2022년 IoT기반 스마트 솔루션 개발자 양성과정

Embedded Application

[심화자료]-DC Motor

담당 교수 : 윤 종 이

010-9577-1696

ojo1696@naver.com

https://cafe.naver.com/yoons2022

DC Motor Module

센서 모듈 외형	모듈 항목	모듈 항목의 내용
	모터 드라이버	TB6552
	모터	Micro Type DC Motor
	인터페이스	2x5 Header, 2.54mm Pitch
	동작 전압	5V
	크기	40x60mm
액츄에이터로 활용할 수 있는 DC Motor 모듈		

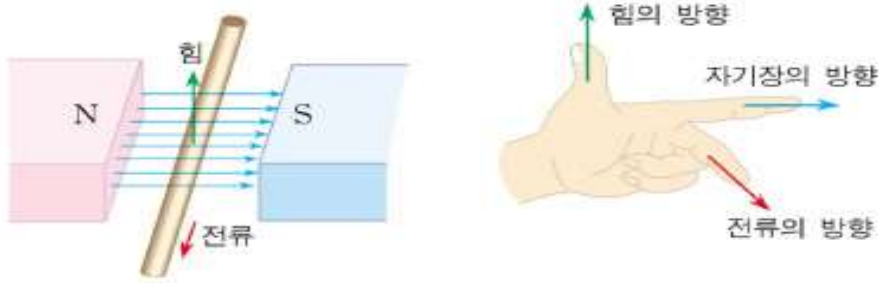
Motor [모터:전동기]

- 모터(Motor, 전동기)는 전력(전기적 에너지)을 이용하여 회전운동의 힘(기계적 에너지)을 얻는 기계
- 전력을 공급하면 전동기의 중심 축을 회전함으로써 돌림힘에 의해 작용하는 각종 기계를 연결하여 운전
- Motor 유형
  - DC Motor
  - AC Motor
  - BLDC Motor
  - Step Motor etc.

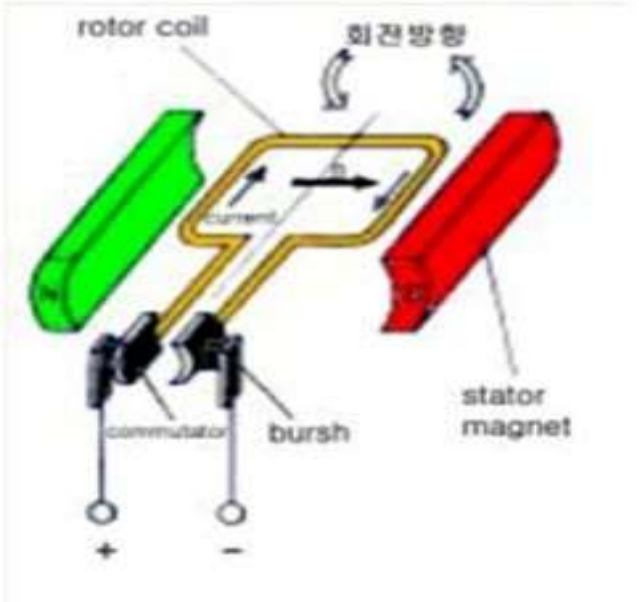
감속기 장착 DC 모터

DC Motor의 구동원리

- 플레밍의 왼손 법칙

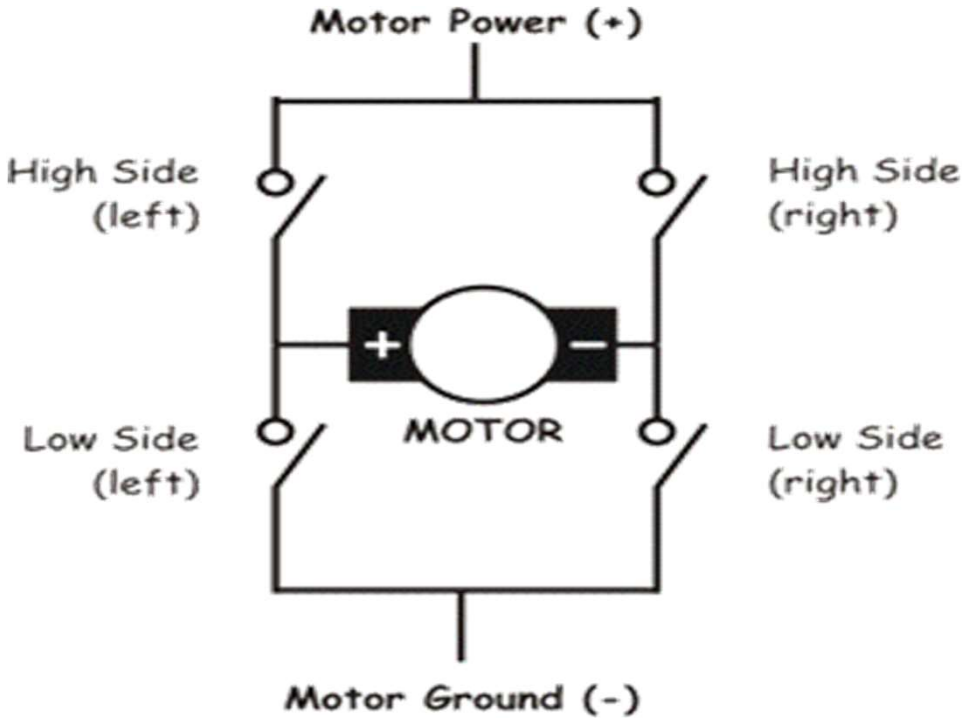


[왼손을 써서 알아보는 방법]



Full Bridge 회로

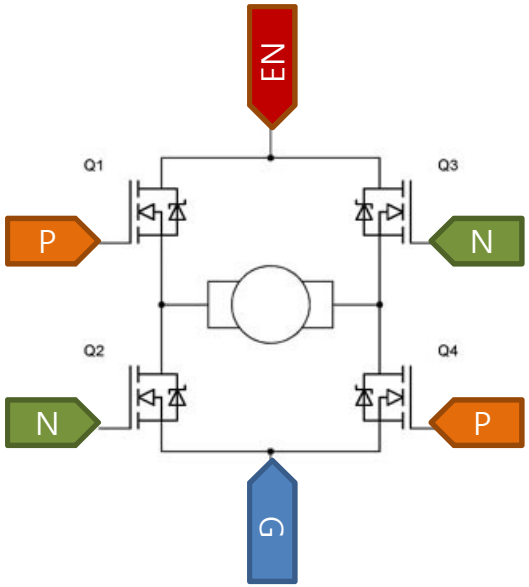
- 정회전
  - High Side (Left)
  - Low Side (right)
- 역회전
  - High Side(right)
  - Low Side(left)



TB6552

Input/Output Function (common for channel A and B)

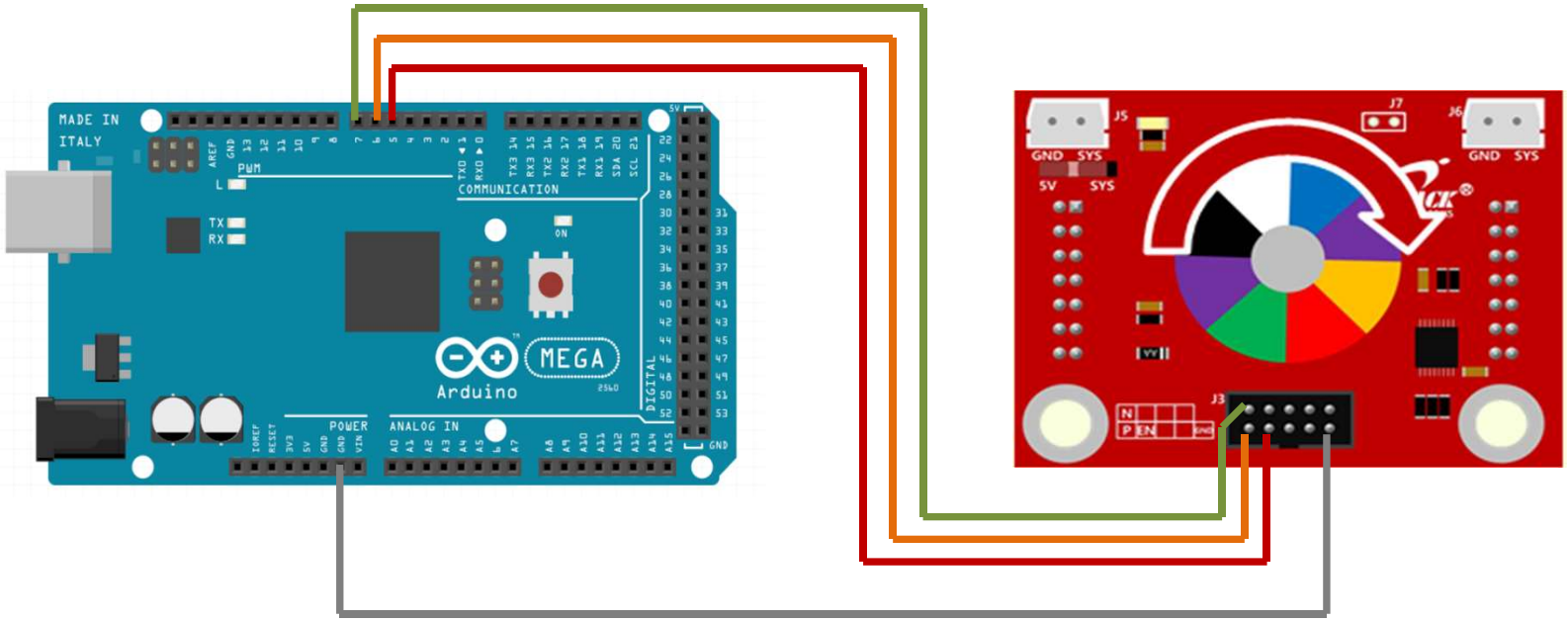
Input				Output		
IN1	IN2	STBY	PWM	O1	O2	Mode
H	H	H	H	L	L	Short brake
			L			
L	H	H	H	L	H	CW/CCW
			L	L	L	Short brake
H	L	H	H	H	L	CCW/CW
			L	L	L	Short brake
L	L	H	H	OFF (high impedance)		Stop
			L			
H/L	H/L	L	H	OFF (high impedance)		Standby
			L			



모터 제어

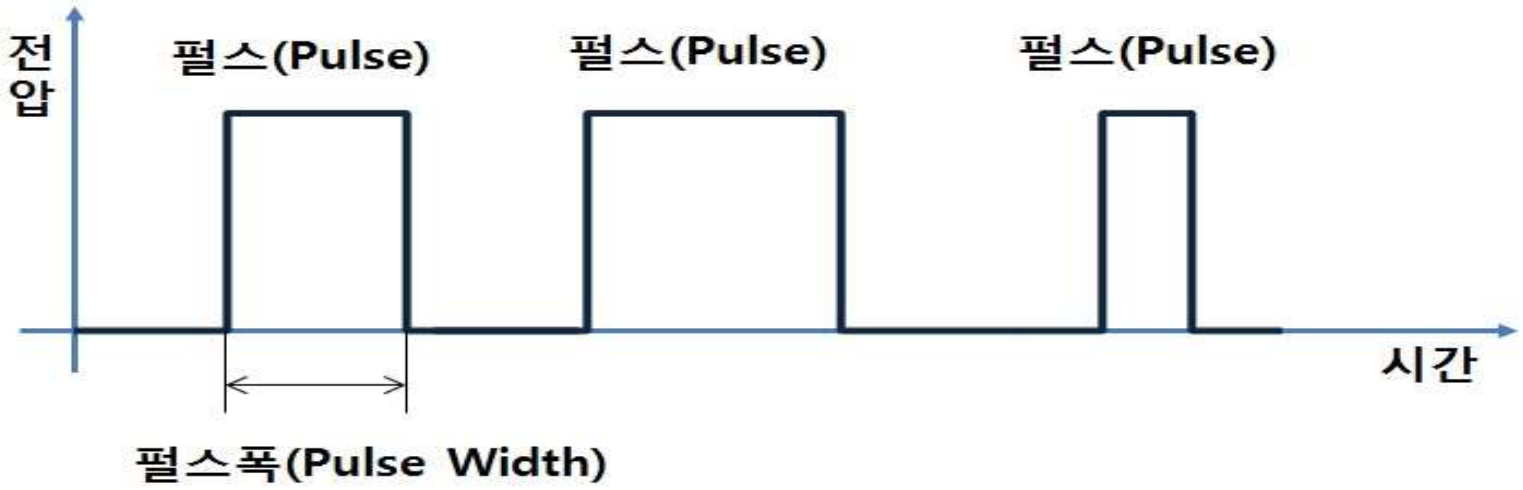
- DC 모터를 제어하기 위해서 전원을 공급
- 모터의 극성을 바꾸어 줌으로써 회전 방향을 조절
- 모터의 회로를 살펴보면 각 전극에 신호를 전달할 수 있는 핀을 확인할 수 있으며, 특정 핀에 신호를 전달하여 제어
- PWM을 사용하면 모터의 속도도 제어 가능
- 모터에 전달하는 신호는 GPIO를 사용
- 모터와 연결된 특정 핀에 적절한 신호를 전달

Wiring


MADE IN
ITALY
PWM
COMMUNICATION
TX
RX
ON
MEGA
Arduino
POWER
ANALOG IN
DIGITAL
GND
SYS
5V
SYS
GND
SYS

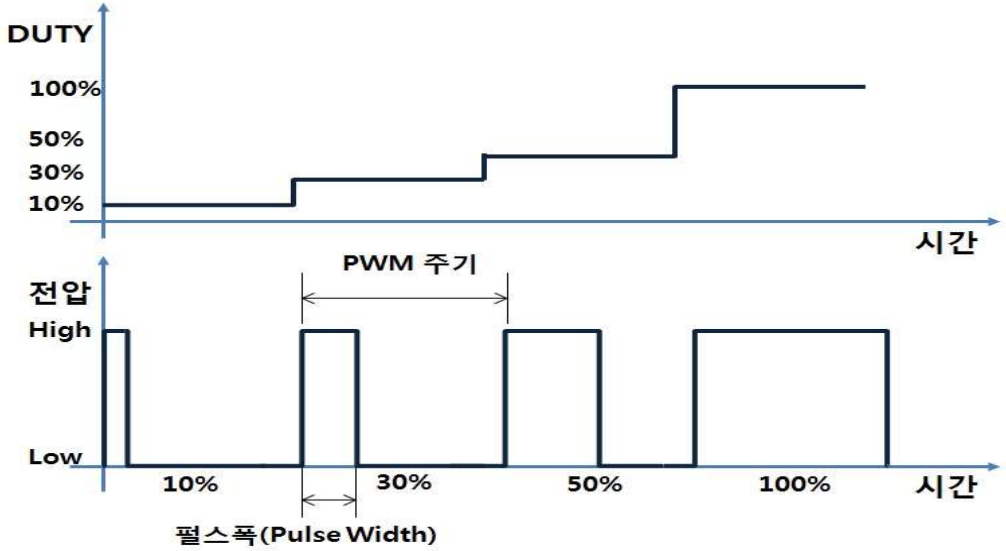
Pulse [펄스]

- 전압이 Low에서 High로, High에서 Low로 변하는 것과 같이 어떤 기준에서 진폭이 신속하고 과도하게 변화하는 것을 펄스라고 함.



PWM

- PWM(Pulse Width Modulation) 펄스 폭 변조
- 펄스 전압의 High/Low를 유지하는 폭을 조절함으로써 해당 시간에 전압이 어느 정도 걸리는 지를 제어하는 방식.
- PWM 주기(단위 시간)에 발생하는 신호를 조절하는 것으로 주기에서 어느 정도의 시간 동안 펄스가 발생했는지를 듀티(Duty)비로 표현



CW / CCW

```
#define Motor_N 7
#define Motor_P 6
#define Motor_EN 5

void setup( ) {
  Serial.begin(9600);
  pinMode(Motor_N,OUTPUT);
  digitalWrite(Motor_N,LOW);
  pinMode(Motor_P,OUTPUT);
  digitalWrite(Motor_P,LOW);
  pinMode(Motor_EN,OUTPUT);
  digitalWrite(Motor_EN,LOW);
}

void loop( ) {
  Motor_CW(150);      delay(1000);
  Motor_STOP();       delay(2000);
  Motor_CCW(150);     delay(1000);
  Motor_STOP();       delay(2000);
}
```

```
void Motor_CW(int Speed){
  digitalWrite(Motor_P,HIGH);
  digitalWrite(Motor_N,LOW);
  analogWrite(Motor_EN,Speed);
}

void Motor_CCW(int Speed){
  digitalWrite(Motor_P,LOW);
  digitalWrite(Motor_N,HIGH);
  analogWrite(Motor_EN,Speed);
}

void Motor_STOP(){
  digitalWrite(Motor_P,LOW);
  digitalWrite(Motor_N,LOW);
  digitalWrite(Motor_EN,LOW);
}
```

Serial Motor Control Packet

Head	Motion	Separetor	Speed	Terminator
@	0 : Stop 1 : CW 2 :CCW	,	0~255	₩n

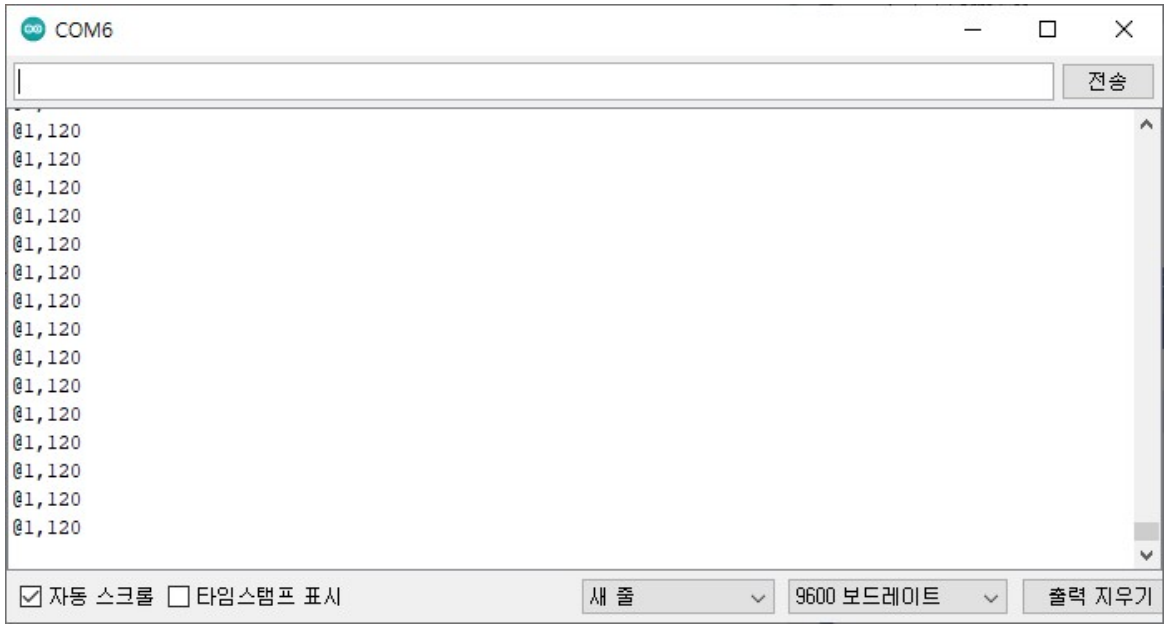
Serial Motor Control

```
int Motion=0; //0:Stop, 1:CW, 2:CCW,
int Speed=0;  //0~255
```

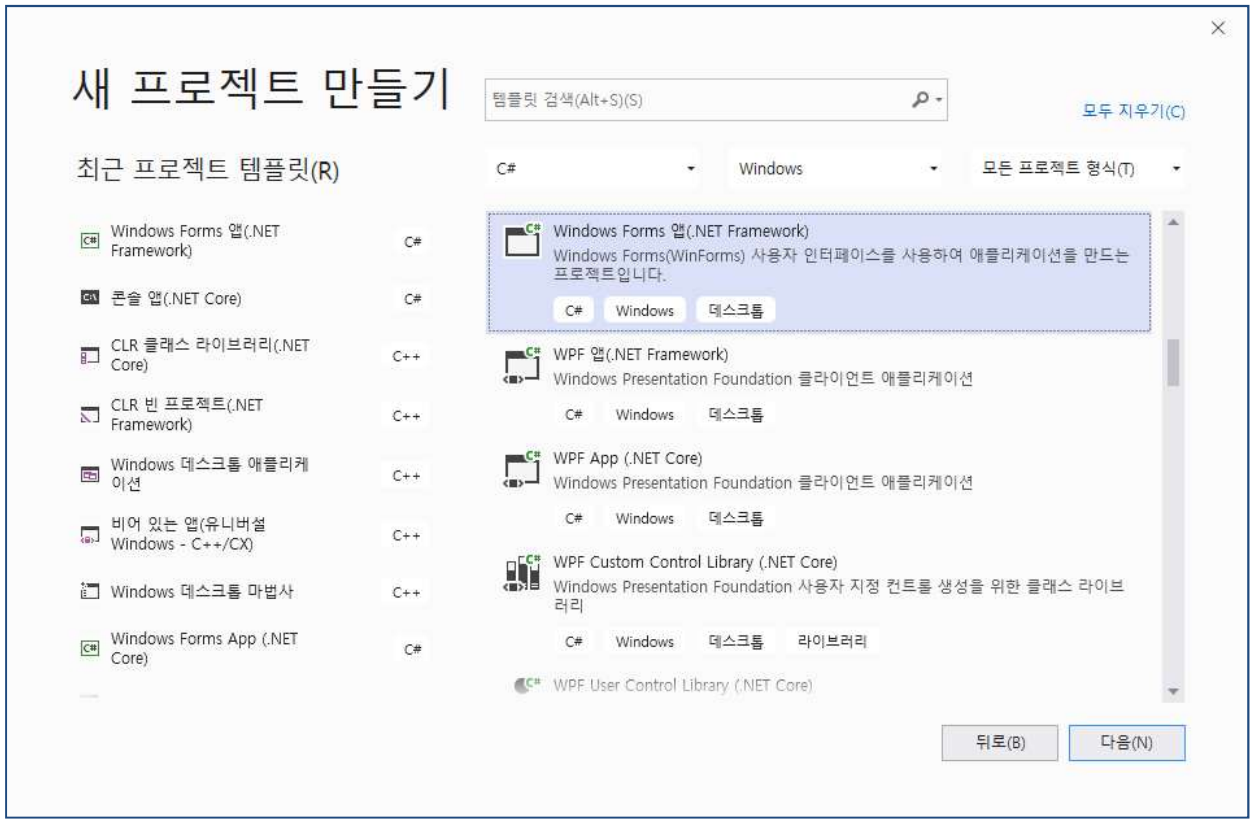
```
void loop( ) {
  switch (Motion){
    case 0:
      Motor_STOP( );          Speed=0;        break;
    case 1:
      Motor_CW(Speed);     break;
    case 2:
      Motor_CCW(Speed);   break;
    default:
      Motor_STOP(   );       Speed=0;         break;
  }

  Motor_Status( );
  delay(100);
}
```

```
void serialEvent( ){
  if (Serial.available( )){
    String Rxd =Serial.readStringUntil("/n");
    String cmd = Rxd.substring(1,Rxd.length( )-1);
    int comma= cmd.indexOf(",");
    Motion=cmd.substring(0,comma).toInt( );
    Speed=cmd.substring(comma+1).toInt( );

    Serial.println(Motion);
    Serial.println(Speed);
  }
}

void Motor_Status( ){
  Serial.print("@");
  Serial.print(Motion);
  Serial.print(",");
  Serial.println(Speed);
}
```

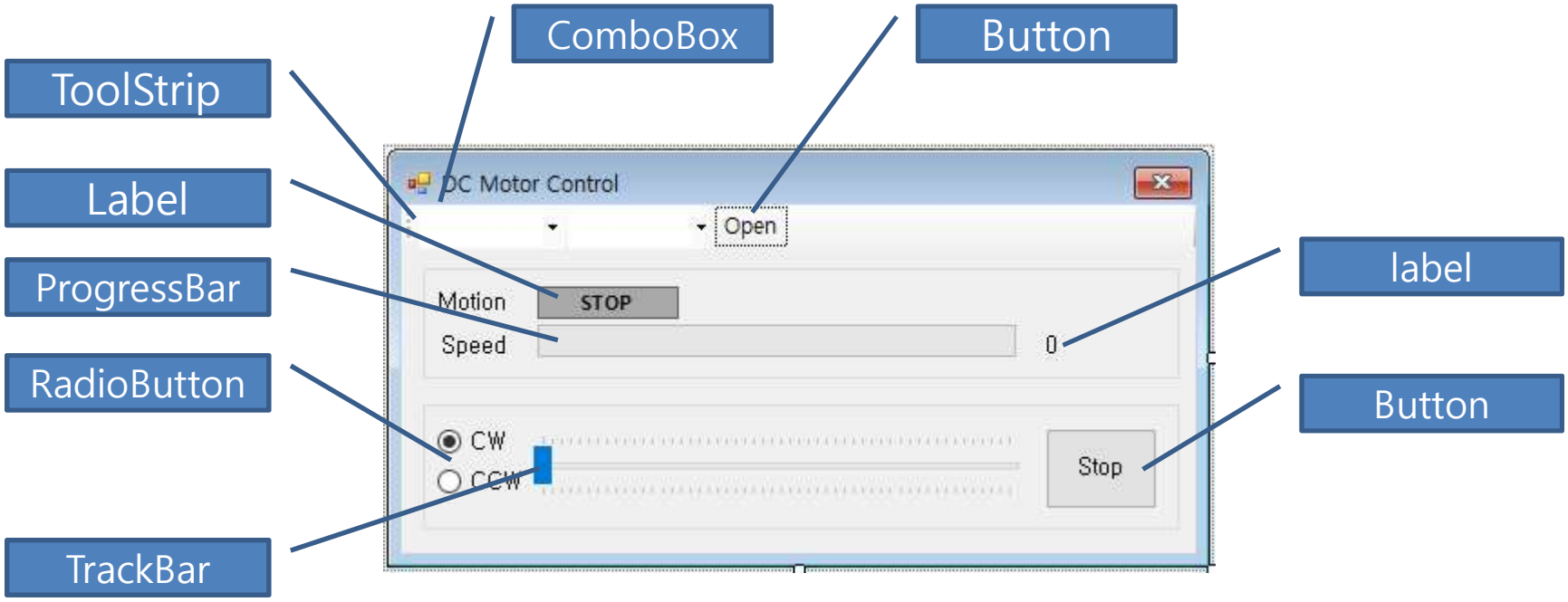
Run


COM6
전송
@1,120
@1,120
@1,120
@1,120
@1,120
@1,120
@1,120
@1,120
@1,120
@1,120
@1,120
@1,120
@1,120
@1,120
@1,120
자동 스크롤
타임스탬프 표시
새 줄
9600 보드레이트
출력 지우기

Visual Studio C#



Form Design


ToolStrip
ComboBox
Button
Label
ProgressBar
RadioButton
TrackBar
label
Button
DC Motor Control
Open
Motion
STOP
Speed
0
CW
CCW
Stop

Property

Form1	
Text	DC Motor Control
MaximizeBox	False
MinimizeBox	False
ToolStripComboBox1	
Name	cmbComPort
ToolStripComboBox2	
Name	cmbBoardRate
ToolStripButton1	
Name	btnConnect
Text	Open
Label1	
Name	lblMotion
Text	STOP

ProgressBar	
Maximum	255
Minimum	0
RadioNutton1	
Name	rdoCW
Checked	True
Text	CW
RadioNutton1	
Name	rdoCCW
Checked	False
Text	CCW
TrackBar1	
Maximum	255
Minimum	0

Define

```
using System;
using System.Drawing;
using System.Windows.Forms;
using System.IO.Ports;
```

```
SerialPort ComPort = new SerialPort();
private delegate void SetTextDelegate(string getString);

String Motion = "";
String SetMotion = "1";
int Speed = 0;
int SetSpeed = 0;
```

Serial Event / Delegate

```
public Form1( ) {
    InitializeComponent( );
    ComPort.DataReceived += new SerialDataReceivedEventHandler(DataReceived);
}

private void DataReceived(object sender, System.IO.Ports.SerialDataReceivedEventArgs e)  {
    string rxd = ComPort.ReadTo("₩n");
    this.BeginInvoke(new SetTextDelegate(SerialReceived), new object[ ] { rxd });
}
```

Serial Receive / Write

```
private void SerialReceived(string inString)  {
    try   {
        string Head = inString.Substring(0, 1);
        string Data = inString.Substring(1, inString.Length - 1);

        if (Head == "@")  {
            string[ ] PasingData = Data.Split(',');

            Motion = PasingData[0];
            Speed = Convert.ToInt16(PasingData[1]);
            Status(Motion, Speed);
         }
    } catch {
            //
    }
}
```

```
private void SerialWrite(string motion,int speed)  {
      if (ComPort.IsOpen)  {
          string msg = "@" + motion + "," + speed.ToString() + "₩n";
          ComPort.Write(msg);
      }
}
```

Status

```
private void Status(string motion, int speed)  {
    switch (motion)  {
        case "0":
            lblMotion.Text = "STOP";        lblMotion.BackColor = Color.DarkGray;          break;
        case "1":
            lblMotion.Text = "CW";          lblMotion.BackColor = Color.Orange;            break;
        case "2":
            lblMotion.Text = "CCW";         lblMotion.BackColor = Color.YellowGreen;       break;
        default:
            lblMotion.Text = "unknown";  lblMotion.BackColor = Color.Red;                  break;
    }

    progressBar1.Value = speed;
    lblSpeed.Text = speed.ToString();
}
```

Form Load / Closing

```
private void Form1_Load(object sender, EventArgs e)
{
      cmbComPort.Items.Clear( );
      var portName = System.IO.Ports.SerialPort.GetPortNames( );
      cmbComPort.Items.AddRange(portName);
      cmbComPort.SelectedIndex = cmbComPort.Items.Count - 1;

      cmbBoardRate.Items.Clear( );
      cmbBoardRate.Items.Add("9600");
      cmbBoardRate.Items.Add("19200");
      cmbBoardRate.Items.Add("57600");
      cmbBoardRate.Items.Add("115200");
      cmbBoardRate.SelectedIndex = 0;
}
```

```
private void Form1_FormClosing(object sender, FormClosingEventArgs e)  {
      if (ComPort.IsOpen)  {
            SerialWrite("0", 0);

            ComPort.Close();
       }

       ComPort.Dispose();
       ComPort = null;
 }
```

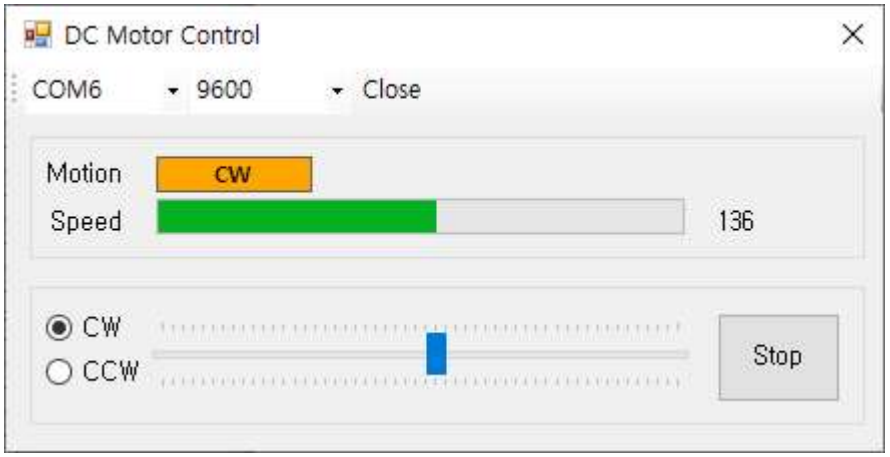
Serial Open

```
private void btnOpen_Click(object sender, EventArgs e)  {
    if (btnOpen.Text == "Open")  {
        ComPort.PortName = cmbComPort.Text;
        ComPort.BaudRate = Convert.ToInt32(cmbBoardRate.Text);
        ComPort.DataBits = 8;
        ComPort.Parity = Parity.None;
        ComPort.StopBits = StopBits.One;
        ComPort.Handshake = Handshake.None;
        ComPort.Open( );
        ComPort.DiscardInBuffer( );
        btnOpen.Text = "Close";
    }  else  {
        SerialWrite("0", 0);

        ComPort.Close( );
        btnOpen.Text = "Open";
    }
}
```

Object Event

```
private void btnStop_Click(object sender, EventArgs e)  {
      SerialWrite("0", 0);
}

private void rdoCW_CheckedChanged(object sender, EventArgs e)  {
      SetMotion = "1";
}

private void rdoCCW_CheckedChanged(object sender, EventArgs e)  {
      SetMotion = "2";
}

private void trackBar1_MouseUp(object sender, MouseEventArgs e)  {
      SetSpeed = trackBar1.Value;
      SerialWrite(SetMotion, SetSpeed);
}
```

Run


DC Motor Control
COM6
9600
Close
Motion
CW
Speed
136
CW
CCW
Stop