E4: Introduction to making things move on Arduino: Motors, A Primer.

Introduction

Motor turn electrical energy into mechanical energy. If simply supply voltage and current, the motor will turn. Nonetheless, if some form of control of the motor is needed, such as rotating clock wise (CW), rotating counter clock wise (CCW), rotating at constant speed (velocity), and also rotating at varying speed (acceleration and deceleration) a MCU is required. MCU provides the computation that transforms input data acquired through the sensors and output to the motor.

Motors come in several categories. It is broadly divided into DC motors and AC motors. Within the DC motor category, it is further subdivided into Servo, Brushed DC, Brushless DC, and Stepper motor. Motors alone would only provide very high revolution rotating mechanism. To subject the motor to load, gears are introduced. There are many types of mechanical gears, such as planetary gear, worm gear, etc that transform electrical energy into kinetic energy, measurable in the form of torque, stall torque, Round Per Minute (RPM) and %efficiency. Engineering diagram that accompanies the motor will define the physical measurements of the shaft, size and weight. Very useful when designing custom solution that uses motors. Furthermore, there are several electrical characteristics that must be observed and circuit designed for these parameters. Such parameters are operating voltage, operating current, peak current (under load or reverse suddenly), and stall current. The current can range from tens of **mA** to tens of **A**. Hence, a **motor driver board** is often necessary. One such specimen acquired by the lab, click here for more info.

But what all these translate to? With this knowledge, designing a custom MCU based solution that fits the electrical characteristic (of the moving parts) and the mechanical properties would be.

Want More? Continue reading at Making Things Move, Page128-136

Check out the classic implementations of arduino, motors, and algorithm in SP by DCPE students.

http://www.youtube.com/watch?v=ezcfI4DkVTc

http://www.youtube.com/watch?v=RmRU54z7Ccc

http://www.youtube.com/watch?v=ffyXbnGJ9QQ

http://www.youtube.com/watch?v=VOasE3KyHus

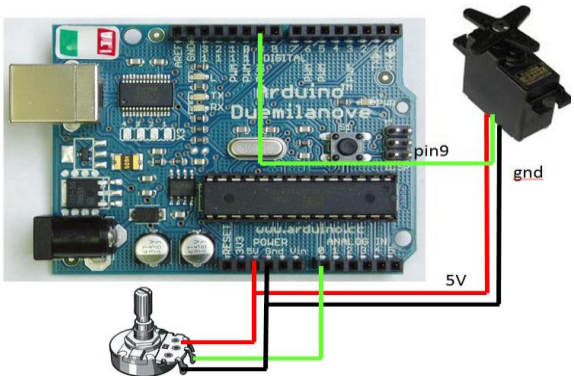
Deliverables

You are required to work in groups of 3to4 to create a stand-up poster that summarizes the electrical and mechanical characteristics of a DC motor and servo motor, which will be used later in the experiments. Refer to the specification sheet from the recommended text book pg130 for a sample motor.

E4: Introduction to making things move on Arduino: Servo and the OOP approach

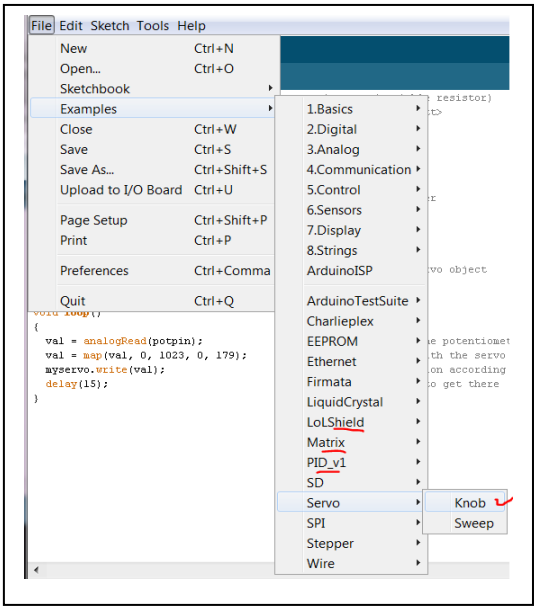
Introduction

This act will discover on how to use the code libraries provided by Arduino with the example on using servo motor. The code libraries are created using an Object Oriented Programming (OOP) approach. In this example, the servo.h library will be used. To use a library, add the following line to the top of the application. For en example **#include <servo.h>** This will import all the servo functionality, allow programmer to create new Stepper objects and call functions already predefined.



Procedure

1. Connect the potentiometer as per procedure 4 of act2.4, either side of the pin on potentiometer can connect to gnd or +5v on arduino, and the other side to the opposite, the middle pin connect to analog pin 0.
2. Connect the servo to the arduino. Red to 5v White to pwm pin 9, Black to gnd
3. **Open the code from File -> Examples -> Servo -> Knob as per the diagram on the R.H.S.**
4. **Compile the code and download to Arduino.**
5. **Turn the knob and observe the output.**




```
#include <Servo.h>//header file to use servo

Servo myservo;// create servo object to control a servo

int potpin = 0;// analog pin used to connect the potentiometer
int val;// variable to read the value from the analog pin

void setup()
{
  myservo.attach(9);// attaches the servo on pin 9 to the servo object
}

void loop()
{
  val = analogRead(potpin);// reads the value of the potentiometer (value between 0 and 1023)
  val = map(val, 0, 1023, 0, 179);// scale it to use it with the servo (value between 0 and 180)
  myservo.write(val);// sets the servo position according to the scaled value
  delay(15);// waits for the servo to get there
}
```

In the diagram above, there might be a different presentation from your computer. Do not be alarmed. The items underscored with red refer to libraries that were added manually (by extracting the files to the **arduino\libraries** folder) to facilitate the development of the respective software modules.

Questions:

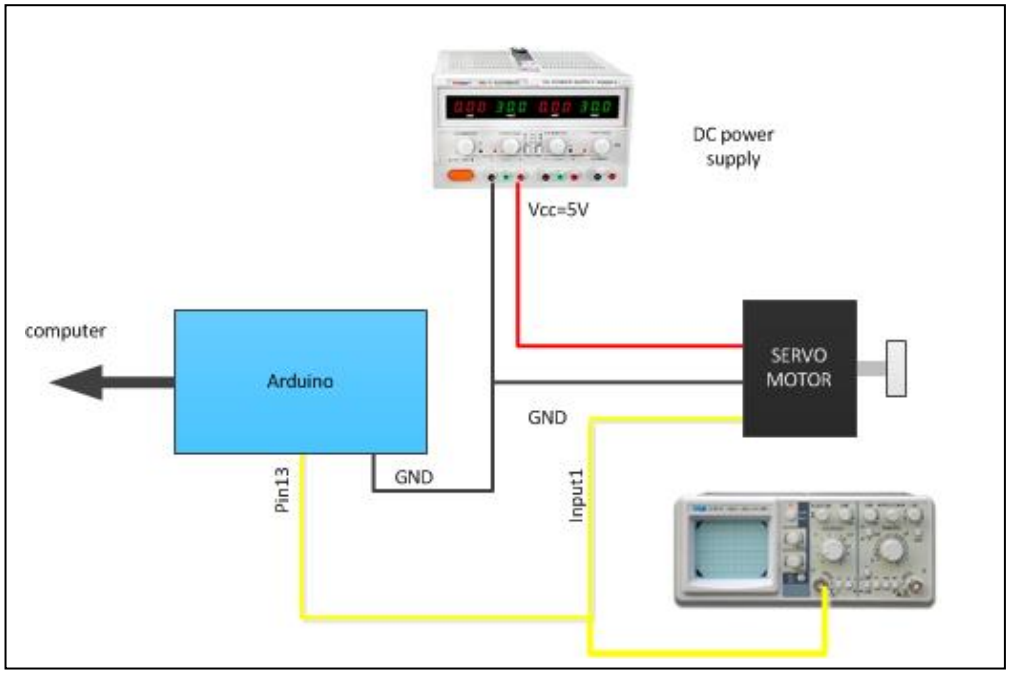
Referring to the Arduino API reference section (link from below), describe the purpose of the functions

1. Servo ____
2. Myservo.attach(__) and Myservo.write(__)
3. What is pulse per width used to drive the servo? **hint: pg 169 of recommended text
4. Draw a diagram to loosely depict the effect of the code *map(val,0,1023,0,179).*

Click to see hacked servo to do full rotation in action http://shin-ajaran.blogspot.sg/2010/04/arduino-with-servo-360deg.html

E4: Observing Servo PPM on oscilloscope

The following diagram describes the setup for observing the servo's pulse per minute using oscilloscope.



E4: Introduction to making things move on Arduino: connect small DC motor Quick & Dirty

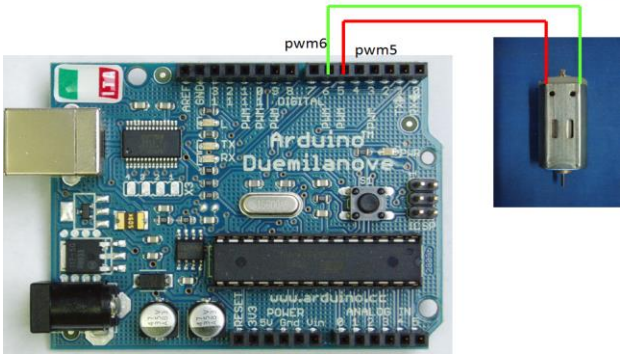
Introduction

Caution: Proceed with Care. **NEVER** connect the DC motor(s) directly to Arduino. There is a reason for power transistors (TIP31C, TIP130, etc) or IC (L293, ULN2003N, L2603, etc) exist to switch the higher voltage and current that the motor takes via the MCU. There is a high chance of **FRYING** the Arduino if one try to connect the motors directly to it and TURN ON. Nonetheless, still want to control a small DC motor bi-directionally without any other hardware such as the ArduMotor, the MotorShield or D&I Lab Bi-Dir DC Motor control?
Example using D&I motor control **PCB (Printed Circuit Board)**
http://shin-ajaran.blogspot.com/2011/10/arduino-bi-directional-dc-motor-control.html
You will need a SMALL DC motor (from EEE store) with small current requirements. Arduino only supply 40mA per digital pin I/O. Regular DC motor will draw much higher current then the pins can provide and this will fry the Arduino. The **BEST** or **PROPER** way will be using driver IC such as the L298, and opto-coupler or SSR to isolate the MCU I/O.
Still want to connect it Quick & Dirty ??

Procedure

1. Connect a small DC motor +ve to PWM pin5 and –ve to PWM pin6.
2. Browse to the blog listed on the R.H.S to acquire the source code.
3. DO NOT place any load (weights, pinching with your fingers) at the motor shaft or repeat this experiment with another type of motor of higher calibre.

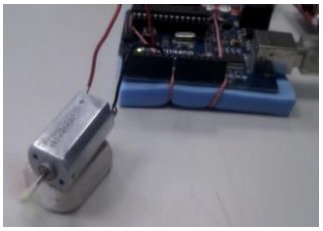
Control small DC motor Quick&Dirty



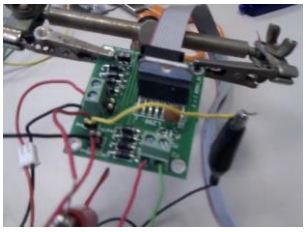
http://shin-ajaran.blogspot.sg/2011/10/arduino-bi-directional-dc-motor-control_10.html

Deliverables

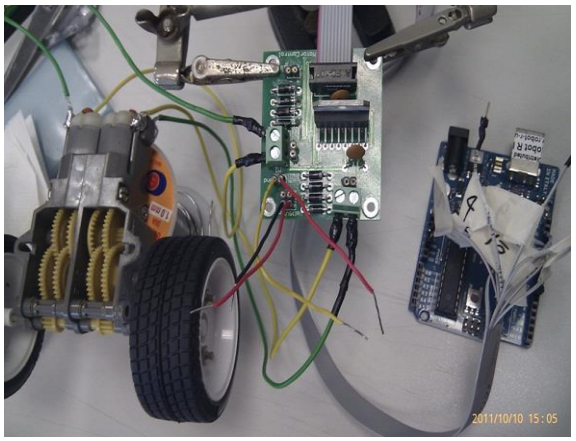
The ONLY deliverables of this experiment is NOT TO fry the Arduino MCU.



http://www.youtube.com/watch?v=AWMdkqXBraU



http://www.youtube.com/watch?v=GSmml-eukiE



http://shin-ajaran.blogspot.com/2011/10/arduino-bi-directional-dc-motor-control.html

Continue your reading on recommended text: Making Things Move, Dustyn Roberts

PWM speed control, pg156-157

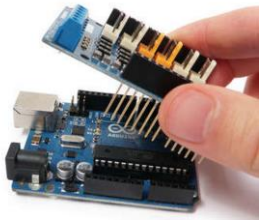
PWM with hardware (potentio meter) and TIP120, pg 158-162

PWM with software and TIP120, pg 163-168

Using PPM (Pulse Per Width) with servo, pg 169-173

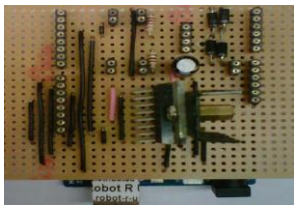
E4: Introduction to making things move on Arduino: motor shield (by DFRobots)

Introduction



Arduino and Shield concept

The Arduino uses an interesting concept to promote flexibility of upgrading or transforming, through the use of the shields. Shields are referring to purpose built and custom made PCB that helps to minimize the hassle of selecting the appropriate electronics components, wiring manually and soldering on a strip board, a nice tidy platform for robotics and mechatronics and hence create a whole new pleasant experience for the novice, artist and people that are not technically trained. Click here for an almost complete list of shields. Usually, these shields come with code library, tutorials and user guides that help the end user to achieve their goal.

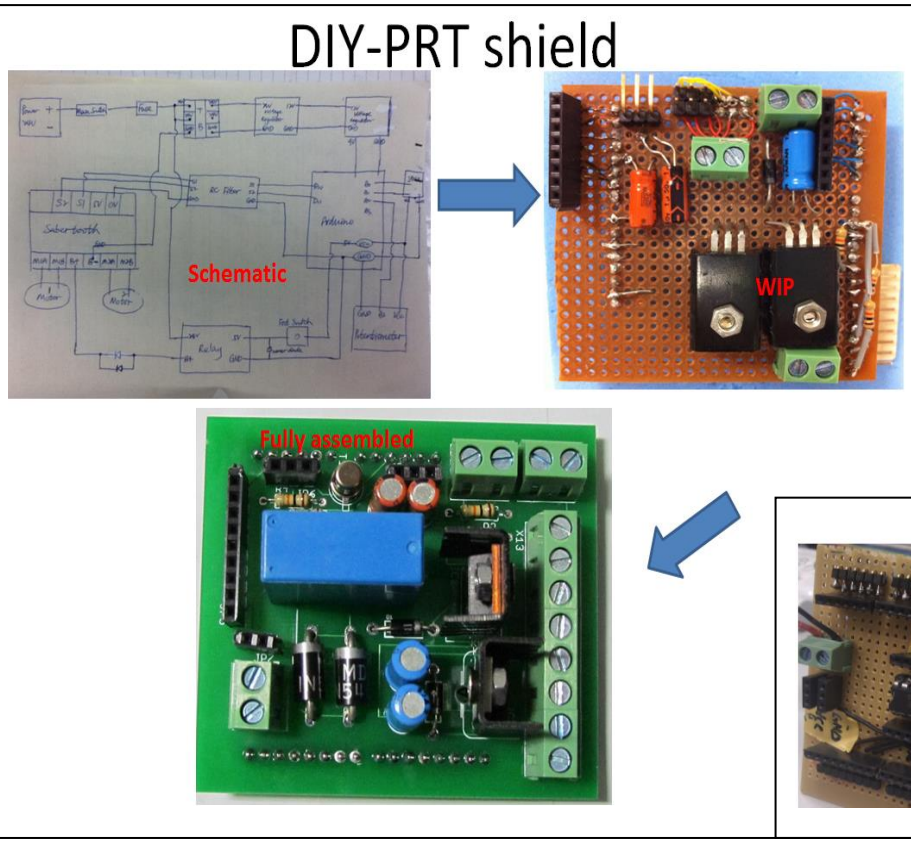


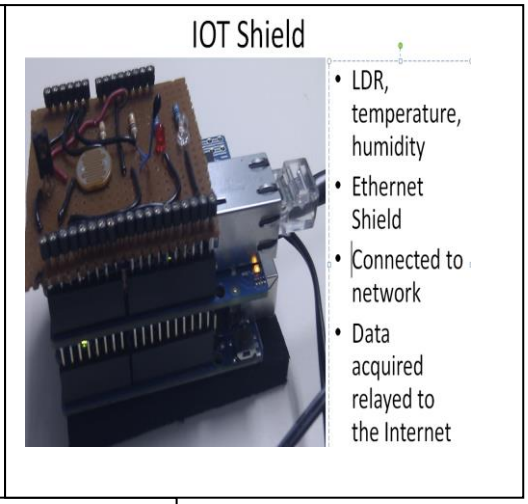
DIY motor shield

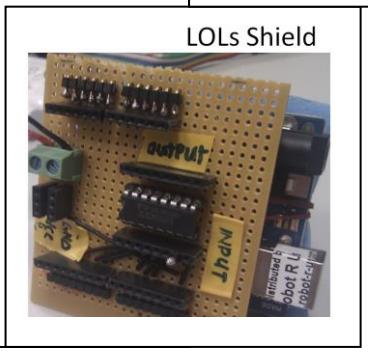
Nonetheless, any technically inclined person could design and fabricate their customized shield. One popular specimen is the motor shield that uses the generic IC L298. Please refer to the attached specification sheet.

Motor Shield allows user to easily control motor direction and speed using an Arduino, by allowing user to simply address Arduino PWN pins. This makes it very simple to incorporate a motor or two into a project. Most of the motor shield allow user to be able to power a motor with a separate power supply of up to 12v or more. It is strongly recommended to use external power source to supply the motors with voltage and current requirements that is above what the Arduino can supply. Best still, the motor shield is very easy to source for with many vendors producing their own flavour of it. Check out Ada Fruit's motor shield, DFR motor shield, sparkfun ArduMoto and Official Arduino Motorshield. Having an Arduino Motor Shield is a must to prototype things that move and general experimenting. With so many vendors providing it, the end users are spoilt for choice. The lab is well stocked with the DFR motor shield for generic DC motor usage.

Some specimens of shields created in SP by staffs and students.



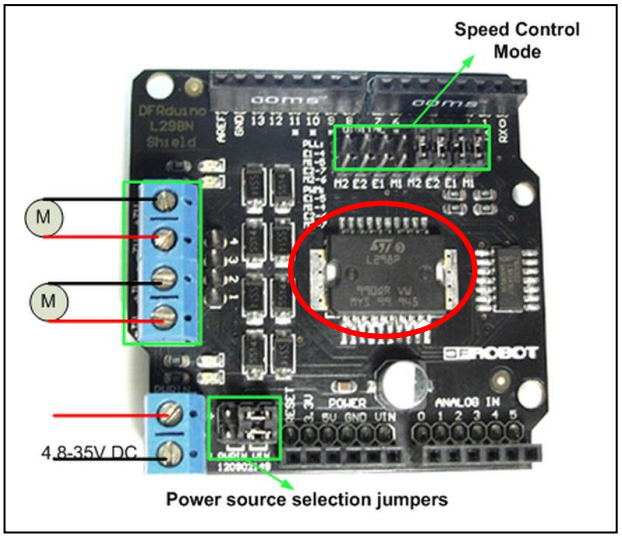




Parts list

1x DFRobot Motor Shield http://www.dfrobot.com/wiki/index.php?title=Arduino_Motor_Shield_(L298N)_(SKU:DRI0009)
1x DC motor (6V 1A) http://www.robot-r-us.com/vmchk/motor-brushed/gear-head-motor-6v-751-ratio.html
1x Lab Power supply.
1x Arduino
***Note**: This experiment will use external lab power supply to the motor shield to power the motor.

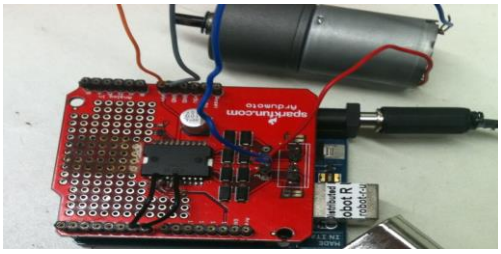
Procedure



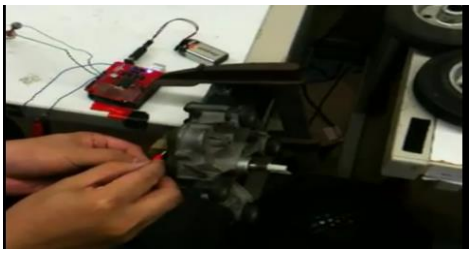
1. Refer to the DFR motor shield wiki, observe the electrical and electronic parameters. List down the parameters here.
2. Refer to the DC motor website, list down the electrical and mechanical properties here.
3. Configure the **Speed Control** jumpers to use PWM.
4. Configure the **Power Source** jumpers to use External.
5. Connect the motor's leads to the motor shield as per the diagram on the R.H.S. G-Clamp motor if necessary.
6. Turn on the lab power supply and set to 6V, 0.5A and **TURN IT OFF** before connecting the +ve and –ve leads to the motor shield. **Caution**: Take note on the polarity of the external power supply. Reversed polarity will damage the motor shield permanently.
7. Acquire the example code from the wiki, compile and download to Arduino.
8. Connect the motor shield to the Arduino.
9. Double check all the wiring, polarity and connections, then turn on the power supply unit.
10. Record your observations. Do not touch IC circle in red, it will be **hot** from diving the motors.

```
//DFRobot motor shield PWM Speed Control:
int E1 = 5;//pwm pin 5
int M1 = 4;//direction pin motor1
int E2 = 6;//pwm pin 6
int M2 = 7;//direction pin motor2

void setup()
{
    pinMode(M1, OUTPUT);
    pinMode(M2, OUTPUT);
}
 void loop()
{
  int value;
  for(value = 0 ; value <= 255; value+=5)
  {
    digitalWrite(M1,HIGH);
    digitalWrite(M2, HIGH);
    analogWrite(E1, value);   //PWM Speed Control
    analogWrite(E2, value);   //PWM Speed Control
    delay(30);
  }
}
```

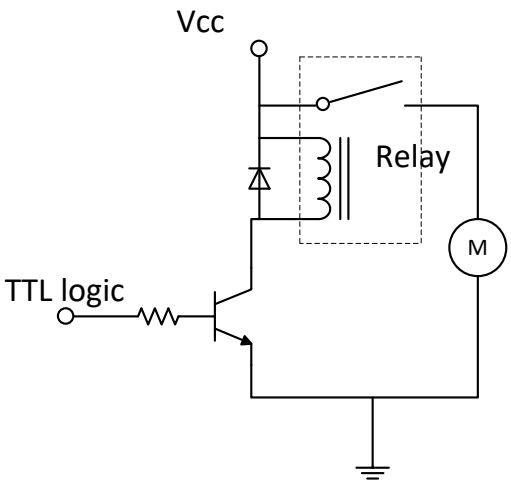


Final setup will be similar to this. Shown here with Sparkfun Ardumotor (red) and 6V DC motor



Ardumotor drivng a car wiper motor with 12V 1A
http://www.youtube.com/watch?v=M-4UPNZJmXI

E4: Control DC motor with relay as a switch



Briefly explain the working principle of above circuit. Is it possible to change the rotation of DC motor?

What is the role of the diode in the above circuit?

Describe the advantage(s) and disadvantage(s) of using a relay as opposed to BJT to control a motor.

Questions

DC clamp multi meter

1. Modify the source code such that it is able to **demonstrate**
    a. The motor to turn CW, stop for 3seconds then turn CCW at a constant speed.
    b. The motor to accelerate in CW direction
    c. The motor to turn CW at highest speed and then decelerate gradually.
2. With the use of a DC Clamp, **record** the current usage of the motor
    a. Under no load condition
    b. Under the condition 1a, but with sudden change of direction
    c. Lifting up a 9V battery pack
3. Justify the data collected at 2.as compared to the specification sheet of L298, motor driver and motor.
4. Evaluate the current usage, when there are 2 motors instead of one.

Challenge

Design a **simple** mechanism that mimics the action of walking/crawling /grabbing/clawing and etc motion inspired by animals/insects/fish/amphibians, possibly with the use of motor driver, DC motors and etc. **Sketch** your design at the space given below.

http://www.youtube.com/watch?v=bxxmN2sQvjk

http://www.youtube.com/watch?v=3X322SL2UdA

http://www.youtube.com/watch?v=e73T7jMWLNc

http://www.youtube.com/watch?v=_idnua6BEc4