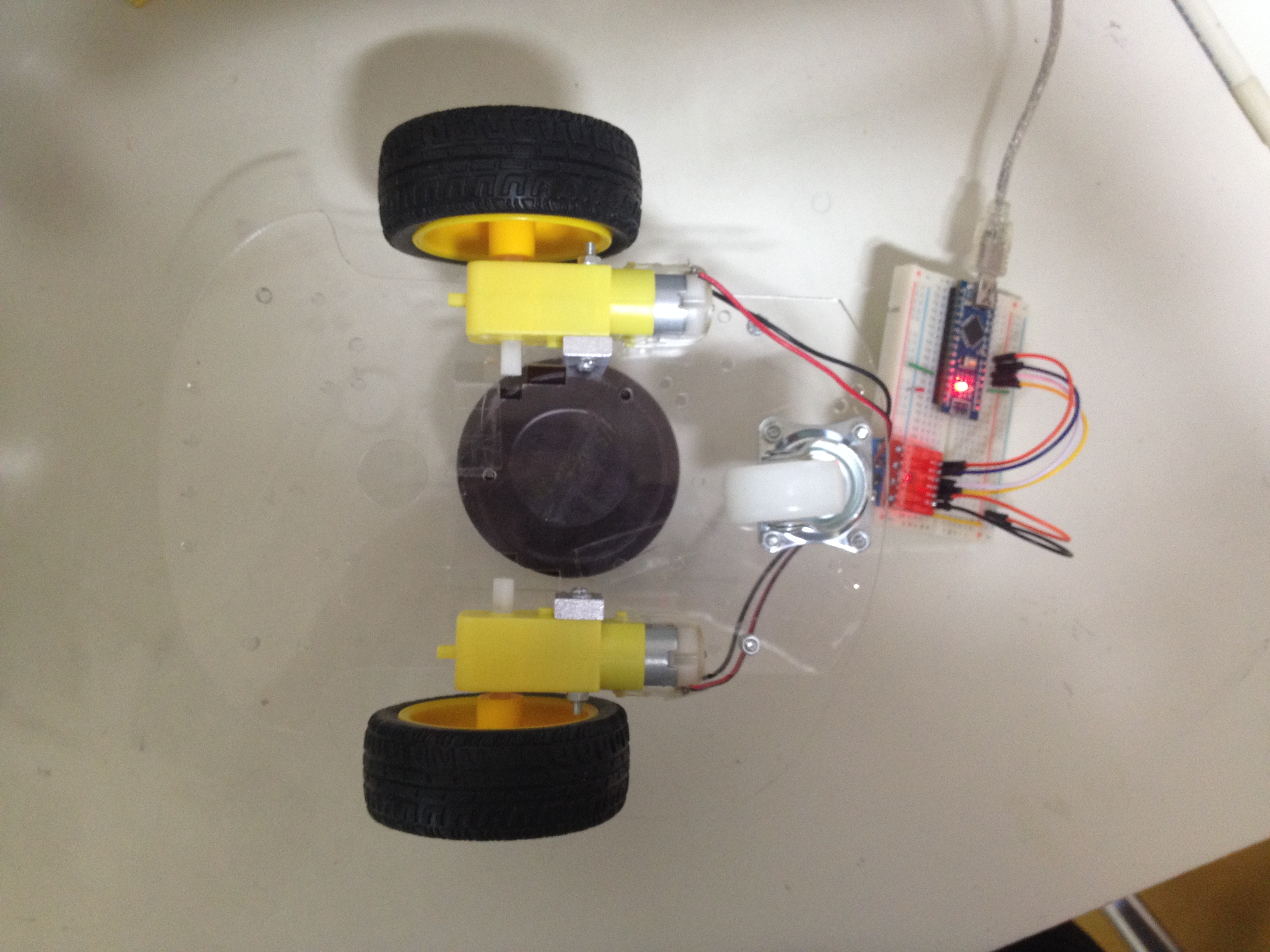
**부품 연결**



[그림 1] 부품 연결 사진

* L9110S 모터 드라이버 모듈의
  + L9110S 모듈의 Gnd 핀을 아두이노의 Gnd 핀에 연결합니다.
  + L9110S 모듈의 Vcc 핀을 아두이노의 5V 핀에 연결합니다.
  + L9110S 모듈의 A-IA 핀을 아두이노의 D5 핀에 연결합니다.
  + L9110S 모듈의 A-IB 핀을 아두이노의 D4 핀에 연결합니다.
  + L9110S 모듈의 B-IA 핀을 아두이노의 D3 핀에 연결합니다.
  + L9110S 모듈의 B-IB 핀을 아두이노의 D2 핀에 연결합니다.
* Motor-A 단자에 왼쪽 바퀴 모터를 연결합니다. 드라이버를 이용해서 단자에 있는 나사를 풀어서 전선을 집어 넣은 뒤에 다시 나사를 조이시면 됩니다. 왼쪽 오른쪽은 여러분들이 마음대로 선택해도 됩니다.
* Motor-B 단자에 오른쪽 바퀴 모터를 연결합니다.
* 아두이노의 USB 케이블을 PC에 연결하고 아두이노 프로그램(IDE)를 실행합니다.

**일단 모터를 돌려보기**

부품 연결이 다 되었으면 [소스 1]과 같이 코딩을 하시고 스케치를 업로드 합니다.

[소스 1]

|  |
| --- |
| 1 : **const** **int** AIA = 5;  2 : **const** **int** AIB = 4;  3 : **const** **int** BIA = 3;  4 : **const** **int** BIB = 2;  5 :  6 : **void** setup() {  7 : pinMode(AIA, OUTPUT);  8 : pinMode(AIB, OUTPUT);  9 : pinMode(BIA, OUTPUT);  10 : pinMode(BIB, OUTPUT);  11 :  12 : digitalWrite(AIA, HIGH);  13 : digitalWrite(AIB, LOW);  14 : digitalWrite(BIA, HIGH);  15 : digitalWrite(BIB, LOW);  16 : }  17 :  18 : **void** loop()  19 : {  20 : } |

업로드가 완료되면 자동차가 앞으로 또는 뒤로 가는 방향으로 두 바퀴가 모두 움직이게 됩니다. 만약 이 상태를 앞으로 가도록 하고자 한다면, 모터에 연결된 두 선을 서로 바꾸어서 연결하면 됩니다.

1-4: 사용 될 아두이노 핀 번호를 이해하기 쉽게 L9110S 모듈의 핀 이름과 같은 상수로 지정하였습니다.

7-10: 모든 아두이노 핀을 출력 모드로 지정합니다.

12: L9110S 모듈의 AIA 핀을 HIGH(+) 상태로 변경합니다.

13: L9110S 모듈의 AIB 핀을 LOW(-) 상태로 변경합니다.

12-13: 라인에 의해서, 모터 A의 각 전극에 + 전원과 – 전원이 걸려서 모터가 돌아가기 시작합니다.

14-15: 라인에 의해서 모터 B도 같은 방향으로 돌아가기 시작합니다.

**속도 조절하기**

모터(자동차)의 속도를 조절하기 위해서는 [소스 2]와 같이 아날로그 아웃풋을 사용합니다.

[소스 2]

|  |
| --- |
| 1 : **const** **int** AIA = 5;  2 : **const** **int** AIB = 4;  3 : **const** **int** BIA = 3;  4 : **const** **int** BIB = 2;  5 :  6 : byte speed = 255;  7 :  8 : **void** setup() {  9 : pinMode(AIA, OUTPUT);  10 : pinMode(AIB, OUTPUT);  11 : pinMode(BIA, OUTPUT);  12 : pinMode(BIB, OUTPUT);  13 :  14 : analogWrite(AIA, speed);  15 : analogWrite(AIB, 0);  16 : analogWrite(BIA, speed);  17 : analogWrite(BIB, 0);  18 : }  19 :  20 : **void** loop()  21 : {  22 : } |

달라진 부분만 살펴보겠습니다.

14-17: 라인에 보시면 [소스 1]에서 digitalWrite 함수를 사용하던 것이 analogWrite으로 모두 바뀐 것을 보실 수가 있습니다. 그리고 LOW는 0으로 바뀌어 있습니다. 이 부분은 그대로 LOW라고 하셔도 상관없습니다. LOW는 결국 0의 다른 표현일 뿐입니다. HIGH는 speed로 바뀌어 있습니다. 그리고 6: 라인에서 255라는 숫자가 지정되어있습니다. speed는 0에서 255 사이의 정수 중에 아무 것이나 사용해도 됩니다. 숫자가 높을수록 모터는 빠르게 돌아갑니다.

**방향 바꾸기**

[소스 3]은 전진, 후진, 좌회전, 우회전을 각각의 함수로 표현한 것 입니다. 바퀴의 방향을 바꾸기 위해서 전원의 상태를 바꾸는 것을 유의 깊게 보시기 바랍니다. [소스 3]을 실행하면 바퀴가 전진하다가 2초 후에 후진하고 다시 2초 후에 좌회전, 그리고 마지막으로 우회전을 하게 됩니다.

[소스 3]

|  |
| --- |
| 1 : **const** **int** AIA = 5;  2 : **const** **int** AIB = 4;  3 : **const** **int** BIA = 3;  4 : **const** **int** BIB = 2;  5 :  6 : byte speed = 255;  7 :  8 : **void** setup()  9 : {  10 : pinMode(AIA, OUTPUT);  11 : pinMode(AIB, OUTPUT);  12 : pinMode(BIA, OUTPUT);  13 : pinMode(BIB, OUTPUT);  14 : }  15 :  16 : **void** loop()  17 : {  18 : forward();  19 : delay(2000);  20 :  21 : backward();  22 : delay(2000);  23 :  24 : left();  25 : delay(2000);  26 :  27 : right();  28 : delay(2000);  29 : }  30 :  31 : **void** forward()  32 : {  33 : analogWrite(AIA, speed);  34 : analogWrite(AIB, 0);  35 : analogWrite(BIA, speed);  36 : analogWrite(BIB, 0);  37 : }  38 :  39 : **void** backward()  40 : {  41 : analogWrite(AIA, 0);  42 : analogWrite(AIB, speed);  43 : analogWrite(BIA, 0);  44 : analogWrite(BIB, speed);  45 : }  46 :  47 : **void** left()  48 : {  49 : analogWrite(AIA, speed);  50 : analogWrite(AIB, 0);  51 : analogWrite(BIA, 0);  52 : analogWrite(BIB, speed);  53 : }  54 :  55 : **void** right()  56 : {  57 : analogWrite(AIA, 0);  58 : analogWrite(AIB, speed);  59 : analogWrite(BIA, speed);  60 : analogWrite(BIB, 0);  61 : } |