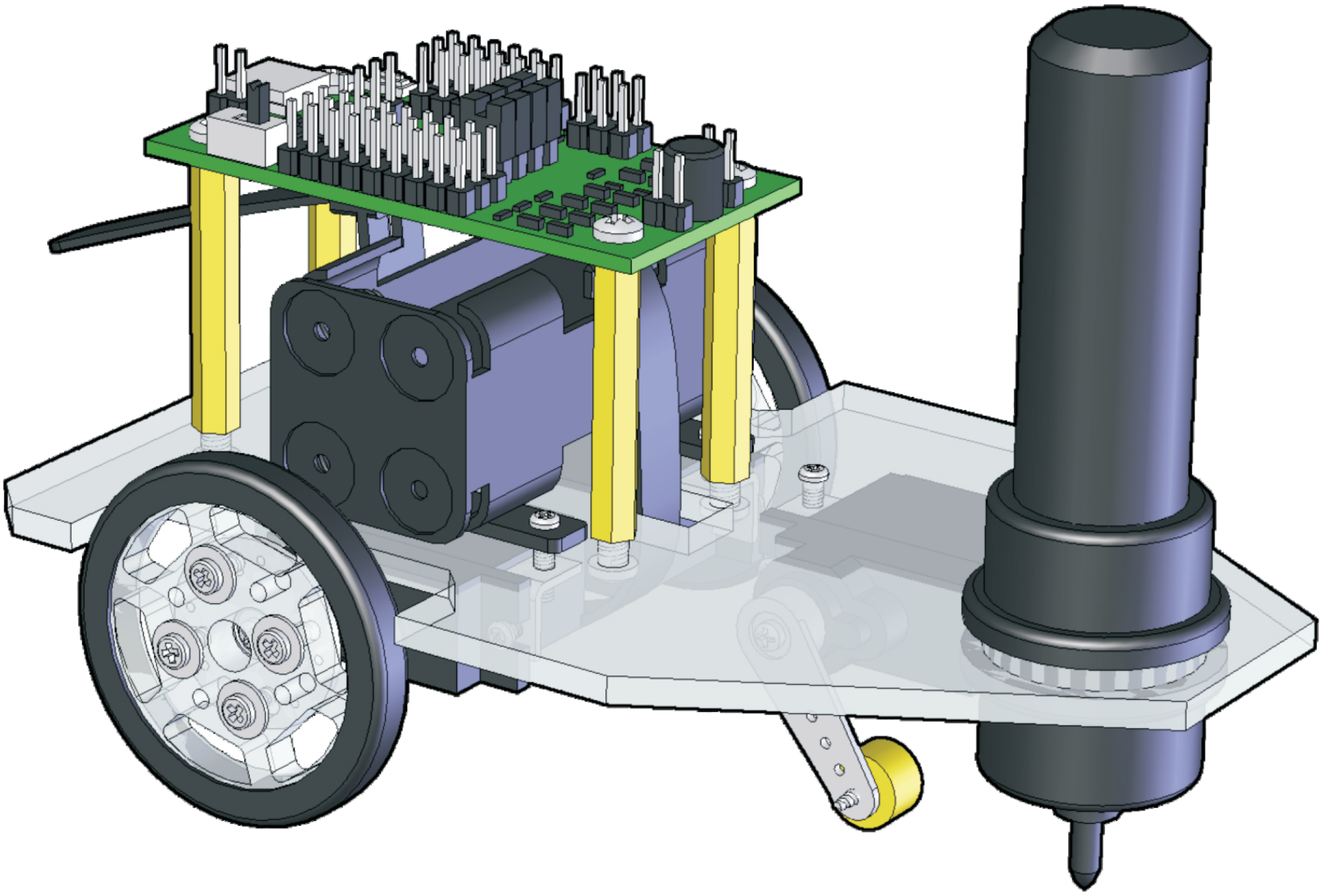
RS025

DOODLE BOT



The manufacturer and distributor cannot be held responsible for any damages occurred by mishandling, mounting mistakes or misuse due to non-respect of the instructions contained in this manual.

DAGU Electronic

Manufacturer:
DAGU Hi-Tech Electronic Co.,LTD
WWW.AREXX.COM.CN

Address:NO.4-107/108 HengXing Street, HengHai Rd, South District, ZhongShan City of GuangDong China
TEL:0760-88811951
http://www.arexx.com.cn
E-mail:info@arexx.com.cn

CONTENTS

Product Description

Thank you for selecting the DOODLE BOT for your next do-it-your-self project. This kit allows you to build and program a small drawing robot that can write messages and draw pictures using a large pen, crayon or chalk stick.

Product Features

1. Arduino compatible robot controller with USB interface.
2. Laser cut, transparent acrylic base plate.
3. 2x high torque geared motor with ball bearing.
4. 1x high torque miniature servo with ball bearing.
5. Rubber tires and pen gripper.

Build it now! Realize your dream! Create your next masterpiece!

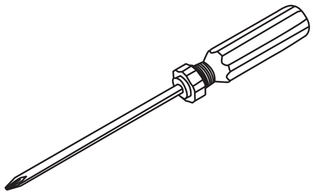
⚠ Warnings:

- Opened packages can not be returned. Please check package contents before opening.
- Read the instructions carefully before assembly.
- Use all tools carefully.
- Small parts are a choking hazard. Keep this kit away from young children and babies during construction and operation.
- Not for children under 8 years. Not to be used by children except under adult supervision.
- Observe the correct polarity of the battery.
- Keep the battery dry at all times.
- Do not mix old and new batteries. Do not mix alkaline, standard (carbon-zinc), or rechargeable (nickel-cadmium) batteries.
- Remove the battery if the kit is not used for a long period of time.

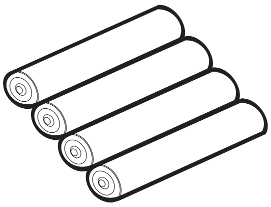
Necessary Tools:

Please read this manual thoroughly before you start assembling the kit. Please follow the assembling instructions exactly to avoid problems. If you work accurately and follow the instructions in this manual exactly, you will quickly assemble your DOODLE BOT.

Before you start you must prepare the following tools:



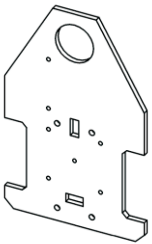
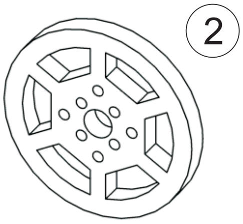
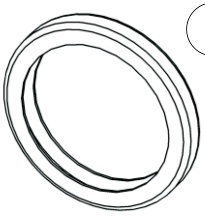
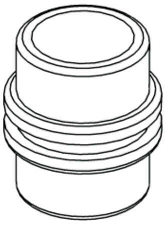
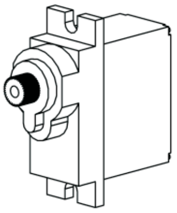
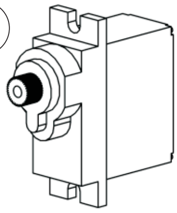
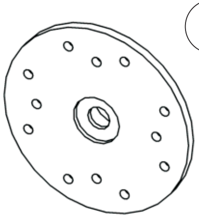
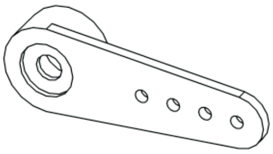
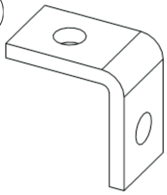
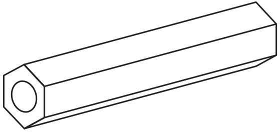
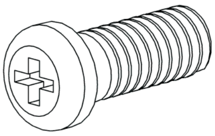
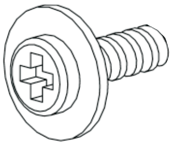
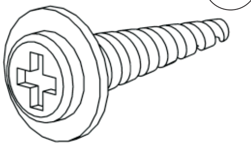
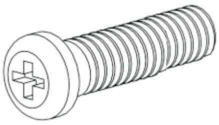
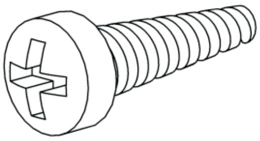
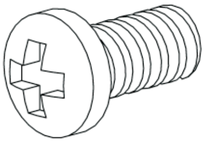
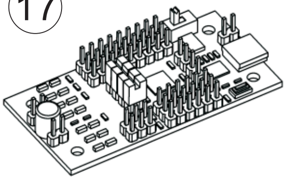
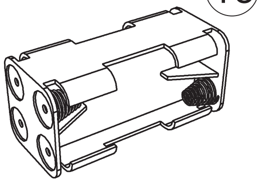
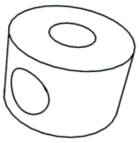
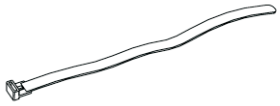
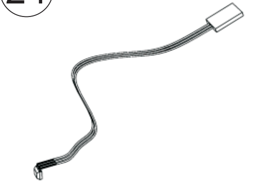
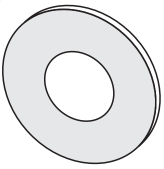
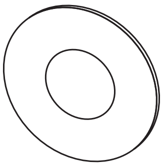
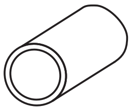
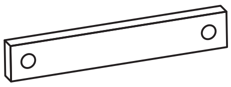
PHILLIPS SCREW DRIVER
(included in kit)



4x AAA BATTERY
(not included)

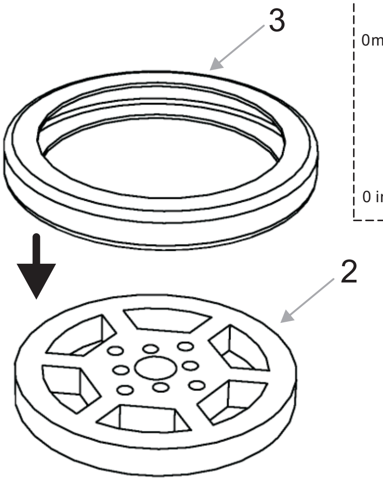
PEN / CRAYON / CHALK
(not included)

Parts List:

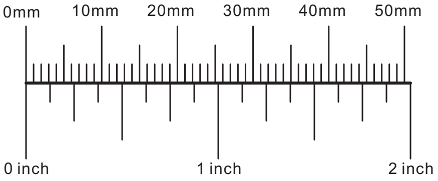
1  Base plate 1pc	2  Wheel 2pcs	3  Rubber tire 2pcs	4  Rubber pen gripper 1pc	5  9g geared motor (two wires) 2pcs
6  12g Miniature servo (three wires) 1pcs	7  Round servo horn 2pcs	8  Servo horn-2 1pc	9  L shape mounting bracket 6pcs	10  L30 distance holder 4pcs
11  M2*5 screw 8pcs	12  M2*5 self-tapping screw with flange 8pcs	13  M2.3*10 self-tapping screw with flange 1pc	14  M2*8 screw 4pcs	15  M2*5 self-tapping screw 3pcs
16  M3*6 screw 8pcs	17  Controller board 1pc	18  battery holder 1pc	19  Collar 1pc	20  Re-useable Cable tie 1pc
21  Hall sensor 2pcs	22  8 pole magnet 2pcs	23  Adhesive ring 2pcs	24  Spacer 1pc	25  Fiberboard strip 2pcs

Assembly Instructions:

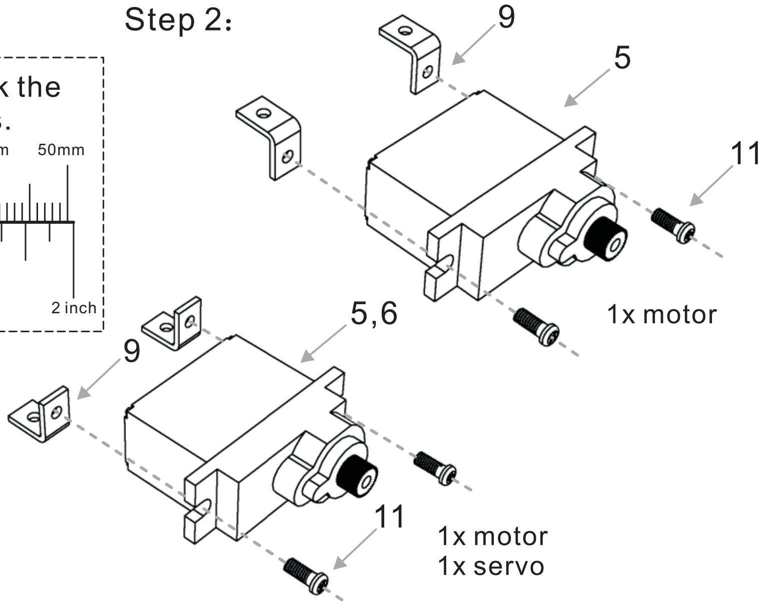
Step 1:



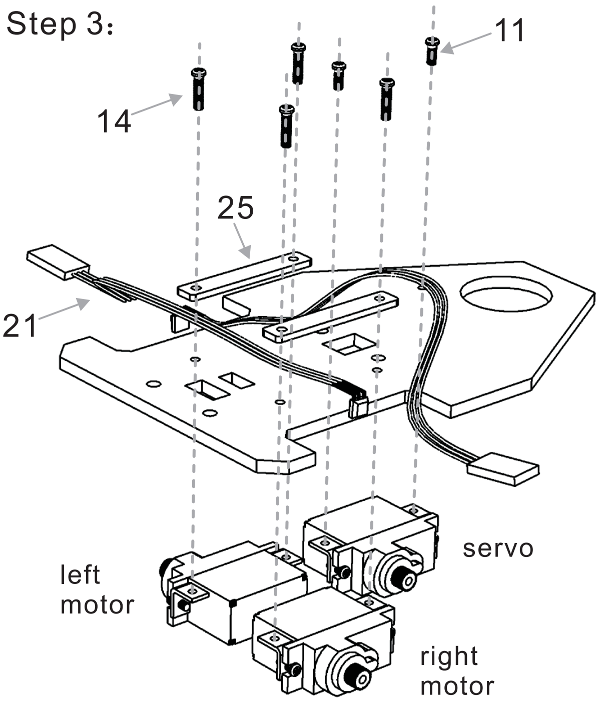
Use the ruler to check the length of your screws.



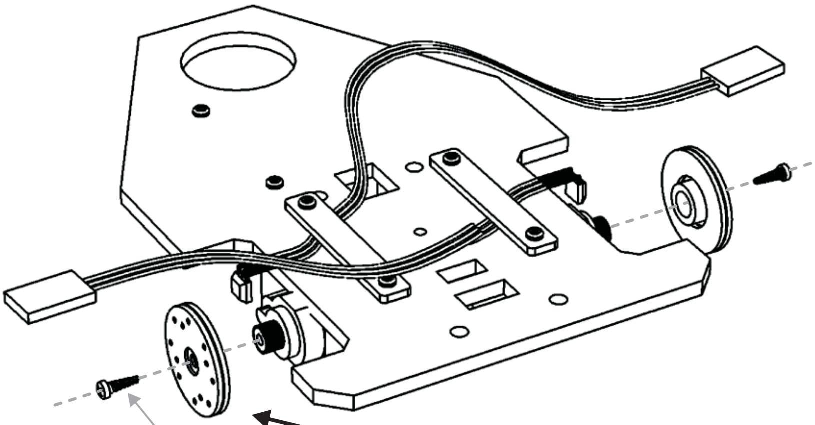
Step 2:

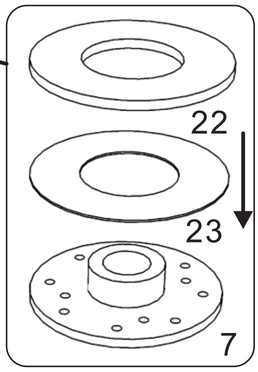


Step 3:



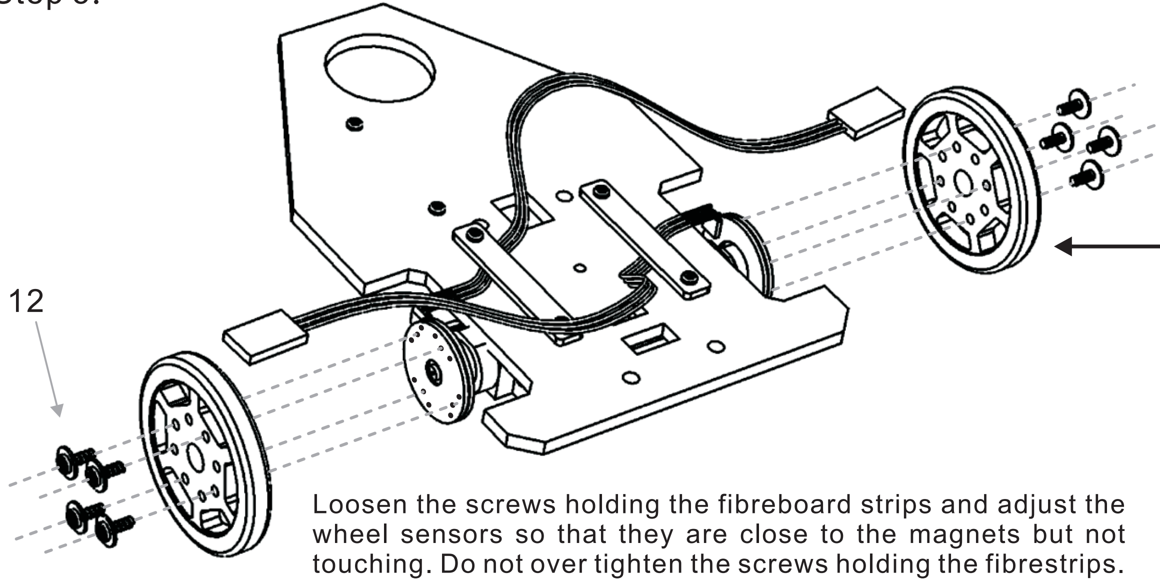
Step 4:

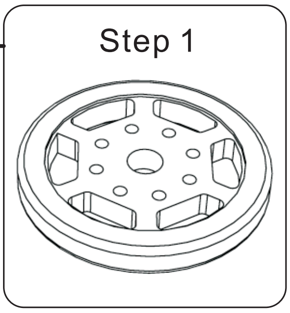




Attach the 8 pole magnets to the round servo horns using the adhesive rings provided. Attached to the motors with M2x5mm screws.

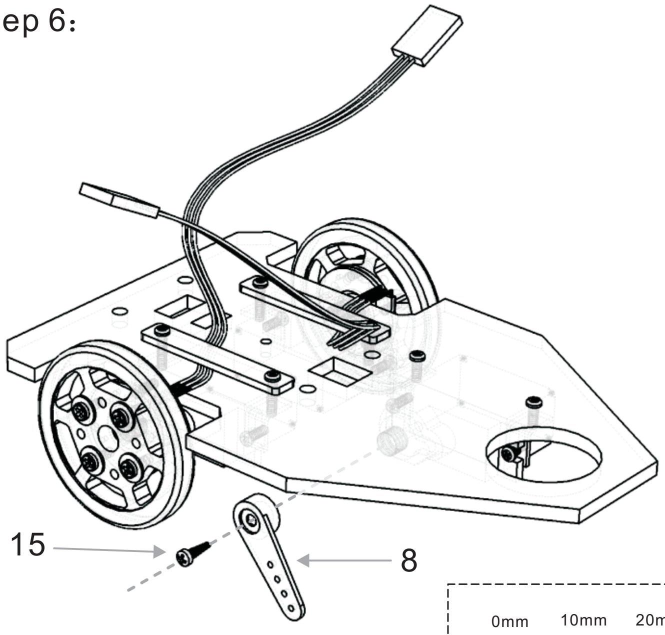
Step 5:



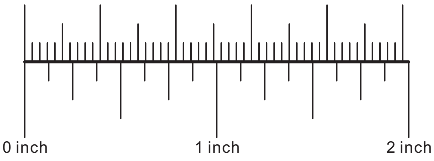


Loosen the screws holding the fibreboard strips and adjust the wheel sensors so that they are close to the magnets but not touching. Do not over tighten the screws holding the fibrestrips.

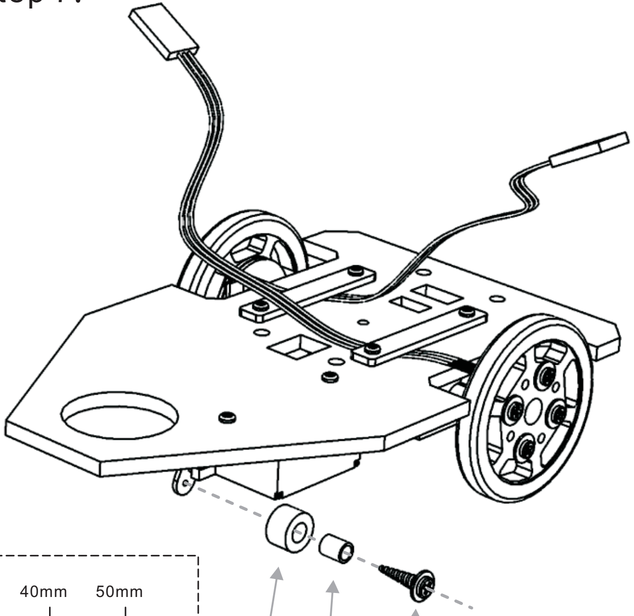
Step 6:



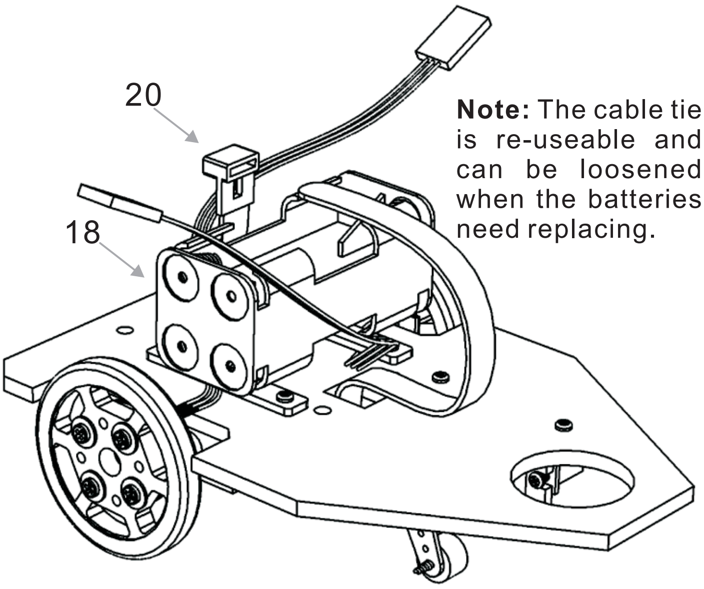
Center your servo and fit the servo horn so it is pointing straight down.



Step 7:

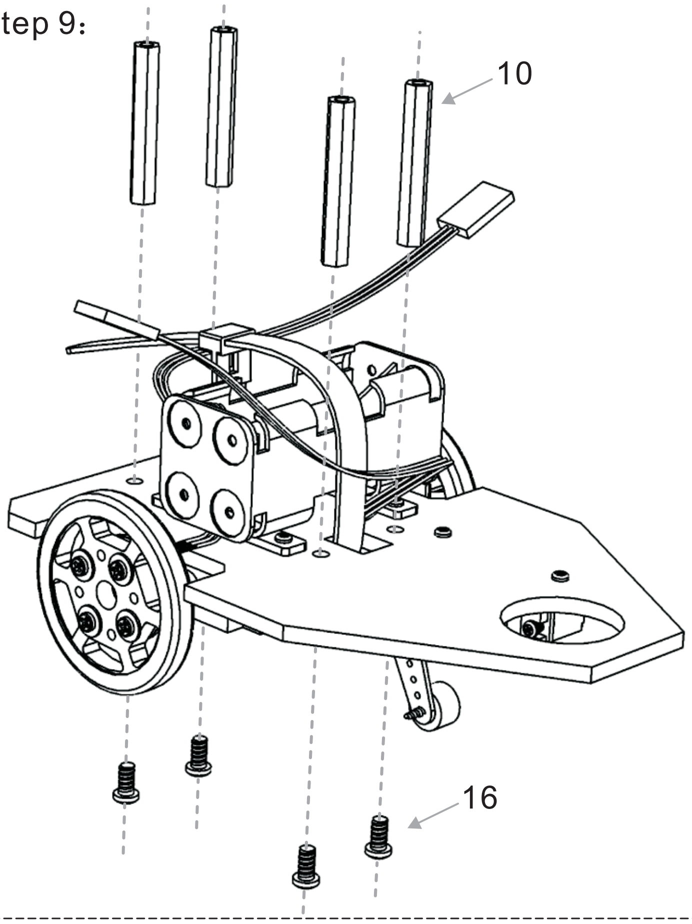


Step 8:

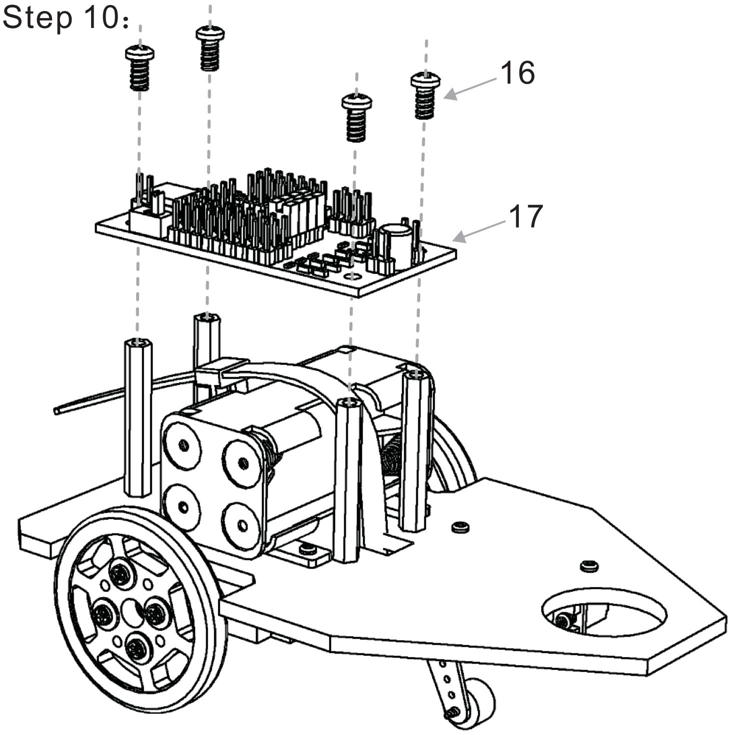


Note: The cable tie is re-useable and can be loosened when the batteries need replacing.

Step 9:



Step 10:

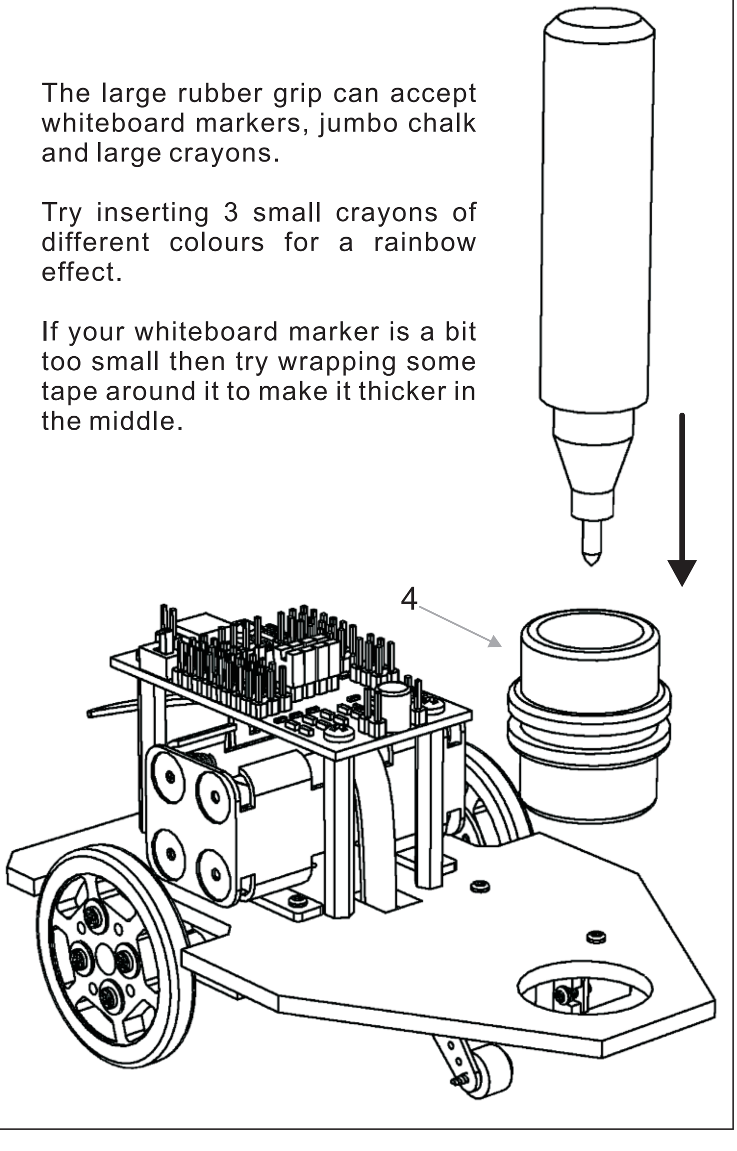


Step 11:

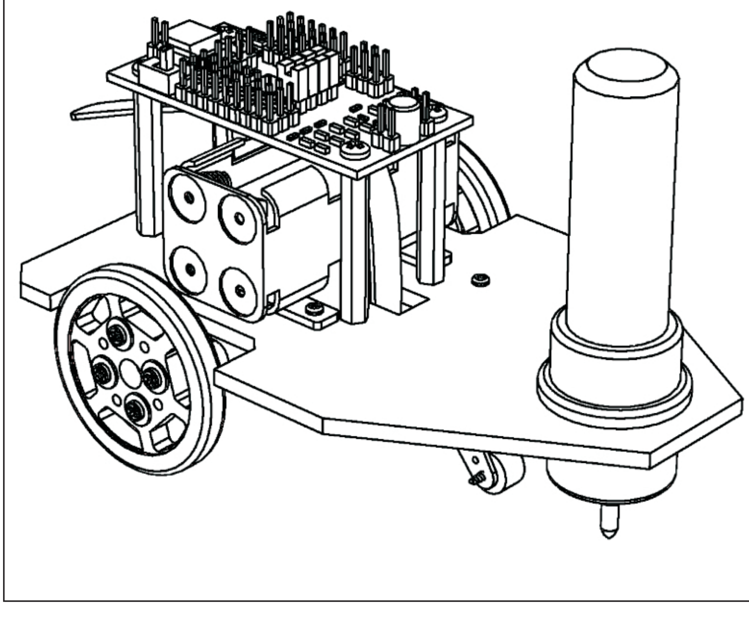
The large rubber grip can accept whiteboard markers, jumbo chalk and large crayons.

Try inserting 3 small crayons of different colours for a rainbow effect.

If your whiteboard marker is a bit too small then try wrapping some tape around it to make it thicker in the middle.



Finished



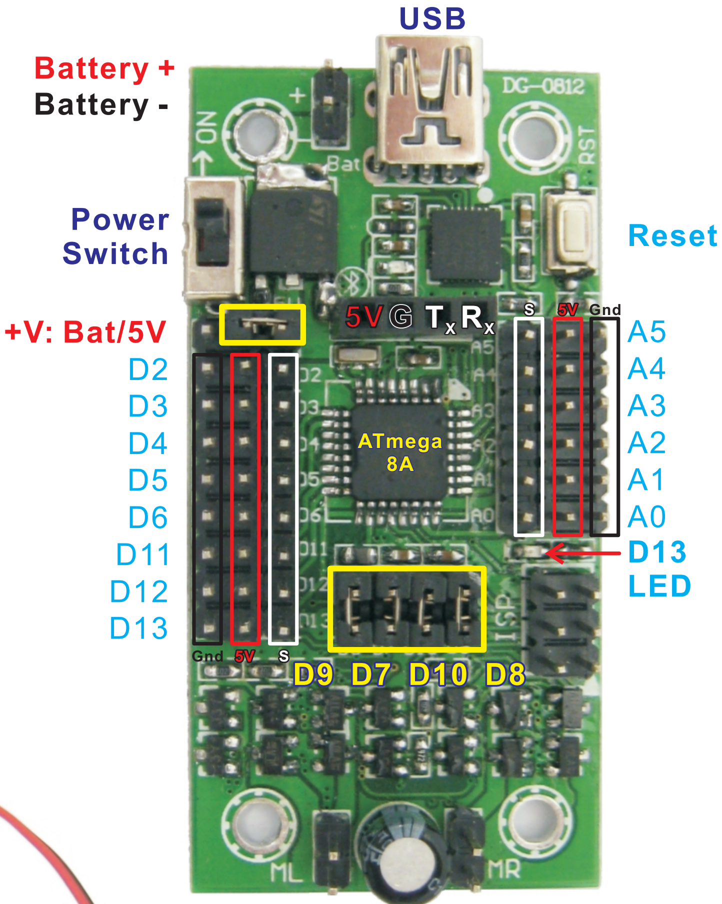
Wiring Instructions:

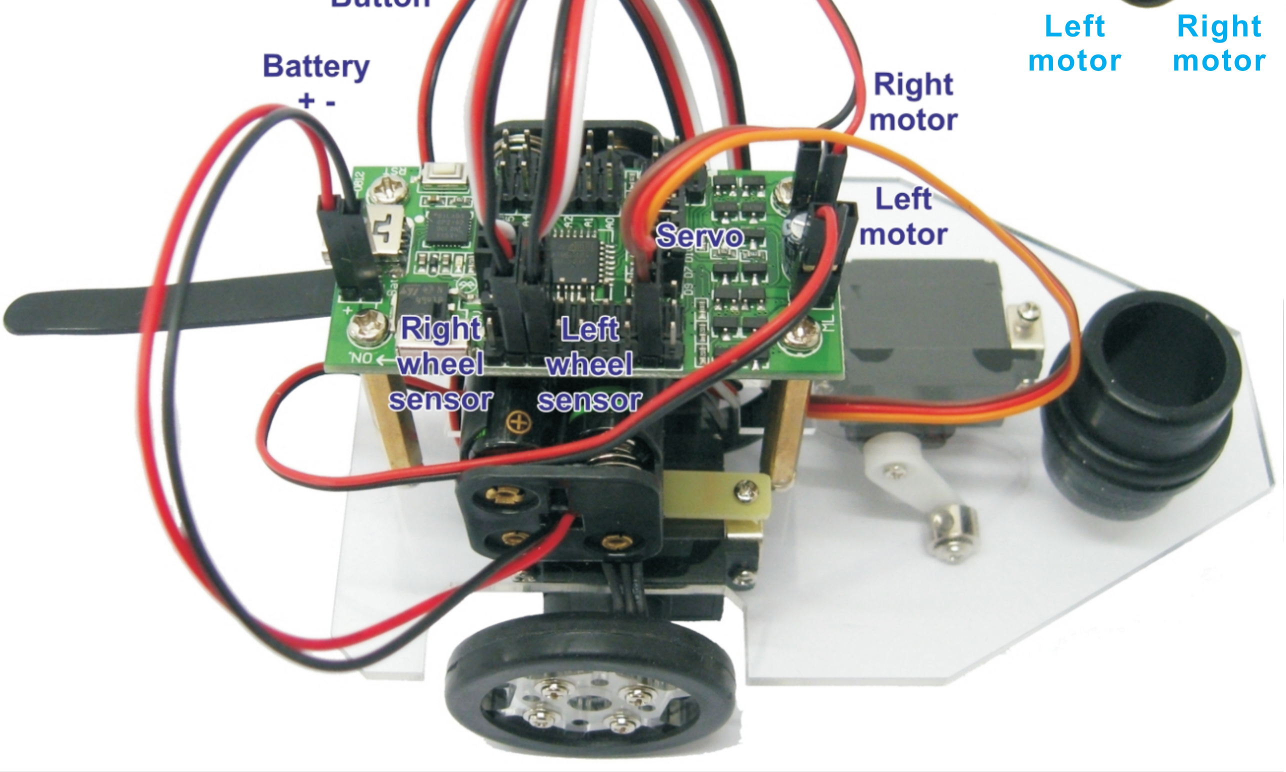
Once you have assembled the robot you need to connect the wires. Start by checking that the jumper near the power switch is selecting 5V for power. If it is set to battery voltage then the sensor outputs might be high enough to damage the digital inputs D2 & D3.

Connect the left and right motors as shown in the photo. Make sure the jumpers are on D7, D8, D9 & D10 so that the processor is connected to the motor driver. Now connect the power wires from the battery. Make sure you connect the red wire to the +positive connection.

Next we need to connect the servo and wheel encoder sensors. These all have three wires - black/brown is the ground wire and goes to the outer edge of the PCB. The red wire is the +positive and should be in the center. The white/orange wire is the signal wire and will be closest to the processor.

The right wheel sensor connects to D2, the left wheel sensor connects to D3 and the servo is connected to D12.





Installing the Software:

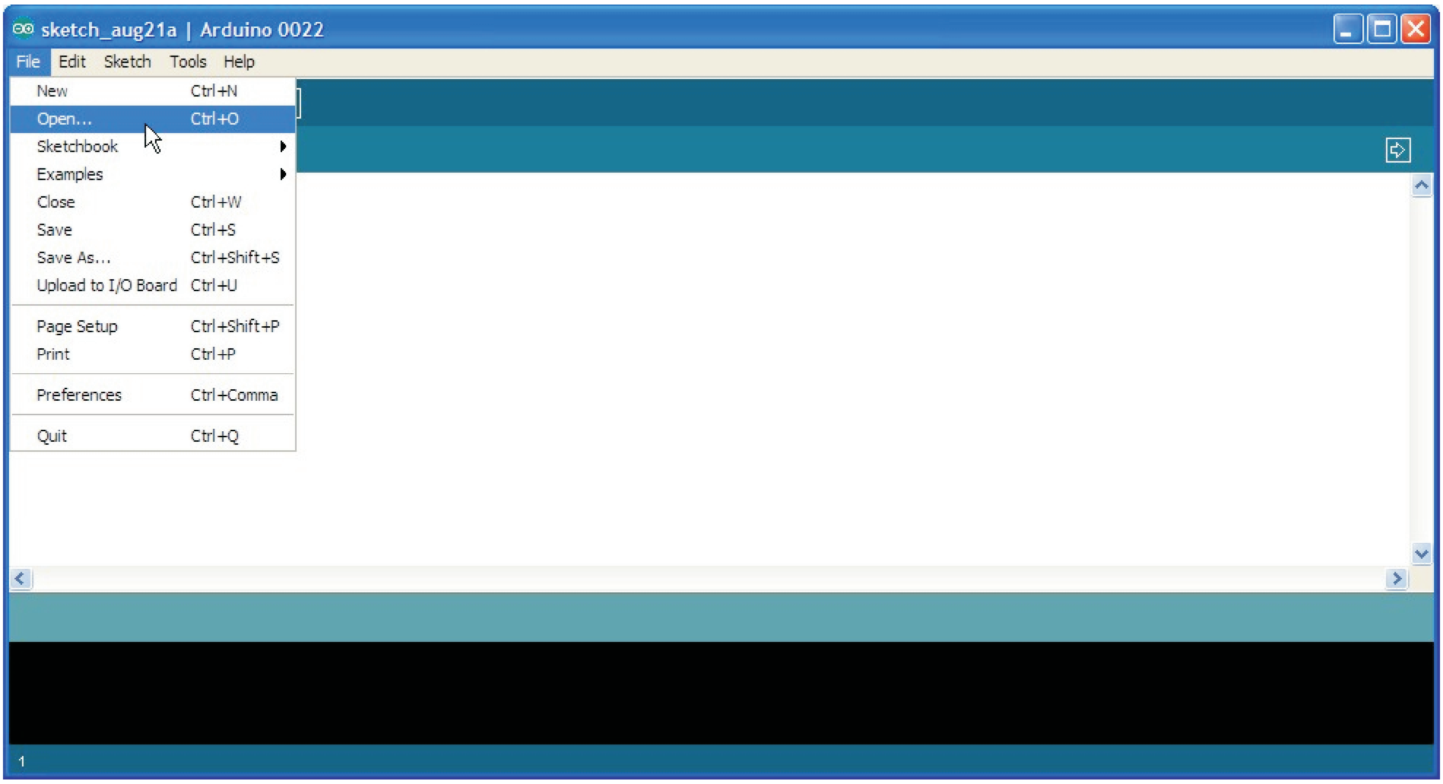
Before you begin you will need to have the Arduino IDE version 0022 installed on your computer. The Arduino IDE is a free integrated development enviroment that allows you to program Arduino development boards and compatible products using a simplified version of C++.

Do not use later versions as the demonstration software will not work and the compiled program uses more memory. If you do not have this software installed already then download it from here:
http://arduino.cc/en/Main/Software

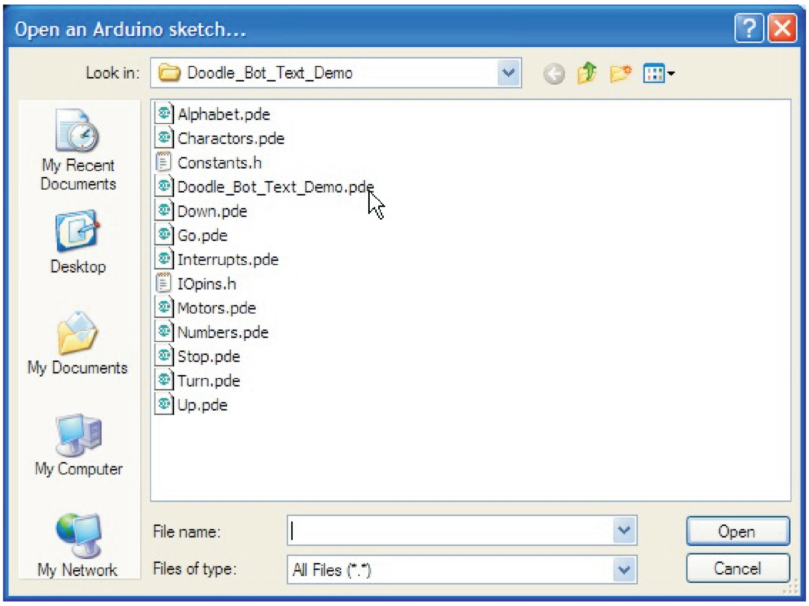
You may also need to install VCP (virtual com port) drivers for the CP2102 USB interface. You can download the latest drivers from here:
http://www.silabs.com/products/mcu/Pages/USBtoUARTBridgeVCPDrivers.aspx

The sample code for this robot should be available from the distributor. It is also available from the DAGU product support site here:
https://sites.google.com/site/daguproducts/home/download-page

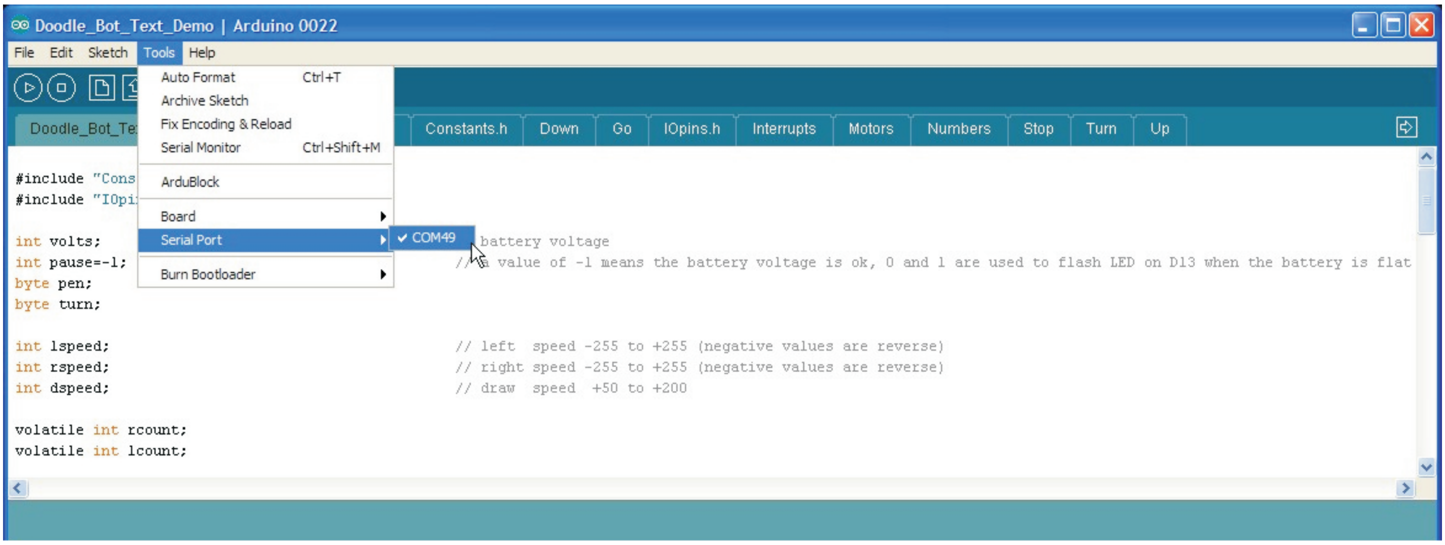
Unzip the sample code and copy the folder into your sketch pad or other directory of your choice. For Windows users this will be a folder called "Arduino" in the "My Documents" folder. Run the Arduino IDE, go to the file menu and select open.



Select the "Doodle_Bot_Text_Demo" folder and then open the "Doodle_Bot_Text_Demo.pde" file.

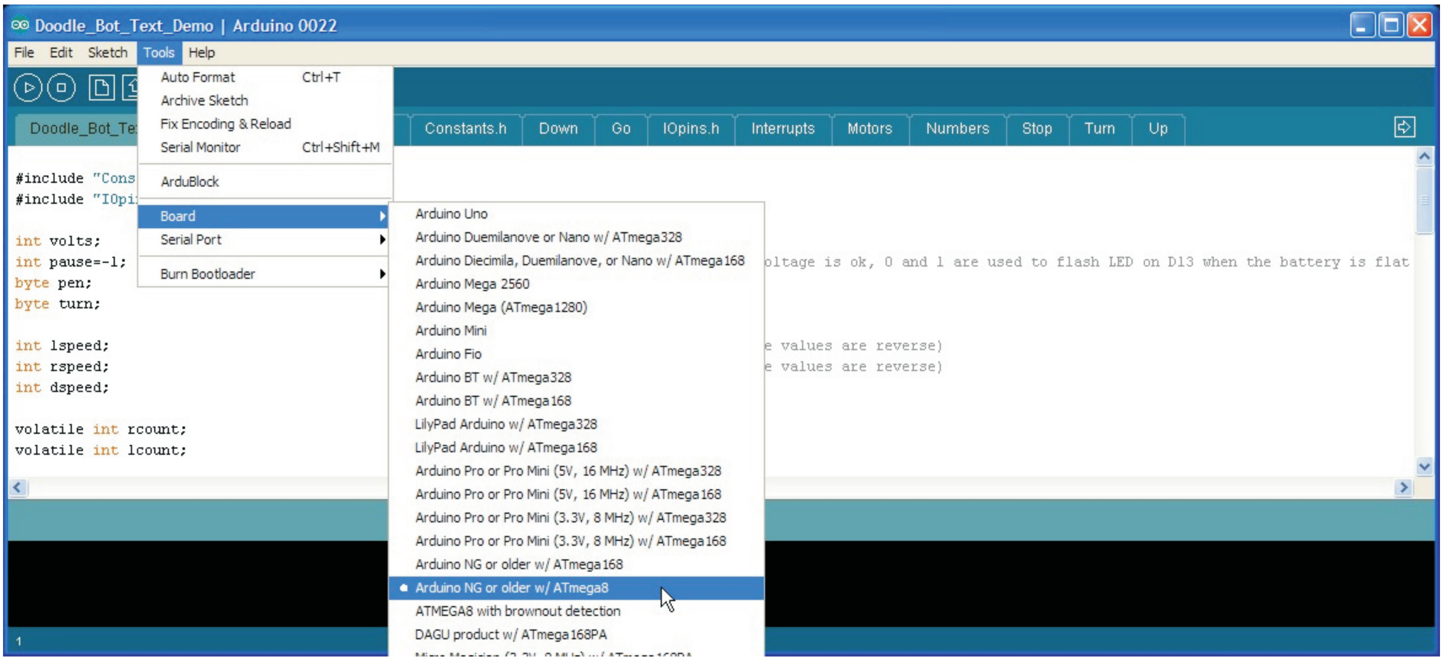


Make sure the robot is turned off and connect it to your computer with a USB cable. The control board will be powered by the USB cable while you upload the program but there will be no power to the motors. Go to the "Tools" menu and select your USB serial port.



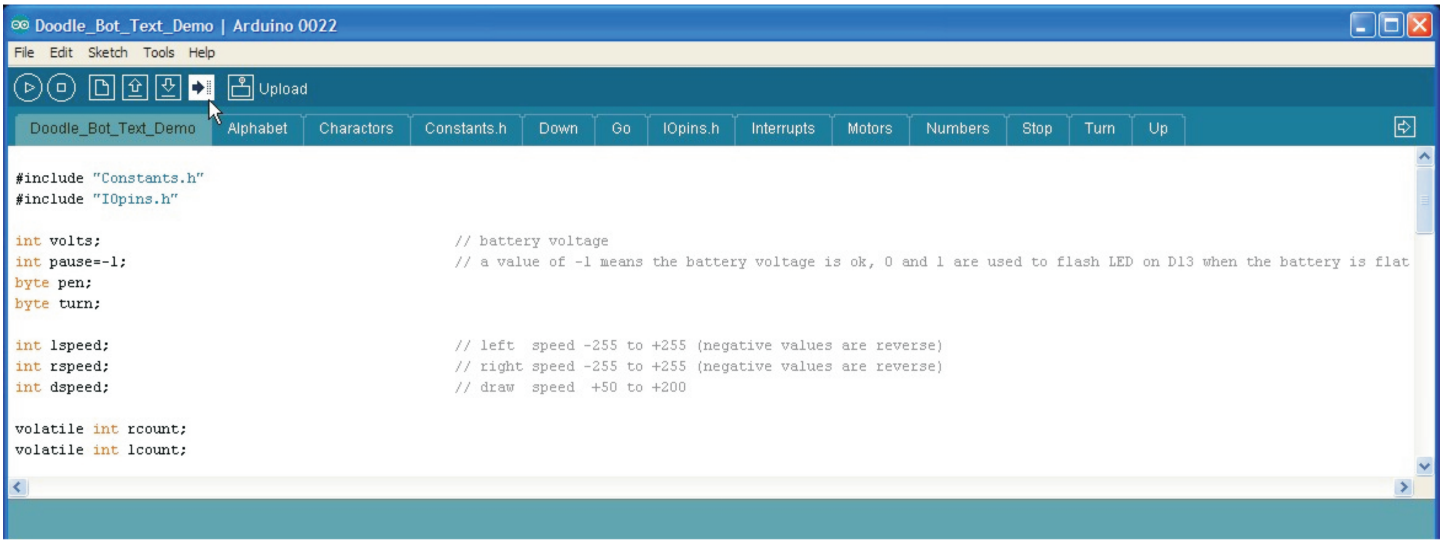
```
#include "Cons
#include "IOpi

int volts;                                          battery voltage
int pause=-1;                                       //  a value of -1 means the battery voltage is ok, 0 and 1 are used to flash LED on D13 when the battery is flat
byte pen;
byte turn;

int lspeed;                                         // left  speed -255 to +255 (negative values are reverse)
int rspeed;                                         // right speed -255 to +255 (negative values are reverse)
int dspeed;                                         // draw  speed  +50 to +200

volatile int rcount;
volatile int lcount;
```

From the "Tools" menu choose the board type. The Doodle Bot controller is compatible with an Arduino NG or older using an ATmega8 processor.


Doodle_Bot_Text_Demo | Arduino 0022
File
Edit
Sketch
Tools
Help
Auto Format Ctrl+T
Archive Sketch
Fix Encoding & Reload
Serial Monitor Ctrl+Shift+M
ArduBlock
Board
Serial Port
Burn Bootloader
Arduino Uno
Arduino Duemilanove or Nano w/ ATmega328
Arduino Diecimila, Duemilanove, or Nano w/ ATmega168
Arduino Mega 2560
Arduino Mega (ATmega1280)
Arduino Mini
Arduino Fio
Arduino BT w/ ATmega328
Arduino BT w/ ATmega168
LilyPad Arduino w/ ATmega328
LilyPad Arduino w/ ATmega168
Arduino Pro or Pro Mini (5V, 16 MHz) w/ ATmega328
Arduino Pro or Pro Mini (5V, 16 MHz) w/ ATmega168
Arduino Pro or Pro Mini (3.3V, 8 MHz) w/ ATmega328
Arduino Pro or Pro Mini (3.3V, 8 MHz) w/ ATmega168
Arduino NG or older w/ ATmega168
● Arduino NG or older w/ ATmega8
ATMEGA8 with brownout detection
DAGU product w/ ATmega168PA
Doodle_Bot_Te
Constants.h
Down
Go
IOpins.h
Interrupts
Motors
Numbers
Stop
Turn
Up
1

```
#include "Cons
#include "IOpi

int volts;
int pause=-1;                                       oltage is ok, 0 and 1 are used to flash LED on D13 when the battery is flat
byte pen;
byte turn;

int lspeed;                                         e values are reverse)
int rspeed;                                         e values are reverse)
int dspeed;

volatile int rcount;
volatile int lcount;
```

When you upload the program you will see a message indicating the program is compiling, then uploading. When the upload is complete you can disconnect the USB. The robot is ready to go! **Note:** It is normal for the robot to wait a few seconds before the program runs.



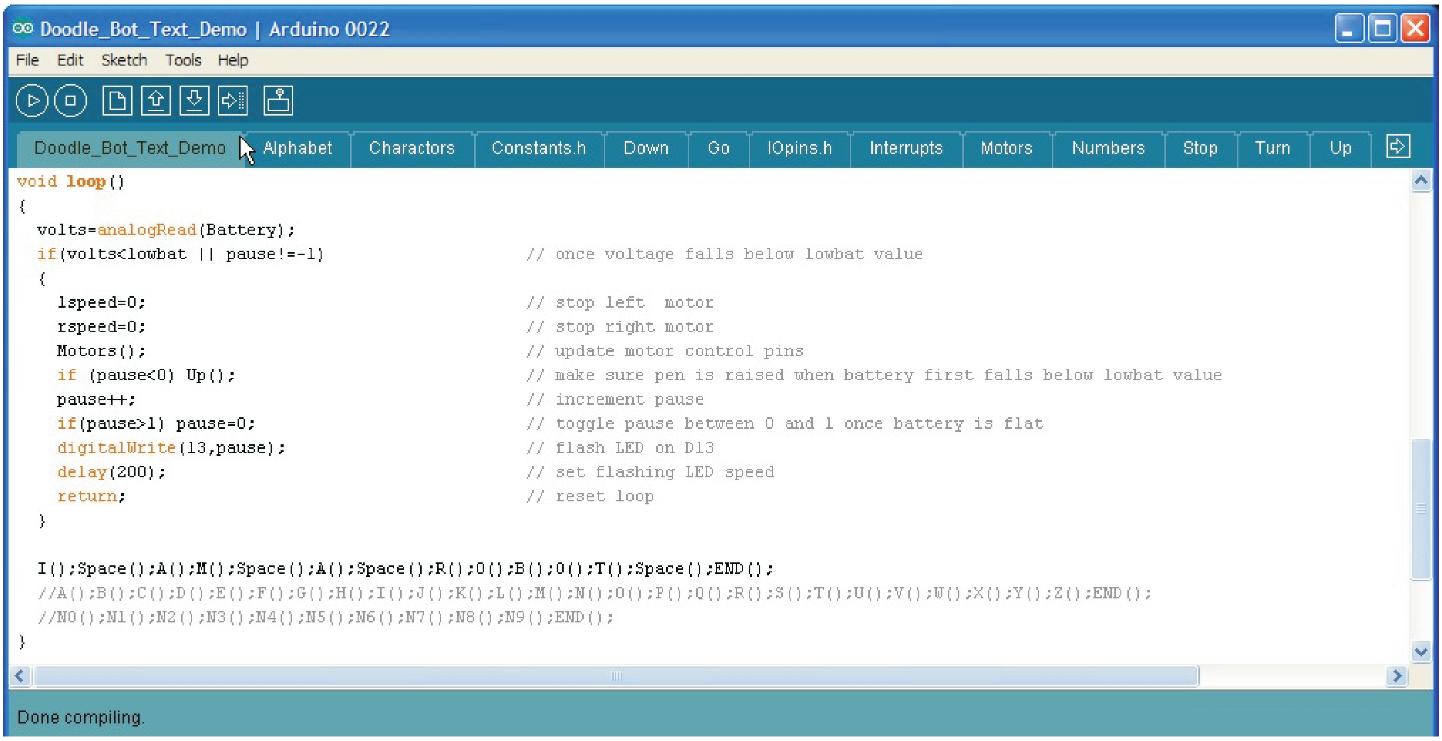
```
#include "Constants.h"
#include "IOpins.h"

int volts;                                          // battery voltage
int pause=-1;                                       // a value of -1 means the battery voltage is ok, 0 and 1 are used to flash LED on D13 when the battery is flat
byte pen;
byte turn;

int lspeed;                                         // left  speed -255 to +255 (negative values are reverse)
int rspeed;                                         // right speed -255 to +255 (negative values are reverse)
int dspeed;                                         // draw  speed  +50 to +200

volatile int rcount;
volatile int lcount;
```

Changing the text message and size :

The first tab contains the setup and loop functions required for all Arduino code. The main loop is where you will program the robot with different messages.

When you run the program, the robot will print the message: "I AM A ROBOT". There are two other samples that allow the robot to print the alphabet or the numbers 0-9. Note that if you do not finish your message with "END();" then the robot will repeat the message continuously.

The main loop also checks the battery voltage. If the battery is too flat then the robot will stop and the D13 LED will flash quickly to indicate the battery is flat.



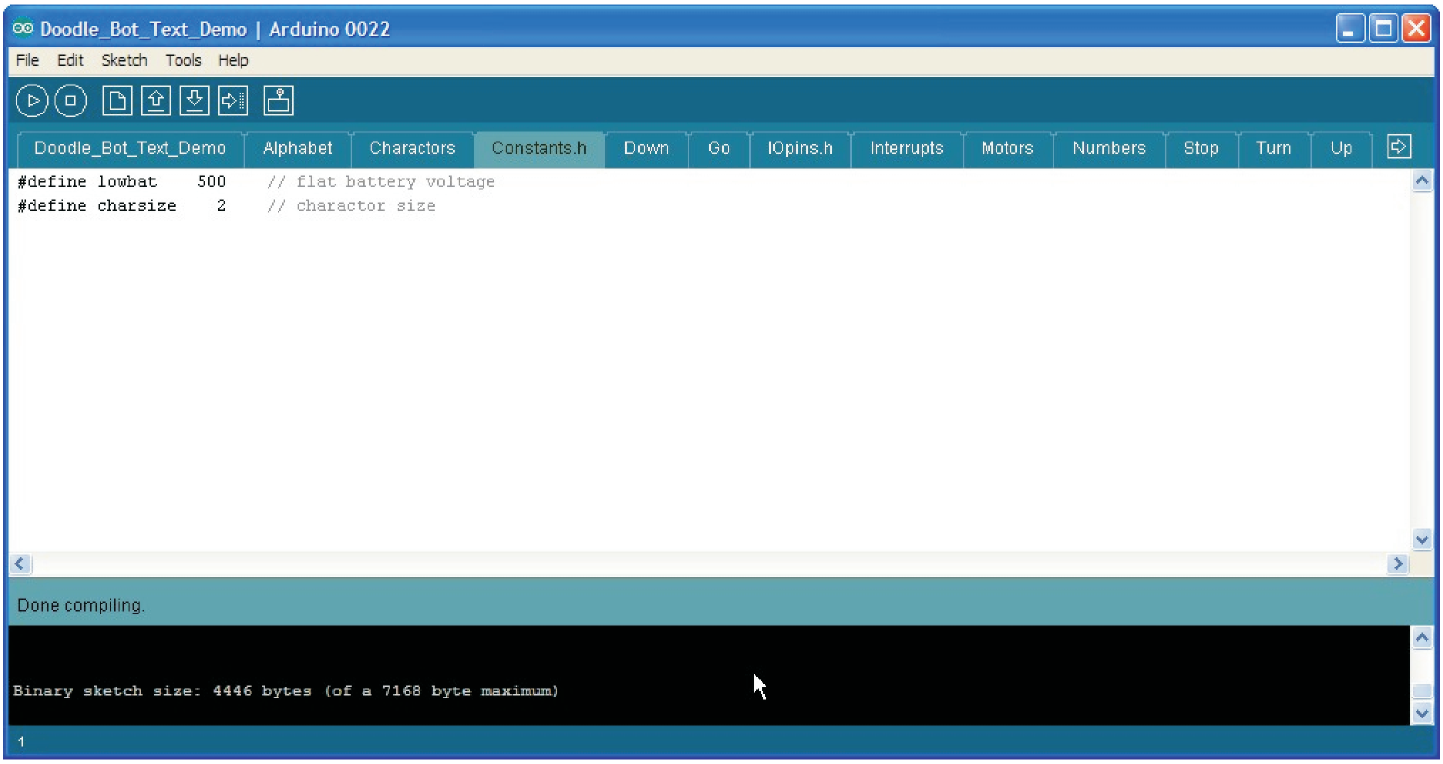
```
void loop()
{
  volts=analogRead(Battery);
  if(volts<lowbat || pause!=-1)                   // once voltage falls below lowbat value
  {
    lspeed=0;                                     // stop left  motor
    rspeed=0;                                     // stop right motor
    Motors();                                     // update motor control pins
    if (pause<0) Up();                            // make sure pen is raised when battery first falls below lowbat value
    pause++;                                      // increment pause
    if(pause>1) pause=0;                          // toggle pause between 0 and 1 once battery is flat
    digitalWrite(13,pause);                       // flash LED on D13
    delay(200);                                   // set flashing LED speed
    return;                                       // reset loop
  }

  I();Space();A();M();Space();A();Space();R();O();B();O();T();Space();END();
  //A();B();C();D();E();F();G();H();I();J();K();L();M();N();O();P();Q();R();S();T();U();V();W();X();Y();Z();END();
  //N0();N1();N2();N3();N4();N5();N6();N7();N8();N9();END();
}
```

Done compiling.

To change the size of the text you must change the value of "charsize" in the "Constants.h" tab. This is a whole number from 1 to 255 with 1 being the smallest.

Changing "low battery" sensitivity :

You can change the point at which the robot decides the battery is too flat in the "constants.h" tab by changing the value of "lowbat". A value of 512 is approximately 5V. Do not set this too low otherwise the bootloader could be corrupted.



```
#define lowbat     500    // flat battery voltage
#define charsize     2    // charactor size
```

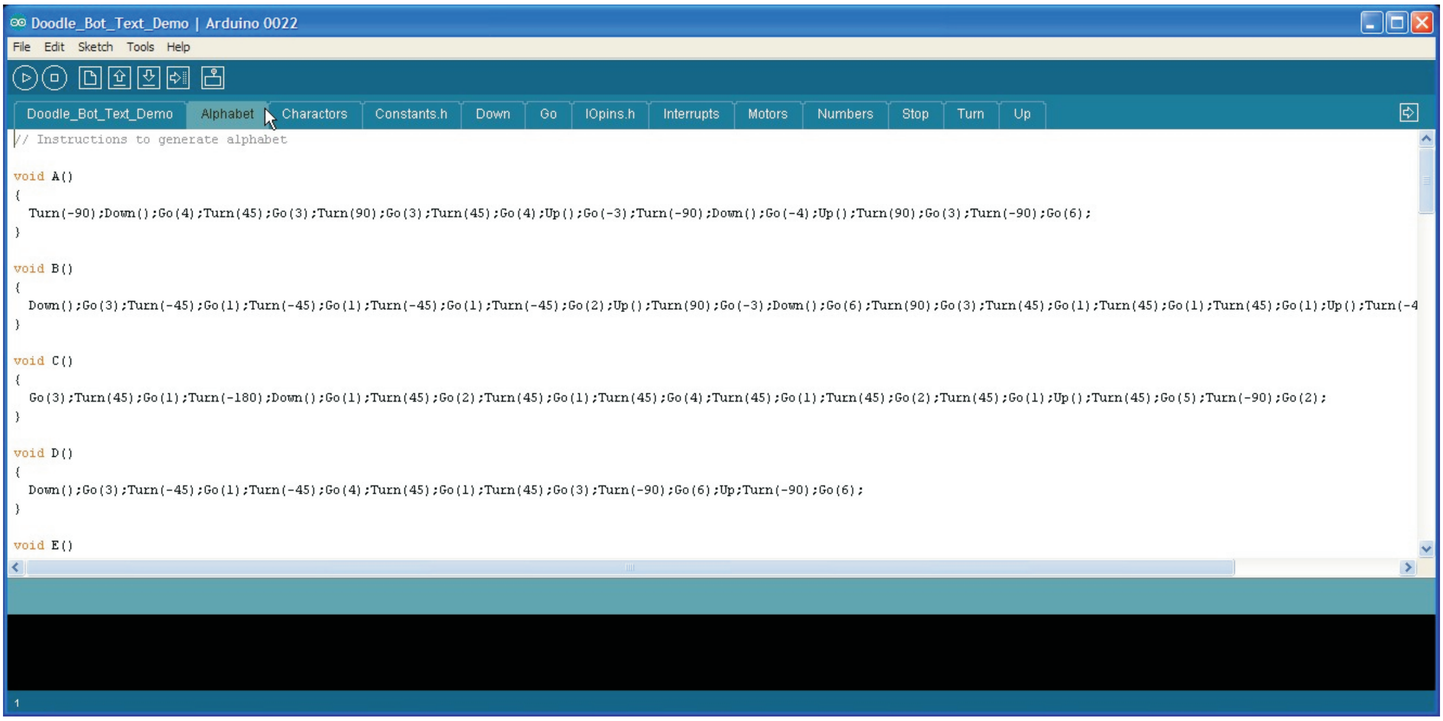
Done compiling.

```
Binary sketch size: 4446 bytes (of a 7168 byte maximum)
```

1

Creating your own charactors :

If you look in the tabs "Alphabet" and "Numbers" you will see that all the numbers and letters are created using combinations of only 4 commands.



```
// Instructions to generate alphabet

void A()
{
  Turn(-90);Down();Go(4);Turn(45);Go(3);Turn(90);Go(3);Turn(45);Go(4);Up();Go(-3);Turn(-90);Down();Go(-4);Up();Turn(90);Go(3);Turn(-90);Go(6);
}

void B()
{
  Down();Go(3);Turn(-45);Go(1);Turn(-45);Go(1);Turn(-45);Go(1);Turn(-45);Go(2);Up();Turn(90);Go(-3);Down();Go(6);Turn(90);Go(3);Turn(45);Go(1);Turn(45);Go(1);Turn(45);Go(1);Up();Turn(-4
}

void C()
{
  Go(3);Turn(45);Go(1);Turn(-180);Down();Go(1);Turn(45);Go(2);Turn(45);Go(1);Turn(45);Go(4);Turn(45);Go(1);Turn(45);Go(2);Turn(45);Go(1);Up();Turn(45);Go(5);Turn(-90);Go(2);
}

void D()
{
  Down();Go(3);Turn(-45);Go(1);Turn(-45);Go(4);Turn(45);Go(1);Turn(45);Go(3);Turn(-90);Go(6);Up;Turn(-90);Go(6);
}

void E()
```

If you look in the tabs "Alphabet" and "Numbers" you will see that all the numbers and letters are created using combinations of only these 4 functions.

Go(distance) Tells the robot to travel in a straight line. The distance can be a positive or negative whole number with negative numbers making the robot from reverse.

Turn(angle) Tells the robot to turn left or right. A positive value turns clockwise, negative turns counter clockwise. Angle can be in 45° steps or values of ±1 to 32 which specifies the number of 22.5° steps.

Down() Lowers the pen. The robot begins drawing.

Up() Raises the pen. The robot stops drawing.

Using just these four functions you can write messages and draw pictures. There is a tab called "Charactors" where you can add your own custom charactors and icons. Advanced users may choose to write additional functions to suit their needs.

Troubleshooting :

Please note: The motors may be a little stiff when new causing the robot to draw wiggly lines. This problem should improve after about an hour of use as the motors wear in. The wiggle is a result of one motor being slower than the other and the demonstration software attempting to correct this by a very simple method. It is not a fault.

Problem: When you power the robot by USB it works ok but when you disconnect the USB and turn the robot ON then nothing happens.
Solution: Check that the power wires from the battery are connected the correct way around. Make sure your batteries have been inserted into the holder correctly. Make sure the batteries are not flat.

Problem: When you turn the robot on it just sits there with the D13 LED flashing quickly.
Solution: The robot has determined that your batteries are flat. Change the batteries.

Problem: Instead of drawing straight lines the robot draws curved lines.
Solution: Watch carefully when the demonstration program first runs. If a motor goes backward instead of forward then swap it's wires around.

Problem: When the program runs, the robot turns clockwise when it should turn anticlockwise.
Solution: Swap the left and right motors.

Troubleshooting continued:

Problem: One of the motors does not turn.
Solution: Check that the jumpers are placed correctly on D7, D8, D9 and D10.

Problem: When the program runs, the robot goes forward slightly and then goes in circles.
Solution: Make sure your encoders are plugged into the correct sockets as shown (Wiring Instructions).

Problem: The pen does not raise or lower. The servo just sits in one position.
Solution: Check that the servo is connected to D12. Make sure the signal wire (orange or white) is closest to the processor as shown in the wiring instructions. Check that V+ selection jumper is set to +5V.

Problem: The USB port is not recognised by your computer or does not show up in the Arduino IDE.
Solution: Make sure you do not have anything connected to the serial communications socket while using the USB cable. Make sure you have the latest CP2102 USB interface drivers installed (Installing the Software).

Using the controller as an Arduino board:

The DAGU mini driver is software compatible with an Arduino NG. It has been designed specifically for robotics so it has a few extra features.

Up to 8 servos can be connected directly to the PCB by changing the +V selector from 5V to Bat. This allows the servos to be powered directly from the battery as the 5V regulator is only capable of powering 1 miniature servo under a light load.

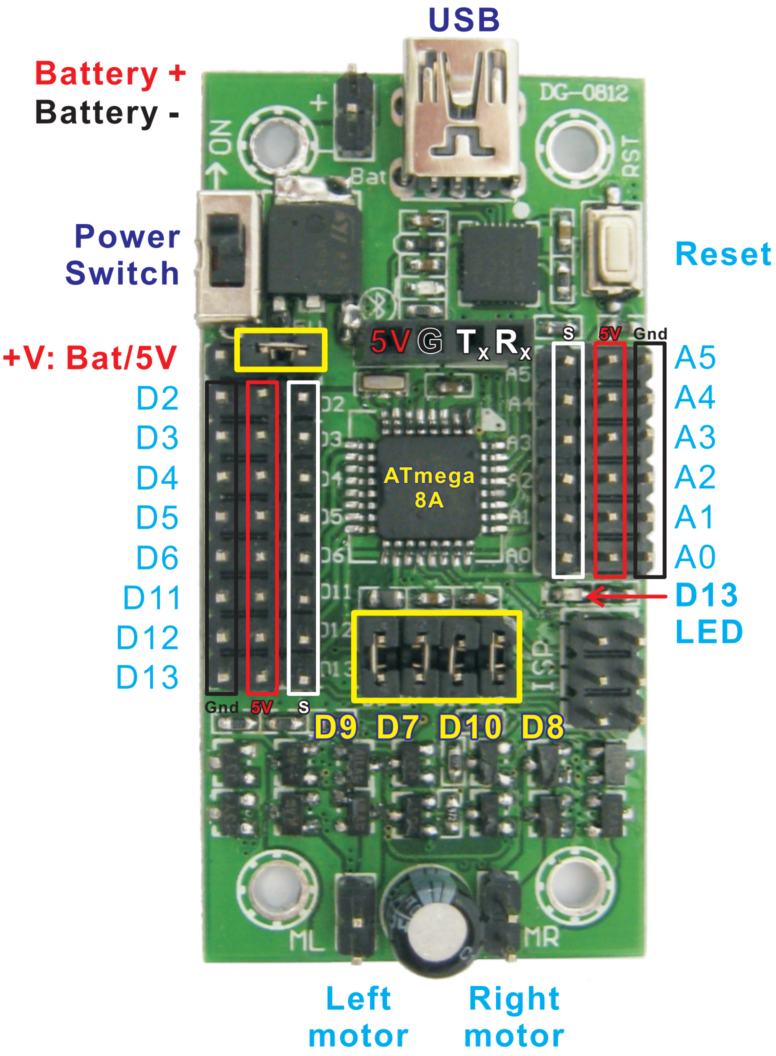
Analog pins A0-A5 have 5V and Gnd pins available for powering 5V sensors. As the SMD chip has extra pins there are 2 exta analog inputs. A6 is not used although it can be accessed by a solder pad under the PCB. A7 is configured to monitor battery voltage. The maximum allowable battery voltage is 9V.

The dual "H" bridge is capable of driving 2 motors, each with a stall current of 2A or 1 stepper motor with a maximum of 2A per winding.

If you do not wish to use the motor driver then you can remove the jumpers on D7, D8, D9 and D10. You can then use these pins for alternate purposes.

There is a 4 pin female header on the PCB that is intended for use with a wireless transceiver such as an Xbee or Bluetooth module. You can access D0 and D1 (Rx and Tx) directly from this header.

The ISP socket is the same as a standard Arduino board and can be used to re-burn the bootloader. Newer versions of the Arduino IDE will allow you to program the board directly from this socket. Note that there is a small white arrowhead next to pin 1.



Specifications:

Controller:	
Processor:	ATmega8A
Clock speed:	16MHz
FLASH:	8K
SRAM:	1K
EEPROM:	512 bytes
USB interface:	CP2102
Supply voltage:	5V - 9V
Regulator:	1A max.
Dual "H" bridge:	2A per motor

Output devices:		
Servo voltage:	4V - 6V	
Servo torque:	2Kg/cm	6V
Servo speed:	0.08 sec/60°	6V
Servo weight:	12g	
Motor voltage:	6V	
Gearbox torque:	2Kg/cm max.	
Speed:	125RPM ±10%	6V
Weight:	9g	