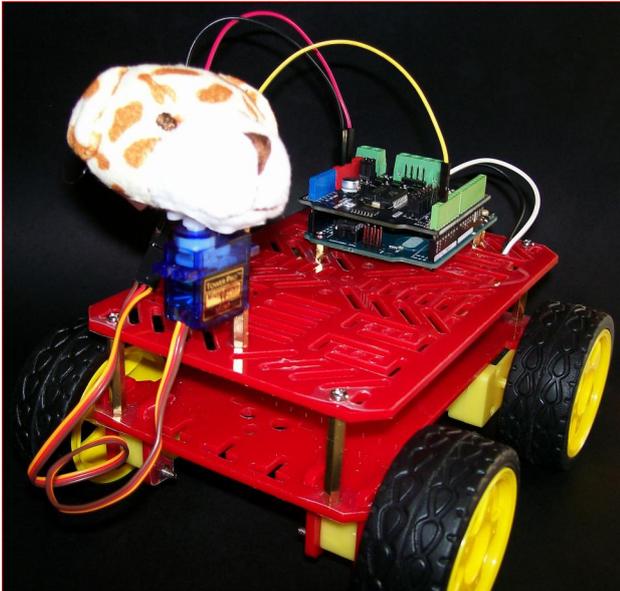


FIGURA 28: COLLEGAMENTO DEL SERVO AD ARDUINO

una testa di un cagnolino, come visibile in figura 29. Il listato che segue permette di far effettuare delle rotazioni, a destra e a sinistra al cane, come se stesse guardando in entrambe le direzioni. Un'eventuale applicazione, più complessa, potrebbe prevedere il movimento della testa nella stessa direzione del robot, come se stesse seguendo la strada durante il suo cammino. L'algoritmo utilizza la libreria apposita per la gestione dei servi, già disponibile con l'IDE di Arduino. E' necessario includerla con la direttiva `#include <Servo.h>`, all'inizio del programma. Quindi viene decisa la porta digitale su cui è collegato il servo. Tale libreria crea un oggetto, completamente gestibile tramite funzioni e metodi. Con il metodo `write` si può specificare l'angolo di rotazione dell'albero, compreso tra  $0^\circ$  e  $180^\circ$ . Le applicazioni di un servo con un robot sono pressoché infinite e solo la fantasia può limitare il campo di azione della sua sperimentazione. Può essere utilizzato per muovere sensori, lampade, telecamere, ecc. Per movimenti più complessi del servo si possono utilizzare i cicli "for" e le condizioni. L'utilizzo delle librerie consente, indubbiamente, di ottimizzare i tempi di sviluppo e di programmare in



**FIGURA 29: ROBOT/CANE CON SERVO PER LA ROTAZIONE DELLA TESTA**

maniera assolutamente semplice. In ogni caso sappiate che la programmazione di un servo può anche avvenire senza l'ausilio di alcuna libreria, agendo direttamente sul pilotaggio a basso livello delle porte digitali. Ciò si effettua usando intensivamente la tecnica del PWM e dosando opportunamente la durata degli impulsi e i relativi duty cycles.

## Conclusioni

L'esperienza con il Robot Beginner KIT 4WD è stata molto appassionante.

Dall'osservazione preliminare del prodotto e dei suoi componenti si è capito immediatamente che avevamo tra le mani un prodotto di qualità, ben progettato in tutti i particolari. Gli elementi da montare sono relativamente pochi e si assemblano con estrema semplicità. Lo chassis è stato progettato estremamente bene e offre infinite soluzioni di personalizzazione, grazie alla presenza dei numerosi fori e fessure di diversa misura, che consentono il montaggio di qualsiasi dispositivo in varie posizioni. La programmazione del sistema è molto semplice ed è possibile iniziare da subito la creazione dei primi prototipi. Si consiglia di utilizzare le pile alcaline, e non le batterie economiche, poco adatte per via delle basse correnti erogate. Come tutti i dispositivi elettronici si deve sempre prestare la massima cautela nel montaggio e nell'utilizzo. Il prodotto ha suscitato grande interesse di chi scrive, e ha dato tanti nuovi spunti e idee per il suo interessante utilizzo. Un apparecchio che, soprattutto, ha dato nuove opportunità per la realizzazione di progetti più importanti e sostanziosi.

```
/*----Il Robot/cane ruota la testa a destra e a sinistra----*/
#include <Servo.h>
int cane = 2;
Servo Servo1;
void setup()
{
  Servo1.attach(cane);
}
void loop()
{
  Servo1.write(90);
  delay(1000);
  Servo1.write(30);
  delay(1000);
  Servo1.write(90);
  delay(1000);
  Servo1.write(150);
  delay(1000);
}
```