

Microcontroller for Robot Car



กฤตินันท์ ชอบตรง
krittinunt@gmail.com



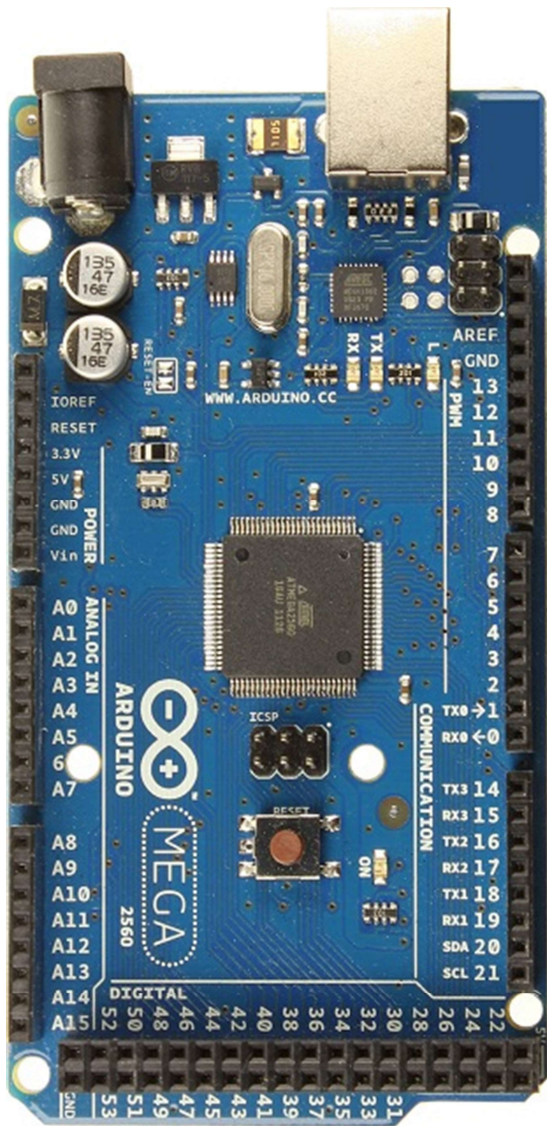
<https://github.com/krittinunt>



<https://krittinunt.medium.com/>



<https://www.youtube.com/chobtrong>



Arduino Mega 2560

Microcontroller: ATmega2560

Operating Voltage: 5V

Input Voltage (recommended): 7-12V

Input Voltage (limit) : 6-20V

Digital I/O Pins: 54 (of which 15 provide PWM output)

Analog Input Pins: 16

DC Current per I/O Pin: 20 mA

DC Current for 3.3V Pin: 50 mA+

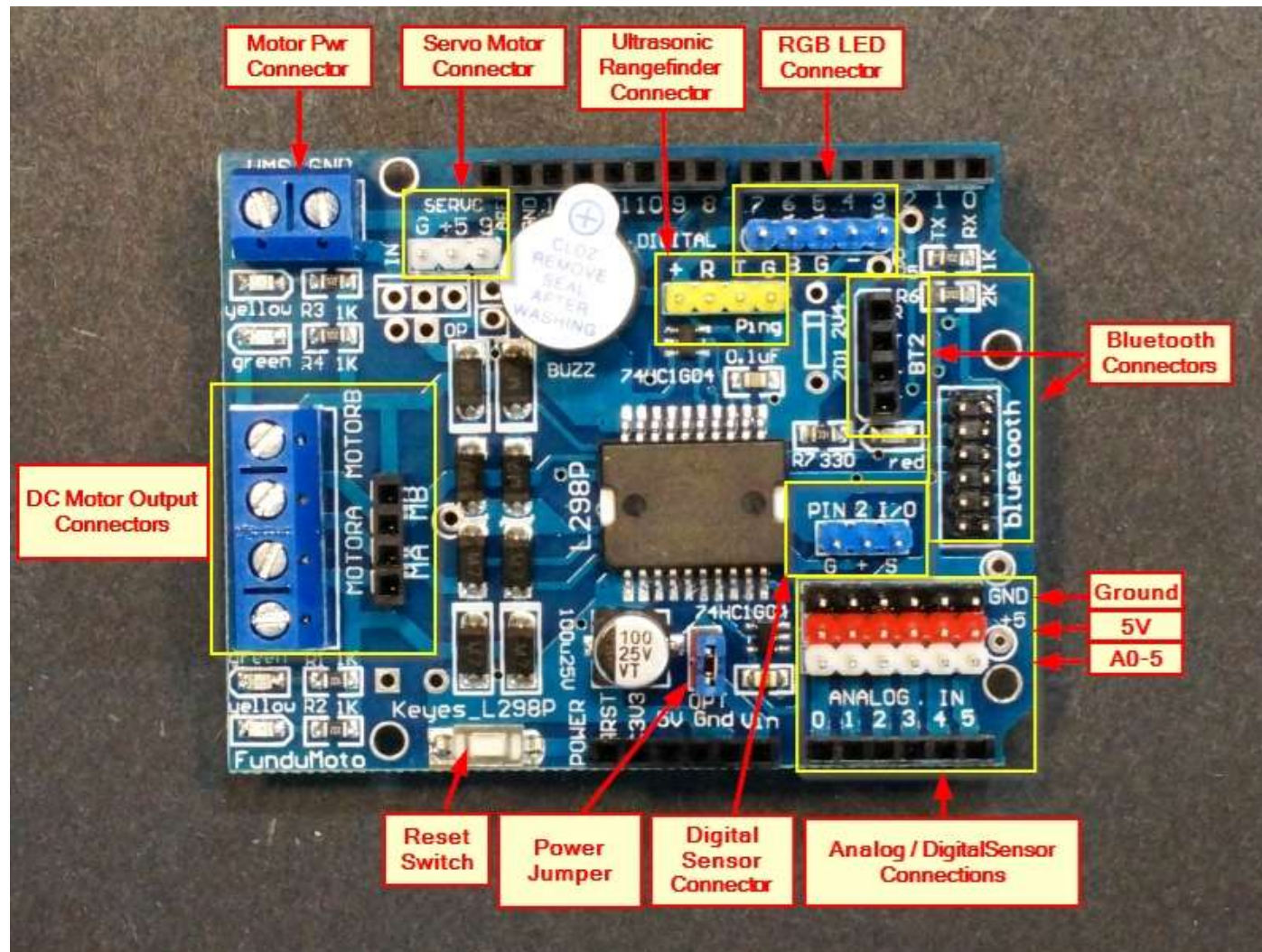
Flash Memory: 256 KB of which 8 KB used by bootloader

SRAM: 8 KB

EEPROM: 4 KB

Clock Speed: 16 MHz

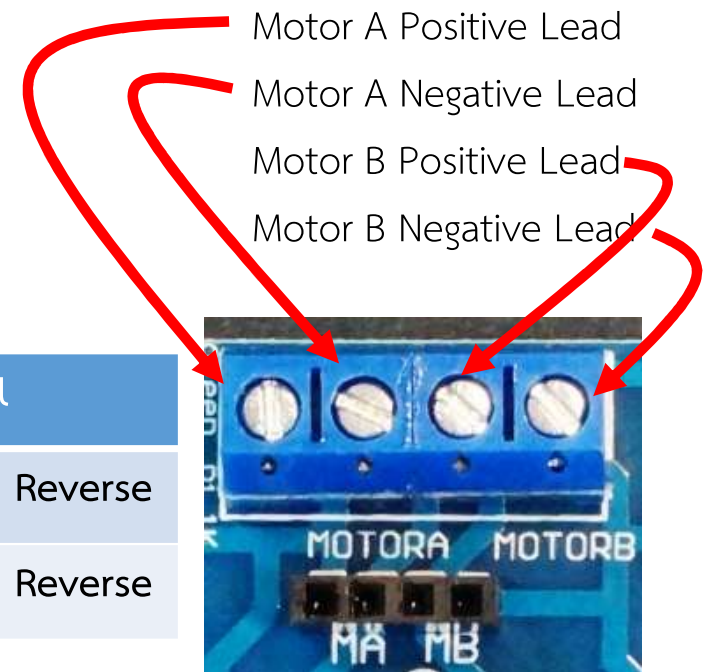
LED_BUILTIN: 13





VMS = Motor Vcc which must be between 4.8 and 24V.

GND = Motor Ground



	Speed Pins	Speed Control	Direction Pins	Direction Control	
Motor A	D10	PWM 0-100	D12	HIGH = Forward	LOW = Reverse
Motor B	D11	PWM 0-100	D13	HIGH = Forward	LOW = Reverse

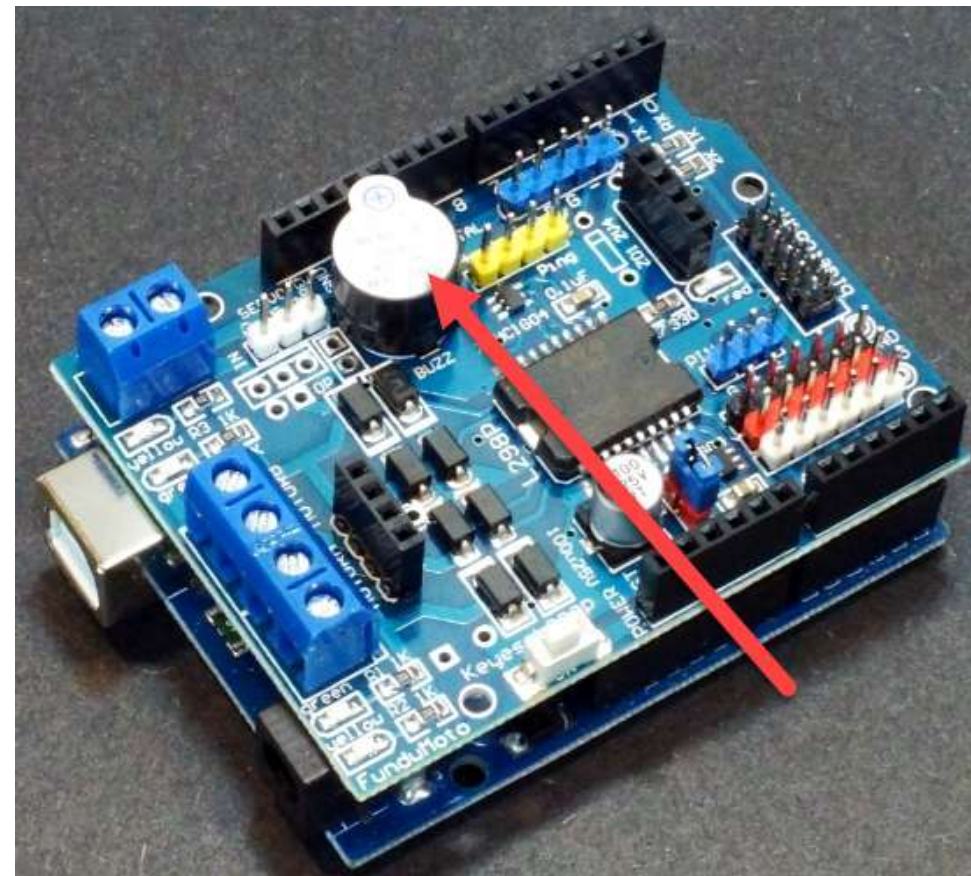
01_Buzzer | Arduino 1.8.13

File Edit Sketch Tools Help



01_Buzzer

```
1 int const BUZZER = 4;
2
3 void setup() {
4   // put your setup code here, to run once:
5   pinMode(BUZZER, OUTPUT);
6
7   // beep beep
8   digitalWrite(BUZZER, HIGH);
9   delay(70);
10  digitalWrite(BUZZER, LOW);
11  delay(50);
12  digitalWrite(BUZZER, HIGH);
13  delay(70);
14  digitalWrite(BUZZER, LOW);
15 }
16
17 void loop() {
18   // put your main code here, to run repeatedly:
19
20 }
```





02_Serial

```
1 int const BUZZER = 4;
2
3 void setup() {
4   // put your setup code here, to run once:
5   Serial.begin(9600);
6   while (!Serial)
7   {
8     ;
9   }
10
11  // digital pin mode
12  pinMode(BUZZER, OUTPUT);
13
14  // beep beep
15  digitalWrite(BUZZER, HIGH);
16  delay(70);
17  digitalWrite(BUZZER, LOW);
18  delay(50);
19  digitalWrite(BUZZER, HIGH);
20  delay(70);
21  digitalWrite(BUZZER, LOW);
22
```

```
23 // Print to serial
24 Serial.println();
25 Serial.println();
26 Serial.println("SRTC Robot ready");
27 Serial.println("=====");
28 }
29
30 void loop() {
31   // put your main code here, to run repeatedly:
32
33 }
```

SRTC Robot ready

=====

1.8.13

is Help

t BUZZER = 4;

up() {

your setup code here, to run o

.begin(9600);

(!Serial)

☒ Autoscroll ☐ Show timestamp

Newline

9600 baud

Clear output



03_Read_IR_sensor

```
1 int const BUZZER = 4;
2
3 int const IR_SENSOR_C = 1;
4
5 float ir_sensor_c_value = 0.0;
6
7 void setup() {
8   // put your setup code here, to run once:
9   Serial.begin(9600);
10  while (!Serial)
11  {
12    ;
13  }
14
15  // digital pin mode
16  pinMode(BUZZER, OUTPUT);
17
18  // beep beep
19  digitalWrite(BUZZER, HIGH);
20  delay(70);
21  digitalWrite(BUZZER, LOW);
22  delay(50);
23  digitalWrite(BUZZER, HIGH);
24  delay(70);
25  digitalWrite(BUZZER, LOW);
26
27  // Print to serial
28  Serial.println();
29  Serial.println();
30  Serial.println("SRTC Robot ready");
31  Serial.println("=====");
32 }
33
34 void loop() {
35   // put your main code here, to run repeatedly:
36
37   ir_sensor_c_value = analogRead(IR_SENSOR_C);
38   Serial.print("IR Sensor value : ");
39   Serial.println(ir_sensor_c_value);
40
41   delay(200);
42 }
```

IR Sensor value : 147.00
IR Sensor value : 147.00
IR Sensor value : 147.00
IR Sensor value : 147.00
IR Sensor value : 147.00
IR Sensor value : 148.00
IR Sensor value : 148.00
IR Sensor value : 148.00
IR Sensor value : 148.00

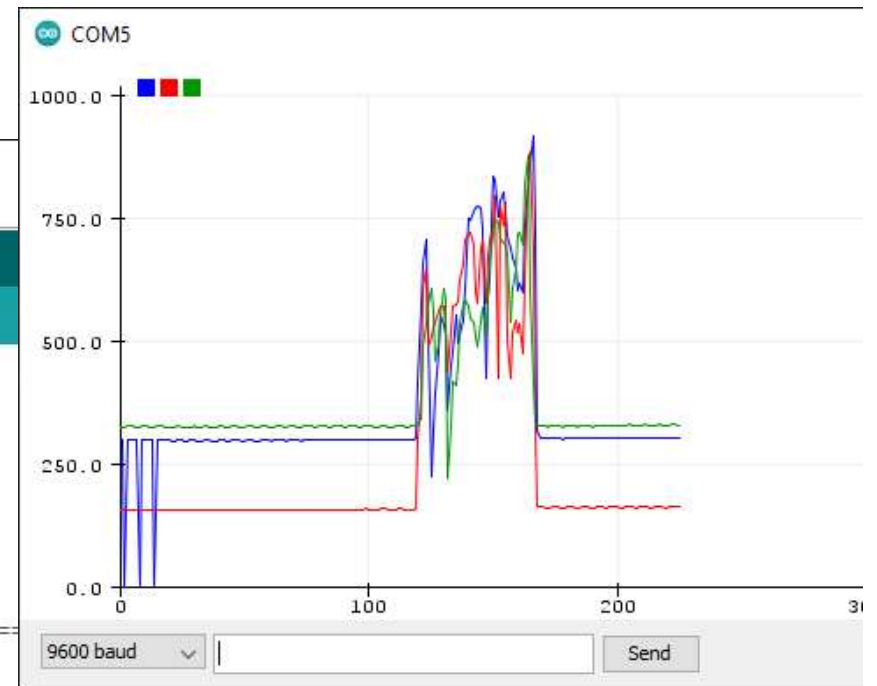
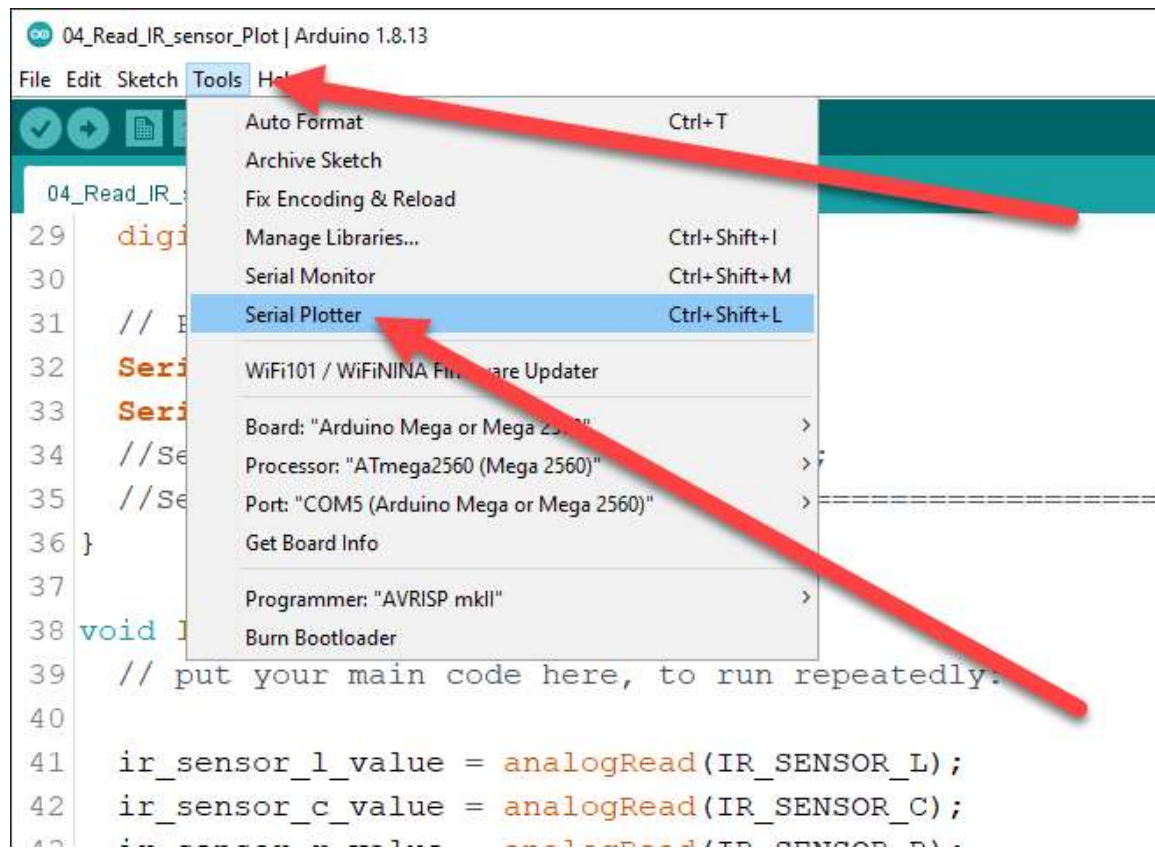
04_Read_IR_sensor_Plot

```
1 int const BUZZER = 4;
2
3 int const IR_SENSOR_L = 0;
4 int const IR_SENSOR_C = 1;
5 int const IR_SENSOR_R = 2;
6
7 float ir_sensor_l_value = 0.0;
8 float ir_sensor_c_value = 0.0;
9 float ir_sensor_r_value = 0.0;
10
11 void setup() {
12     // put your setup code here, to run once:
13     Serial.begin(9600);
14     while (!Serial)
15     {
16         ;
17     }
18
19     // digital pin mode
20     pinMode(BUZZER, OUTPUT);
21
22     // beep beep
23     digitalWrite(BUZZER, HIGH);
24     delay(70);
25     digitalWrite(BUZZER, LOW);
```

```
26     delay(50);
27     digitalWrite(BUZZER, HIGH);
28     delay(70);
29     digitalWrite(BUZZER, LOW);
30
31     // Print to serial
32     Serial.println();
33     Serial.println();
34     //Serial.println("SRTC Robot ready");
35     //Serial.println("=====");
36 }
37
38 void loop() {
39     // put your main code here, to run repeatedly:
40
41     ir_sensor_l_value = analogRead(IR_SENSOR_L);
42     ir_sensor_c_value = analogRead(IR_SENSOR_C);
43     ir_sensor_r_value = analogRead(IR_SENSOR_R);
44
45     //Serial.print("IR Sensor value : ");
46     Serial.print(ir_sensor_l_value);
47     Serial.print(',');
48     Serial.print(ir_sensor_c_value);
49     Serial.print(',');
50     Serial.println(ir_sensor_r_value);
```

```
51 |
52 |   delay(200);
53 | }
```

Done uploading.



Electrical Characteristics

規格特性 Characteristics	代號 Symbol	測試條件 Test conditions	極小 Min.	基準 Ref.	最大 Max.	單位 Units
輸入電壓 Supply voltage	Vcc	- -	3.5	-	20	V
輸出飽和電壓 Output saturation voltage	Vce (sat)	Vcc = 14V ; IC = 20mA	-	300	700	mV
輸出漏電電流 Output leakage current	Icex	Vcc = 14V ; Vcc = 14V	-	< 0.1	10	μA
輸入電流 Supply current	Ice	Vcc = 20V Output open	-	5	10	mA
輸出上升時間 Output rise time	tr	Vcc = 14V ; RL = 820 Ω ; CL = 20pF	-	0.3	1.5	μs
輸出下降時間 Output fall time	tr	Vcc = 14V ; RL = 820 Ω ; CL = 20pF	-	0.3	1.5	μs

● 使用相對溼度：20%~85%RH
Operating relative humidity

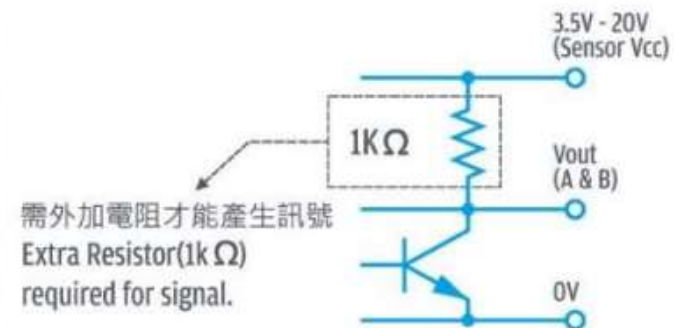
● 使用溫度範圍：-10℃~+60℃
Operating temperature range



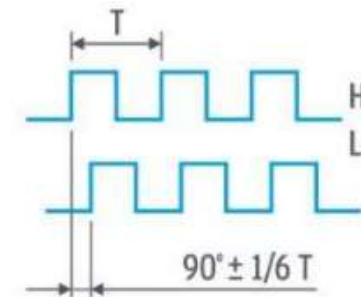
Two Channel Encoder Connections

1. Black : - Motor
2. Red : + Motor
3. Brown : Hall Sensor Vcc
4. Green : Hall Sensor GND
5. Blue : Hall Sensor A Vout
6. Purple : Hall Sensor B Vout

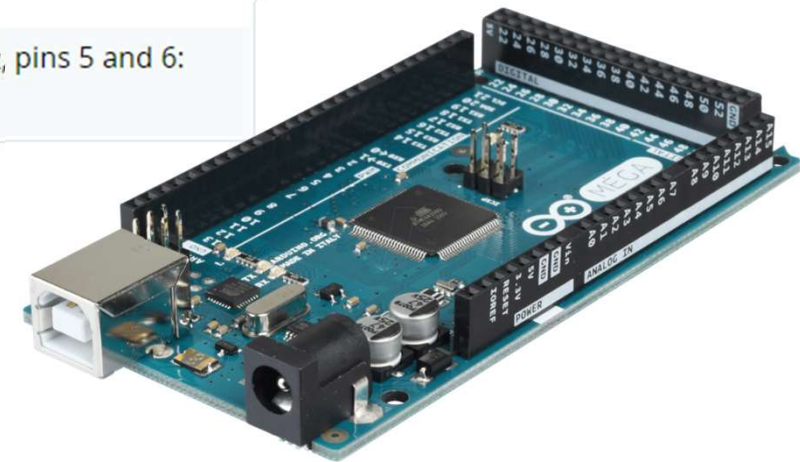
輸出電路 Output circuit

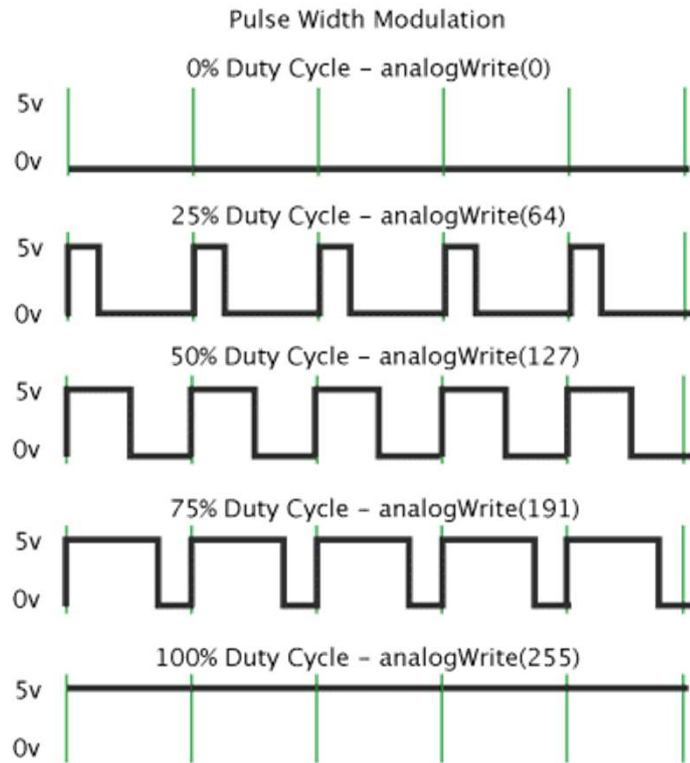


輸出波形 Output wave



BOARD	PWM PINS	PWM FREQUENCY
Uno, Nano, Mini	3, 5, 6, 9, 10, 11	490 Hz (pins 5 and 6: 980 Hz)
Mega	2 - 13, 44 - 46	490 Hz (pins 4 and 13: 980 Hz)
Leonardo, Micro, Yún	3, 5, 6, 9, 10, 11, 13	490 Hz (pins 3 and 11: 980 Hz)
Uno WiFi Rev2, Nano Every	3, 5, 6, 9, 10	976 Hz
MKR boards *	0 - 8, 10, A3, A4	732 Hz
MKR1000 WiFi *	0 - 8, 10, 11, A3, A4	732 Hz
Zero *	3 - 13, A0, A1	732 Hz
Nano 33 IoT *	2, 3, 5, 6, 9 - 12, A2, A3, A5	732 Hz
Nano 33 BLE/BLE Sense	1 - 13, A0 - A7	500 Hz
Due **	2-13	1000 Hz
101	3, 5, 6, 9	pins 3 and 9: 490 Hz, pins 5 and 6: 980 Hz





$$V_{DC} = \frac{VCC \times \text{Duty Cycle}}{100}$$

$$V_{DC} = \frac{12 \times 50}{100}$$

$$V_{DC} = 6 \text{ Volts}$$

```
1126 TCCR0B = (TCCR0B & 0xF8) | 0x04;  
1127 //TCCR2B = (TCCR2B & 0xF8) | 0x06; // set up timer 2 (pin 9, 10) to 122.5Hz  
1128 TCCR4B = (TCCR4B & 0xF8) | 0x04; // set up timer 4 (pin 6, 7, 8) to 112.5Hz  
1129 TCCR5B = (TCCR5B & 0xF8) | 0x04; // set up timer 5 (pin 44, 45, 46) to 112.5Hz  
1130 Timer3.initialize(20000); // set up timer 3 to 20mSec (50Hz)  
1131 Timer3.attachInterrupt(timer3Isr); // enable timer 3 INT  
1132
```



05_Motor_Control

```
1 int const BUZZER = 4;
2
3 int const IR_SENSOR_L = 0;
4 int const IR_SENSOR_C = 1;
5 int const IR_SENSOR_R = 2;
6
7 // EN -> speed, Direction control -> HIGH =
8 // motor A
9 int const MOTOR_A_EN = 10;
10 int const MOTOR_A_DIR = 12;
11
12 // motor B
13 int const MOTOR_B_EN = 11;
14 int const MOTOR_B_DIR = 13;
15
16 float ir_sensor_l_value = 0.0;
17 float ir_sensor_c_value = 0.0;
18 float ir_sensor_r_value = 0.0;
19
20 void setup() {
21     // put your setup code here, to run once:
22     Serial.begin(9600);
23     while (!Serial)
24     {
25         ;
26     }
```

```
26 }
27
28 // digital pin mode
29 pinMode(BUZZER, OUTPUT);
30
31 pinMode(MOTOR_A_EN, OUTPUT);
32 pinMode(MOTOR_A_DIR, OUTPUT);
33
34 // config digital pin
35 digitalWrite(MOTOR_A_DIR, HIGH);
36 analogWrite(MOTOR_A_EN, 0);
37
38 // beep beep
39 digitalWrite(BUZZER, HIGH);
40 delay(70);
41 digitalWrite(BUZZER, LOW);
42 delay(50);
43 digitalWrite(BUZZER, HIGH);
44 delay(70);
45 digitalWrite(BUZZER, LOW);
46
47 // Print to serial
48 //Serial.println();
49 //Serial.println();
50 //Serial.println("SRTC Robot ready");
```



```
51 //Serial.println("=====
52 }
53
54 void loop() {
55 // put your main code here, to run repeatedly:
56
57 ir_sensor_l_value = analogRead(IR_SENSOR_L);
58 ir_sensor_c_value = analogRead(IR_SENSOR_C);
59 ir_sensor_r_value = analogRead(IR_SENSOR_R);
60
61 //Serial.print("IR Sensor value : ");
62 //Serial.print(ir_sensor_l_value);
63 //Serial.print(',');
64 //Serial.print(ir_sensor_c_value);
65 //Serial.print(',');
66 //Serial.println(ir_sensor_r_value);
67
68 // motor A Forward
69 digitalWrite(MOTOR_A_DIR, HIGH);
70 analogWrite(MOTOR_A_EN, 64);
71 delay(5000);
72
73 // motor A stop
74 analogWrite(MOTOR_A_EN, 0);
75 delay(2000);
76 }
```

07_Drive_Two_Motors

```
1 int const BUZZER = 4;
2
3 int const IR_SENSOR_L = 0;
4 int const IR_SENSOR_C = 1;
5 int const IR_SENSOR_R = 2;
6
7 // EN -> speed, Direction control -> HIGH =
8 // motor A
9 int const MOTOR_A_EN = 10;
10 int const MOTOR_A_DIR = 12;
11
12 // motor B
13 int const MOTOR_B_EN = 11;
14 int const MOTOR_B_DIR = 13;
15
16 float ir_sensor_l_value = 0.0;
17 float ir_sensor_c_value = 0.0;
18 float ir_sensor_r_value = 0.0;
19
20 void setup() {
21     // put your setup code here, to run once:
22     Serial.begin(9600);
23     while (!Serial)
24     {
25         ;
```

```
26     }
27
28     // digital pin mode
29     pinMode(BUZZER, OUTPUT);
30
31     pinMode(MOTOR_A_EN, OUTPUT);
32     pinMode(MOTOR_A_DIR, OUTPUT);
33     pinMode(MOTOR_B_EN, OUTPUT);
34     pinMode(MOTOR_B_DIR, OUTPUT);
35
36     // config digital pin
37     digitalWrite(MOTOR_A_DIR, HIGH);
38     analogWrite(MOTOR_A_EN, 0);
39     digitalWrite(MOTOR_B_DIR, HIGH);
40     analogWrite(MOTOR_B_EN, 0);
41
42     // beep beep
43     digitalWrite(BUZZER, HIGH);
44     delay(70);
45     digitalWrite(BUZZER, LOW);
46     delay(50);
47     digitalWrite(BUZZER, HIGH);
48     delay(70);
49     digitalWrite(BUZZER, LOW);
50
```

```

51 // Print to serial
52 //Serial.println();
53 //Serial.println();
54 //Serial.println("SRTC Robot ready");
55 //Serial.println("=====
56 }
57
58 void loop() {
59 // put your main code here, to run repeatedly:
60
61 ir_sensor_l_value = analogRead(IR_SENSOR_L);
62 ir_sensor_c_value = analogRead(IR_SENSOR_C);
63 ir_sensor_r_value = analogRead(IR_SENSOR_R);
64
65 //Serial.print("IR Sensor value : ");
66 //Serial.print(ir_sensor_l_value);
67 //Serial.print(',');
68 //Serial.print(ir_sensor_c_value);
69 //Serial.print(',');
70 //Serial.println(ir_sensor_r_value);
71
72 // motor A Forward
73 //digitalWrite(MOTOR_A_DIR, HIGH);
74 //analogWrite(MOTOR_A_EN, 64);
75 //delay(5000);

```

```

76
77 // motor A stop
78 //analogWrite(MOTOR_A_EN, 0);
79 //delay(2000);
80
81 // motor A Reverse
82 //digitalWrite(MOTOR_A_DIR, LOW);
83 //analogWrite(MOTOR_A_EN, 64);
84 //delay(5000);
85
86 // motor A stop
87 //analogWrite(MOTOR_A_EN, 0);
88 //delay(2000);
89
90 //delay(200);
91
92 // Drive two motors
93 digitalWrite(MOTOR_A_DIR, HIGH);
94 digitalWrite(MOTOR_B_DIR, HIGH);
95 analogWrite(MOTOR_A_EN, 64);
96 analogWrite(MOTOR_B_EN, 64);
97 delay(5000);
98
99 // Stop two motors
100 analogWrite(MOTOR_A_EN, 0);
101 analogWrite(MOTOR_B_EN, 0);
102 delay(2000);
103 }

```


10_Car_Turn_Left_Half_Drive

```
1 int const BUZZER = 4;
2
3 int const IR_SENSOR_L = 0;
4 int const IR_SENSOR_C = 1;
5 int const IR_SENSOR_R = 2;
6
7 // EN -> speed, Direction control -> HIGH = Forward,
8 // motor A
9 int const MOTOR_A_EN = 10;
10 int const MOTOR_A_DIR = 12;
11
12 // motor B
13 int const MOTOR_B_EN = 11;
14 int const MOTOR_B_DIR = 13;
15
16 float ir_sensor_l_value = 0.0;
17 float ir_sensor_c_value = 0.0;
18 float ir_sensor_r_value = 0.0;
19
20 void setup() {
21     // put your setup code here, to run once:
22     Serial.begin(9600);
23     while (!Serial)
24     {
25         ;
```

```
26     }
27
28     // digital pin mode
29     pinMode(BUZZER, OUTPUT);
30
31     pinMode(MOTOR_A_EN, OUTPUT);
32     pinMode(MOTOR_A_DIR, OUTPUT);
33     pinMode(MOTOR_B_EN, OUTPUT);
34     pinMode(MOTOR_B_DIR, OUTPUT);
35
36     // config digital pin
37     digitalWrite(MOTOR_A_DIR, HIGH);
38     analogWrite(MOTOR_A_EN, 0);
39     digitalWrite(MOTOR_B_DIR, HIGH);
40     analogWrite(MOTOR_B_EN, 0);
41
42     // beep beep
43     digitalWrite(BUZZER, HIGH);
44     delay(70);
45     digitalWrite(BUZZER, LOW);
46     delay(50);
47     digitalWrite(BUZZER, HIGH);
48     delay(70);
49     digitalWrite(BUZZER, LOW);
50
```

```

51 // Print to serial
52 //Serial.println();
53 //Serial.println();
54 //Serial.println("SRTC Robot ready");
55 //Serial.println("=====
56 }
57
58 void loop() {
59 // put your main code here, to run repeatedly:
60
61 ir_sensor_l_value = analogRead(IR_SENSOR_L);
62 ir_sensor_c_value = analogRead(IR_SENSOR_C);
63 ir_sensor_r_value = analogRead(IR_SENSOR_R);
64
65 //Serial.print("IR Sensor value : ");
66 //Serial.print(ir_sensor_l_value);
67 //Serial.print(',');
68 //Serial.print(ir_sensor_c_value);
69 //Serial.print(',');
70 //Serial.println(ir_sensor_r_value);
71
72 // motor A Forward
73 //digitalWrite(MOTOR_A_DIR, HIGH);
74 //analogWrite(MOTOR_A_EN, 64);
75 //delay(5000);

```

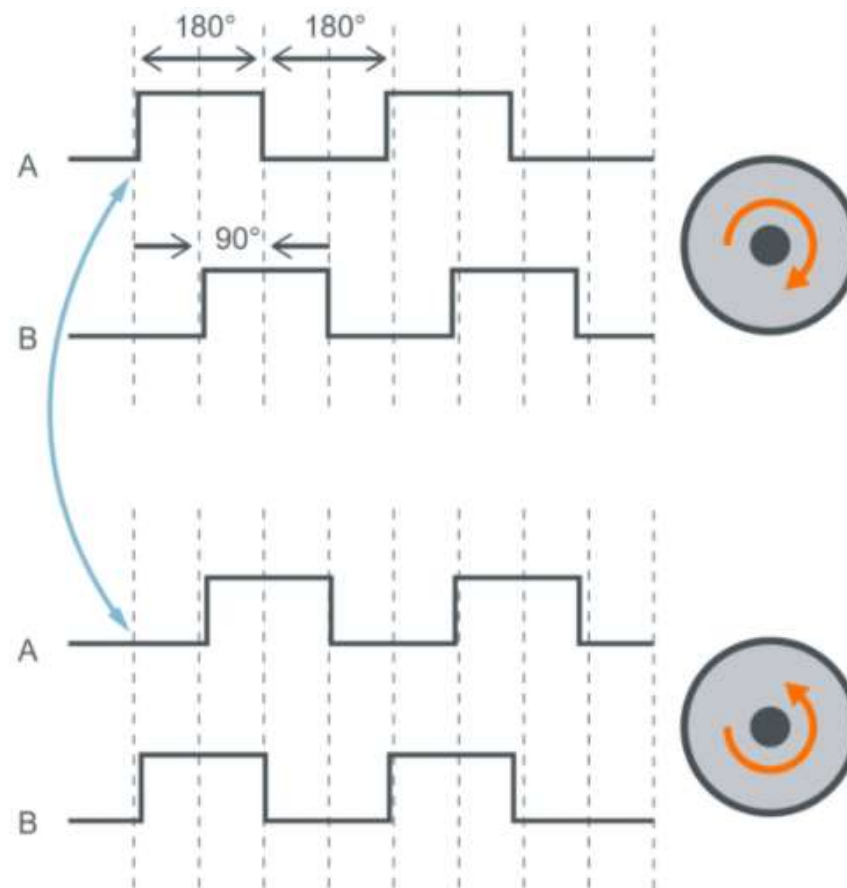
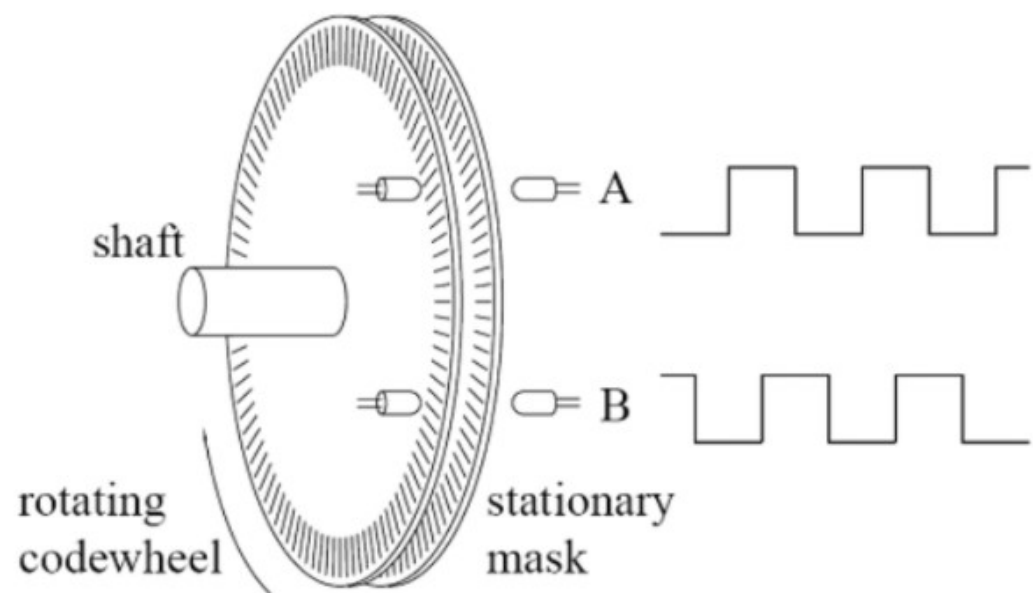
```

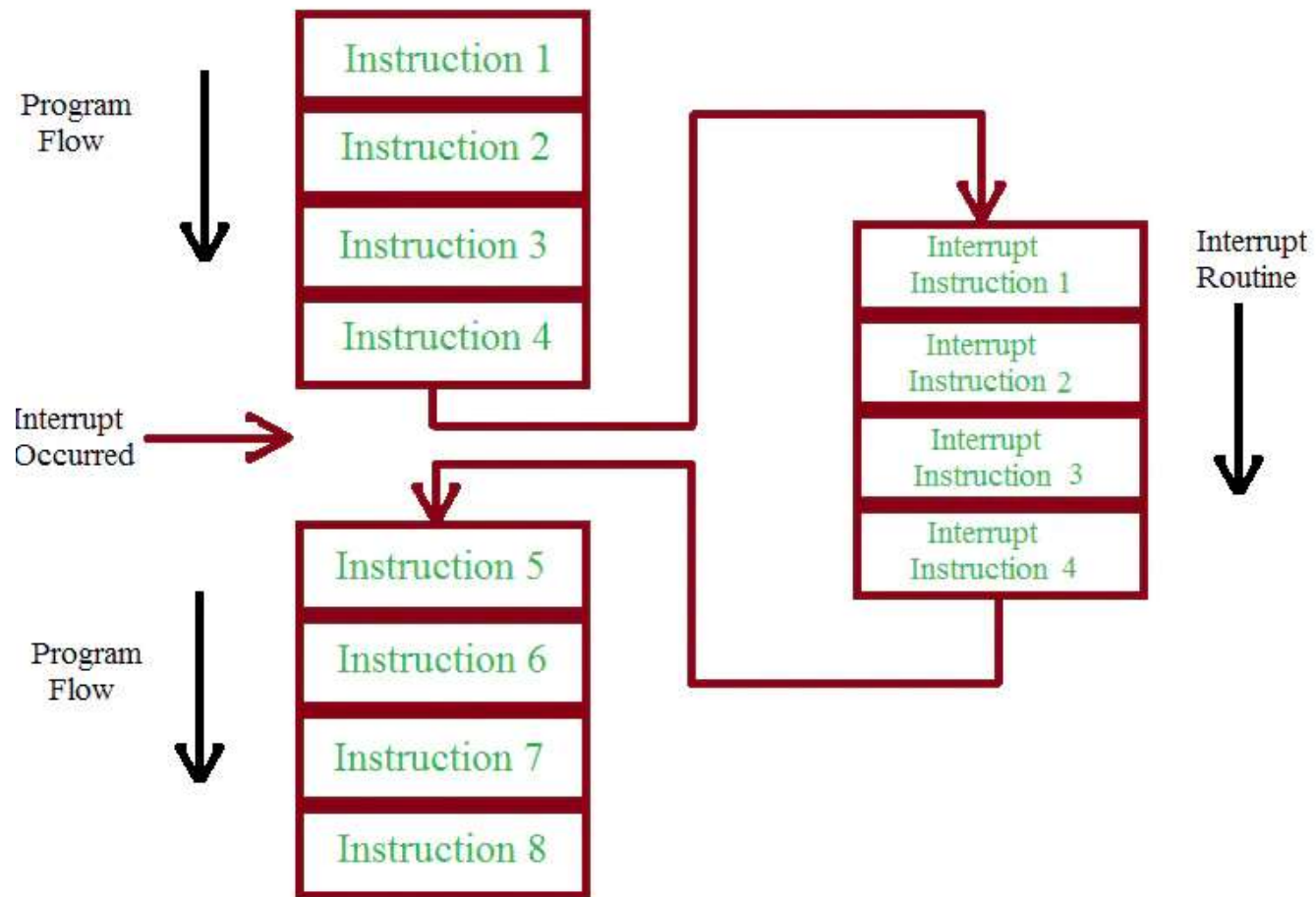
76
77 // motor A stop
78 //analogWrite(MOTOR_A_EN, 0);
79 //delay(2000);
80
81 // motor A Reverse
82 //digitalWrite(MOTOR_A_DIR, LOW);
83 //analogWrite(MOTOR_A_EN, 64);
84 //delay(5000);
85
86 // motor A stop
87 //analogWrite(MOTOR_A_EN, 0);
88 //delay(2000);
89
90 //delay(200);
91
92 // Drive two motors
93 //digitalWrite(MOTOR_A_DIR, HIGH);
94 //digitalWrite(MOTOR_B_DIR, HIGH);
95 //analogWrite(MOTOR_A_EN, 64);
96 //analogWrite(MOTOR_B_EN, 64);
97 //delay(5000);
98
99 // Car Forward
100 //digitalWrite(MOTOR_A_DIR, HIGH);

```

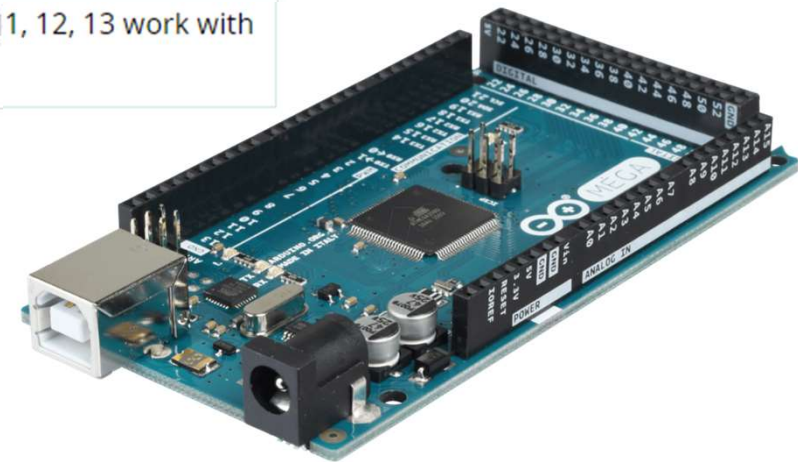
```
101 //digitalWrite(MOTOR_B_DIR, LOW);
102 //analogWrite(MOTOR_A_EN, 64);
103 //analogWrite(MOTOR_B_EN, 64);
104 //delay(5000);
105
106 // Car Reverse
107 //digitalWrite(MOTOR_A_DIR, LOW);
108 //digitalWrite(MOTOR_B_DIR, HIGH);
109 //analogWrite(MOTOR_A_EN, 64);
110 //analogWrite(MOTOR_B_EN, 64);
111 //delay(5000);
112
113 // Car Turn Left Half Drive
114 digitalWrite(MOTOR_B_DIR, LOW);
115 analogWrite(MOTOR_A_EN, 0);
116 analogWrite(MOTOR_B_EN, 64);
117 delay(5000);
118
119 // Stop two motors
120 analogWrite(MOTOR_A_EN, 0);
121 analogWrite(MOTOR_B_EN, 0);
122 delay(2000);
123 }
```

```
113 // Car Turn Left Half Drive
114 //digitalWrite(MOTOR_B_DIR, LOW);
115 //analogWrite(MOTOR_A_EN, 0);
116 //analogWrite(MOTOR_B_EN, 64);
117 //delay(5000);
118
119 // Car Turn Left Full Drive
120 digitalWrite(MOTOR_A_DIR, LOW);
121 digitalWrite(MOTOR_B_DIR, LOW);
122 analogWrite(MOTOR_A_EN, 64);
123 analogWrite(MOTOR_B_EN, 64);
124 delay(5000);
125
126 // Stop two motors
127 analogWrite(MOTOR_A_EN, 0);
128 analogWrite(MOTOR_B_EN, 0);
129 delay(2000);
130 }
```





BOARD	DIGITAL PINS USABLE FOR INTERRUPTS
Uno, Nano, Mini, other 328-based	2, 3
Uno WiFi Rev.2, Nano Every	all digital pins
Mega, Mega2560, MegaADK	2, 3, 18, 19, 20, 21
Micro, Leonardo, other 32u4-based	0, 1, 2, 3, 7
Zero	all digital pins, except 4
MKR Family boards	0, 1, 4, 5, 6, 7, 8, 9, A1, A2
Nano 33 IoT	2, 3, 9, 10, 11, 13, 15, A5, A7
Nano 33 BLE, Nano 33 BLE Sense	all pins
Due	all digital pins
101	all digital pins (Only pins 2, 5, 7, 8, 10, 11, 12, 13 work with CHANGE)



14_Motor_Encoder

```

1 int const BUZZER = 4;
2
3 int const IR_SENSOR_L = 0;
4 int const IR_SENSOR_C = 1;
5 int const IR_SENSOR_R = 2;
6
7 // EN -> speed, Direction control -> |
8 // motor A
9 int const MOTOR_A_EN = 10;
10 int const MOTOR_A_DIR = 12;
11 int const MOTOR_A_SENSOR_A = 18;
12 int const MOTOR_A_SENSOR_B = 19;
13
14 // motor B
15 int const MOTOR_B_EN = 11;
16 int const MOTOR_B_DIR = 13;
17
18 float ir_sensor_l_value = 0.0;
19 float ir_sensor_c_value = 0.0;
20 float ir_sensor_r_value = 0.0;
21
22 int i;
23 double motor_A_pos = 0;
24
25 void motor_A_encoder_A(void);

```

```

26 void motor_A_encoder_B(void);
27
28 void setup() {
29     // put your setup code here, to run once:
30     Serial.begin(9600);
31     while (!Serial)
32     {
33         ;
34     }
35
36     // digital pin mode
37     pinMode(BUZZER, OUTPUT);
38
39     pinMode(MOTOR_A_EN, OUTPUT);
40     pinMode(MOTOR_A_DIR, OUTPUT);
41
42     pinMode(MOTOR_A_SENSOR_A, INPUT_PULLUP);
43     pinMode(MOTOR_A_SENSOR_B, INPUT_PULLUP);
44
45     // config digital pin
46     digitalWrite(MOTOR_A_DIR, HIGH);
47     analogWrite(MOTOR_A_EN, 0);
48
49     digitalWrite(MOTOR_A_SENSOR_A, HIGH);
50     digitalWrite(MOTOR_A_SENSOR_B, HIGH);

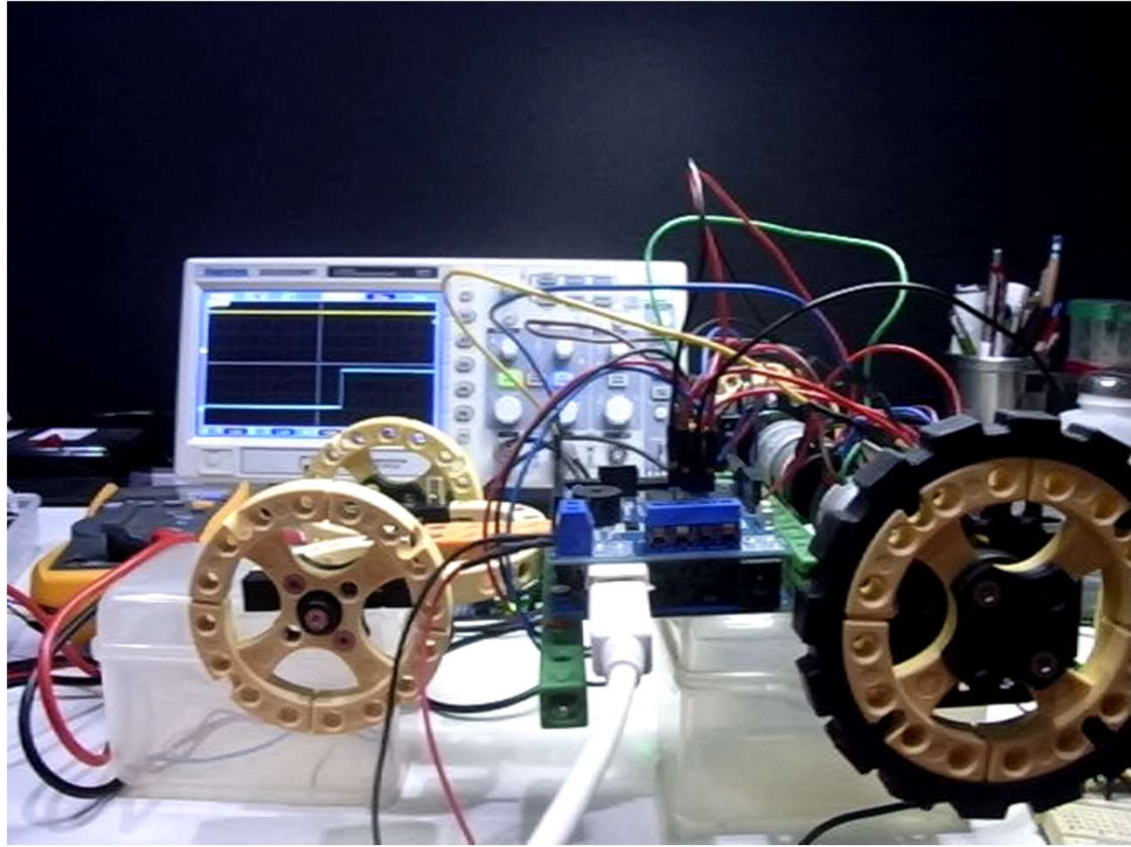
```

```
51
52 // Interrupt
53 attachInterrupt(digitalPinToInterrupt(MOTOR_A_SENSOR_A), motor_A_encoder_A, CHANGE);
54 attachInterrupt(digitalPinToInterrupt(MOTOR_A_SENSOR_B), motor_A_encoder_B, CHANGE);
55
56 // beep beep
57 digitalWrite(BUZZER, HIGH);
58 delay(70);
59 digitalWrite(BUZZER, LOW);
60 delay(50);
61 digitalWrite(BUZZER, HIGH);
62 delay(70);
63 digitalWrite(BUZZER, LOW);
64
65 // Print to serial
66 //Serial.println();
67 //Serial.println();
68 //Serial.println("SRTC Robot ready");
69 //Serial.println("=====");
70 }
71
72 void loop() {
73     // put your main code here, to run repeatedly:
74
75     motor_A_pos = 0;
```



```
76 digitalWrite(MOTOR_A_DIR, HIGH);
77 analogWrite(MOTOR_A_EN, 64);
78 while(motor_A_pos >= -800) delay(1);
79 Serial.println(motor_A_pos);
80 analogWrite(MOTOR_A_EN, 00);
81 digitalWrite(BUZZER, HIGH);
82 delay(100);
83 digitalWrite(BUZZER, LOW);
84 delay(8000);
85 }
86
87 void motor_A_encoder_A(void)
88 {
89     if (digitalRead(MOTOR_A_SENSOR_A) == HIGH)
90     {
91         if (digitalRead(MOTOR_A_SENSOR_B) == LOW) motor_A_pos++; // CW
92         else motor_A_pos--; // CCW
93     }
94     else
95     {
96         if (digitalRead(MOTOR_A_SENSOR_B) == HIGH) motor_A_pos++; // CW
97         else motor_A_pos--; // CCW
98     }
99 }
100
```

```
101 void motor_A_encoder_B(void)
102 {
103     if (digitalRead(MOTOR_A_SENSOR_B) == HIGH)
104     {
105         if (digitalRead(MOTOR_A_SENSOR_A) == HIGH) motor_A_pos++; // CW
106         else motor_A_pos--; // CCW
107     }
108     else
109     {
110         if (digitalRead(MOTOR_A_SENSOR_A) == LOW) motor_A_pos++; // CW
111         else motor_A_pos--; // CCW
112     }
113 }
```



15_Drive_Two_Motor_Encoder

```
1 int const BUZZER = 4;
2
3 int const IR_SENSOR_L = 0;
4 int const IR_SENSOR_C = 1;
5 int const IR_SENSOR_R = 2;
6
7 // EN -> speed, Direction control ->
8 // motor A
9 int const MOTOR_A_EN = 10;
10 int const MOTOR_A_DIR = 12;
11 int const MOTOR_A_SENSOR_A = 18;
12 int const MOTOR_A_SENSOR_B = 19;
13
14 // motor B
15 int const MOTOR_B_EN = 11;
16 int const MOTOR_B_DIR = 13;
17 int const MOTOR_B_SENSOR_A = 20;
18 int const MOTOR_B_SENSOR_B = 21;
19
20 float ir_sensor_l_value = 0.0;
21 float ir_sensor_c_value = 0.0;
22 float ir_sensor_r_value = 0.0;
23
24 int i;
25 double motor_A_pos = 0;
```

```
26 double motor_B_pos = 0;
27
28 void motor_A_encoder_A(void);
29 void motor_A_encoder_B(void);
30 void motor_B_encoder_A(void);
31 void motor_B_encoder_B(void);
32
33 void setup() {
34     // put your setup code here, to run once:
35     Serial.begin(9600);
36     while (!Serial)
37     {
38         ;
39     }
40
41     // digital pin mode
42     pinMode(BUZZER, OUTPUT);
43
44     pinMode(MOTOR_A_EN, OUTPUT);
45     pinMode(MOTOR_A_DIR, OUTPUT);
46
47     pinMode(MOTOR_A_SENSOR_A, INPUT_PULLUP);
48     pinMode(MOTOR_A_SENSOR_B, INPUT_PULLUP);
49     pinMode(MOTOR_B_SENSOR_A, INPUT_PULLUP);
50     pinMode(MOTOR_B_SENSOR_B, INPUT_PULLUP);
```



```
51
52 // config digital pin
53 digitalWrite(MOTOR_A_DIR, HIGH);
54 analogWrite(MOTOR_A_EN, 0);
55 digitalWrite(MOTOR_B_DIR, HIGH);
56 analogWrite(MOTOR_B_EN, 0);
57
58 digitalWrite(MOTOR_A_SENSOR_A, HIGH);
59 digitalWrite(MOTOR_A_SENSOR_B, HIGH);
60 digitalWrite(MOTOR_B_SENSOR_A, HIGH);
61 digitalWrite(MOTOR_B_SENSOR_B, HIGH);
62
63 // Interrupt
64 attachInterrupt(digitalPinToInterrupt(MOTOR_A_SENSOR_A), motor_A_encoder_A, CHANGE);
65 attachInterrupt(digitalPinToInterrupt(MOTOR_A_SENSOR_B), motor_A_encoder_B, CHANGE);
66 attachInterrupt(digitalPinToInterrupt(MOTOR_B_SENSOR_A), motor_B_encoder_A, CHANGE);
67 attachInterrupt(digitalPinToInterrupt(MOTOR_B_SENSOR_B), motor_B_encoder_B, CHANGE);
68
69 // beep beep
70 digitalWrite(BUZZER, HIGH);
71 delay(70);
72 digitalWrite(BUZZER, LOW);
73 delay(50);
74 digitalWrite(BUZZER, HIGH);
75 delay(70);
```

```
76  digitalWrite(BUZZER, LOW);
77
78  // Print to serial
79  //Serial.println();
80  //Serial.println();
81  //Serial.println("SRTC Robot ready");
82  //Serial.println("=====");
83 }
84
85 void loop() {
86   // put your main code here, to run repeatedly:
87
88   motor_A_pos = 0;
89   motor_B_pos = 0;
90   digitalWrite(MOTOR_A_DIR, HIGH);
91   digitalWrite(MOTOR_B_DIR, LOW);
92   analogWrite(MOTOR_A_EN, 64);
93   analogWrite(MOTOR_B_EN, 50);
94   while( (motor_A_pos >= -800) && (motor_B_pos <= 800) )
95   {
96     Serial.print("Motor A Pos = ");
97     Serial.print(motor_A_pos);
98     Serial.print(", Motor B Pos = ");
99     Serial.println(motor_B_pos);
100  }
```

```
101 analogWrite(MOTOR_A_EN, 00);
102 analogWrite(MOTOR_B_EN, 00);
103 Serial.print("Motor A Pos = ");
104 Serial.print(motor_A_pos);
105 Serial.print(", Motor B Pos = ");
106 Serial.println(motor_B_pos);
107 digitalWrite(BUZZER, HIGH);
108 delay(100);
109 digitalWrite(BUZZER, LOW);
110 delay(8000);
111 }
112
113 void motor_A_encoder_A(void)
114 {
115     if (digitalRead(MOTOR_A_SENSOR_A) == HIGH)
116     {
117         if (digitalRead(MOTOR_A_SENSOR_B) == LOW) motor_A_pos++; // CW
118         else motor_A_pos--; // CCW
119     }
120     else
121     {
122         if (digitalRead(MOTOR_A_SENSOR_B) == HIGH) motor_A_pos++; // CW
123         else motor_A_pos--; // CCW
124     }
125 }
```

```
126
127 void motor_A_encoder_B(void)
128 {
129     if (digitalRead(MOTOR_A_SENSOR_B) == HIGH)
130     {
131         if (digitalRead(MOTOR_A_SENSOR_A) == HIGH) motor_A_pos++; // CW
132         else motor_A_pos--; // CCW
133     }
134     else
135     {
136         if (digitalRead(MOTOR_A_SENSOR_A) == LOW) motor_A_pos++; // CW
137         else motor_A_pos--; // CCW
138     }
139 }
140
141 void motor_B_encoder_A(void)
142 {
143     if (digitalRead(MOTOR_B_SENSOR_A) == HIGH)
144     {
145         if (digitalRead(MOTOR_B_SENSOR_B) == LOW) motor_B_pos++; // CW
146         else motor_B_pos--; // CCW
147     }
148     else
149     {
150         if (digitalRead(MOTOR_B_SENSOR_B) == HIGH) motor_B_pos++; // CW
```



```
150     if (digitalRead(MOTOR_B_SENSOR_B) == HIGH) motor_B_pos++; // CW
151     else motor_B_pos--; // CCW
152 }
153 }
154
155 void motor_B_encoder_B(void)
156 {
157     if (digitalRead(MOTOR_B_SENSOR_B) == HIGH)
158     {
159         if (digitalRead(MOTOR_B_SENSOR_A) == HIGH) motor_B_pos++; // CW
160         else motor_B_pos--; // CCW
161     }
162     else
163     {
164         if (digitalRead(MOTOR_B_SENSOR_A) == LOW) motor_B_pos++; // CW
165         else motor_B_pos--; // CCW
166     }
167 }
```

16_Free_Run

```
85 void loop() {
86   // put your main code here, to run repeatedly:
87
88   // fw
89   motor_A_pos = 0;
90   motor_B_pos = 0;
91   digitalWrite(MOTOR_A_DIR, HIGH);
92   digitalWrite(MOTOR_B_DIR, LOW);
93   analogWrite(MOTOR_A_EN, 64);
94   analogWrite(MOTOR_B_EN, 64);
95   while(motor_A_pos >= -10000)
96   {
97     Serial.print("Motor A Pos = ");
98     Serial.print(motor_A_pos);
99     Serial.print(", Motor B Pos = ");
100    Serial.println(motor_B_pos);
101  }
102  analogWrite(MOTOR_A_EN, 00);
103  analogWrite(MOTOR_B_EN, 00);
104  delay(500);
105 }
```

```
106 // 1
107 motor_A_pos = 0;
108 motor_B_pos = 0;
109 digitalWrite(MOTOR_A_DIR, LOW);
110 digitalWrite(MOTOR_B_DIR, LOW);
111 analogWrite(MOTOR_A_EN, 64);
112 analogWrite(MOTOR_B_EN, 64);
113 while(motor_A_pos <= 1000)
114 {
115   Serial.print("Motor A Pos = ");
116   Serial.print(motor_A_pos);
117   Serial.print(", Motor B Pos = ");
118   Serial.println(motor_B_pos);
119 }
120 analogWrite(MOTOR_A_EN, 00);
121 analogWrite(MOTOR_B_EN, 00);
122 delay(500);
123
124 digitalWrite(BUZZER, HIGH);
125 delay(100);
126 digitalWrite(BUZZER, LOW);
127 while(true)
128 {
129   //
130 }
131 }
```