

Arduinoで作る  
ローテータ用リモコン  
製作メモ

2013年11月30日

難波 秀文

## 1. はじめに

ローテータをパソコンでコントロールする仕掛けを作りました。もともと、ローテータのコントローラ自体がリモコンな訳ですから、リモコンのリモコンということになります。

最近、無線機をパソコンに接続してリモートシャックから運用されている方もかなりいらっしゃるようです。となると、無線機だけではなくて、ローテータの制御も遠隔操作したいことでしょう。冬は寒いので、コタツトップパソコンで運用したいと思ったりすることもあります。簡易的なリモートシャックです。

大型のアンテナだと GD2 も大きいので、なるべく低速で回転させるようにしていますが、そうすると、目的の方位に達するまでに結構な時間を要します。その間、ずっとローテータのボタンを押して、動きを見守るなんて・・・とてもかったるいので Preset 動作をさせれば良いのですが、分解能が粗くて私としては我慢できません。

私は4本のタワーを建てていて、4台のローテータがあるのですが、ローテータを回す時に、椅子から立ち上がる必要があり、その時にヘッドフォンのコードが短すぎるために、いちいちヘッドフォンを外さねばならないなんていう不自由さのため、パソコンでローテータをコントロールしたいと考えたのです。

前置きが長くなりましたが、YAESU のローテータ G-2800SDX と G-1000DXA を対象とします。(G-2800SDX を 2 台、G-1000DXA を 2 台使っています。) ローテータ側にインテリジェントにするために、ArudinoDuemilanove を使いました。パソコン側のアプリは Visual Basic 2010 Express を用いて開発しました。

## 2. インタフェース

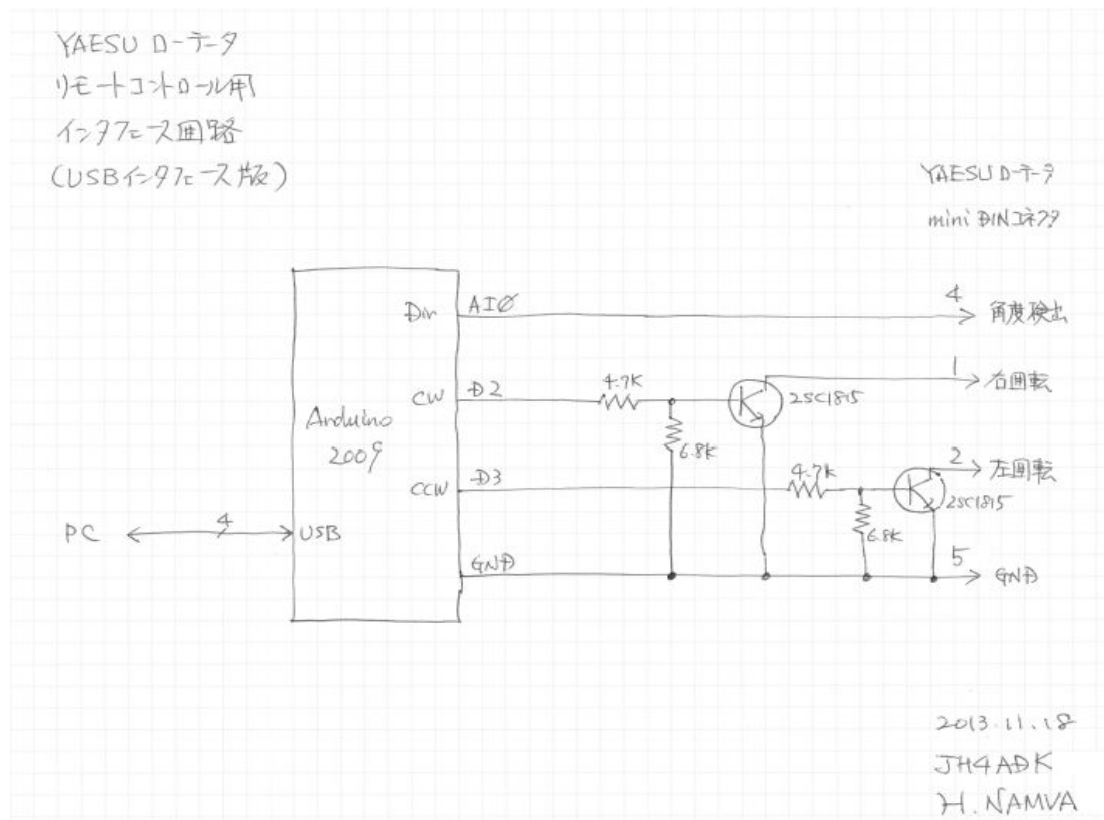
2種類のインタフェースを試しました。一つはパソコンから USB ケーブルで Arduino に接続して、Arduino からローテータコントローラに接続するという簡単なものです。部品点数も少なく製作も簡単なのですが、kW クラスの電力を扱うハイパワー局では高周波が回り込んで、ローテータが誤動作するという可能性があります。(実際私の場合、リニアアンプを使用すると回り込むという現象が起きました。)

二つ目のインタフェースは回り込みを防止するために、パソコンと Arduino の間をフォトカプラで絶縁する回路を設けたバージョンです。このために、余分に USB シリア

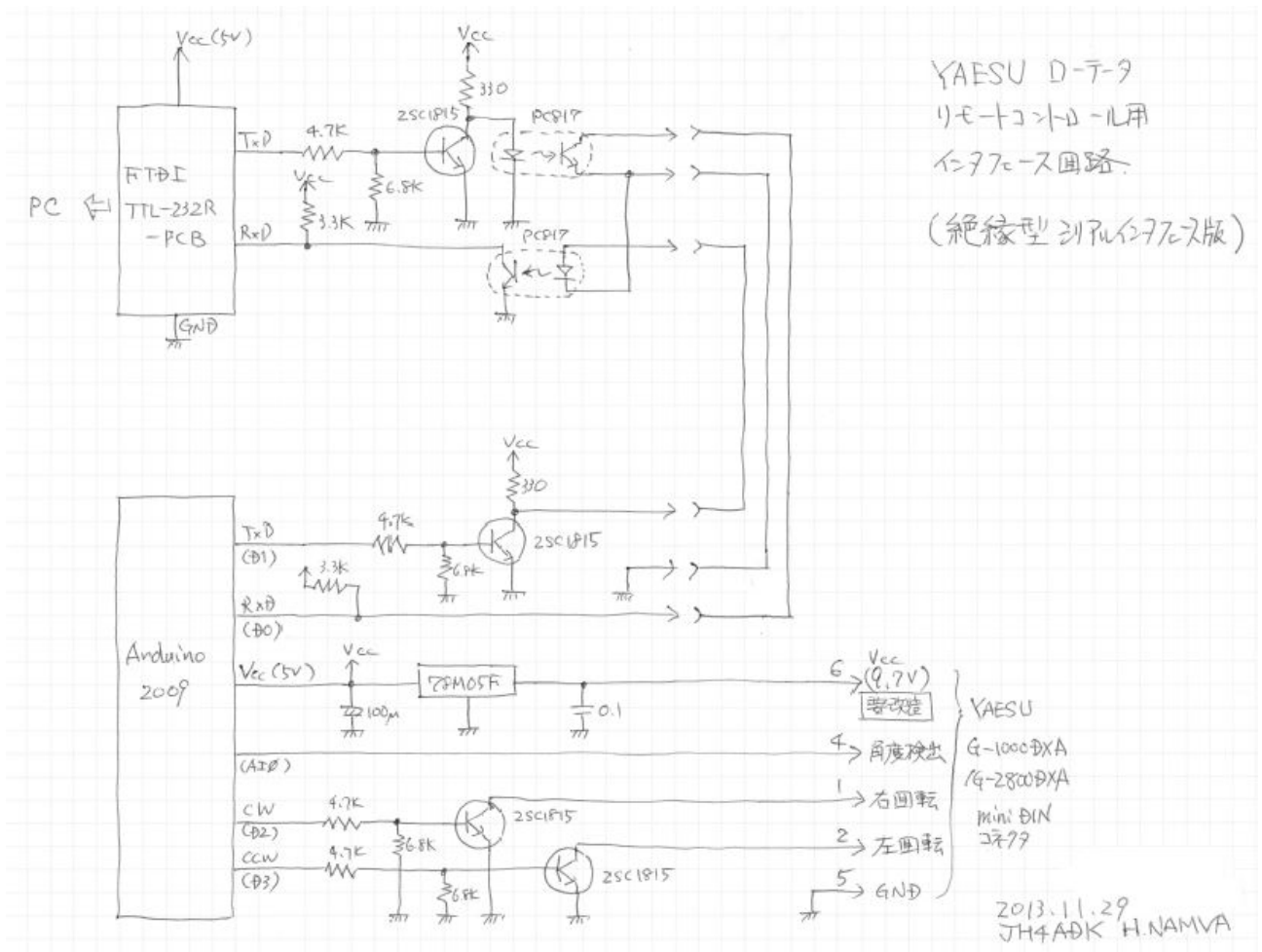
ル変換器を設け、Arduinoの電源をローテータコントローラから供給してもらうように、ローテータ側の一部を改造しました。(パターンを1箇所カットして、1本のジャンパ配線を追加するというものです。)

ハードウェア的には異なるものですが、Arduino側のファームウェアやパソコン側のアプリは全く同じものを使用します。

それぞれのインタフェース回路を以下に示します。



回路図1 USBインタフェース版



回路図 2 絶縁型シリアルインタフェース版

### 3 コマンドの定義

パソコンからはシリアルポートを通じてコマンドを送信します。コマンドを受信した Arduino は、コマンドに応じた動作をし、必要な場合にはレスポンスを返信します。コマンドは5種類です。簡単のためにA/B/C/D/Eコマンドとしました。

Aコマンドは'A'文字だけで、CW（右回転）のスイッチをONにします。

Bコマンドは'B'文字だけで、CCW（左回転）のスイッチをONにします。

Cコマンドは'C'文字だけで、CWおよびCCWのスイッチをOFFにします。

Dコマンドは'D'文字だけで、角度の問合せをします。

Arduino は角度（000～449）およびステータス（0、1または2）を数字4文字で返信します。ステータスが0の時はローテータが停止していることを、1の時はCW方向に回転中であることを、2の時はCCW方向に回転中であることを意味します。例えば、

CW 方向に回転中で角度が 273 の時は、2731 を返します。

E コマンドはプリセット値をロードしスタートさせます。

E コマンドは、数字 3 文字の角度と'E'の 4 文字で構成されます。

プリセット動作が開始され、ローテータが動作し始めると、方位とプリセット値の差が 5 度以内になると停止します。C コマンドを受信した時にも動作を停止します。

シリアル通信のパラメータは次の通りとします。

ボーレート : 1200BPS

8 ビットキャラクタ

2 ストップビット

ノーパリティ

#### 4 Arduino のファームウェア

Arduino について詳しい説明は割愛します。インターネットで検索すると色々と紹介されています。

要は、Arduino はマイコンボードで、フリーの開発環境やライブラリが提供されていて、とても簡単にファームウェアのプログラミングや書き込みができるものだということです。

ソースコードを以下に示します。

```
/*  
  Title:  RotCtrl  
  Description:  Rotator Controller for YAESU G-2800SDX and G-1000DXA  
  Author:  H.NAMVA JH4ADK  
  Original Issued on:  2013.11.18  
  Last Updated on:  2013.12.09 verion 1.01  
*/  
  
const int analogInPin = A0;  // Analog input pin that the potentiometer is  
attached to  
const int CW_OutPin = 2;  
const int CCW_OutPin = 3;  
char rxbuf[10];  
byte rxbuflen;  
int TargetDir;  
boolean Preset;  
byte MotorDirection;  
  
void setup() {  
  // initialize serial communications at 9600 bps:  
  Serial.begin(1200);  
  // initialize Digital Output pins  
  pinMode(CW_OutPin, OUTPUT);  
  pinMode(CCW_OutPin, OUTPUT);
```

```

    rxbuflen = 0;
    TargetDir = 0;
    Preset = 0;
    MotorDirection = 0;
}

void RotCW() {
    digitalWrite(CCW_OutPin, LOW);
    digitalWrite(CW_OutPin, HIGH);
    MotorDirection = 1;
}

void RotCCW() {
    digitalWrite(CCW_OutPin, HIGH);
    digitalWrite(CW_OutPin, LOW);
    MotorDirection = 2;
}

void RotSTOP() {
    digitalWrite(CCW_OutPin, LOW);
    digitalWrite(CW_OutPin, LOW);
    MotorDirection = 0;
}

void loop() {
    byte recvByte;
    int CurrentDir;
    int sensorValue;
    int delta;
    if(Serial.available() > 0){
        recvByte = Serial.read();
        switch(recvByte){
            case 'A': // CW start
                RotCW();
                rxbuflen = 0;
                break;
            case 'B': // CCW start
                RotCCW();
                rxbuflen = 0;
                break;
            case 'C': // stop rotation
                RotSTOP();
                Preset = 0;
                rxbuflen = 0;
                break;
            case 'D': // direction inquiry
                // read the analog in value:
                sensorValue = analogRead(analogInPin);
                // map it to the range of the analog out:
                CurrentDir = map(sensorValue, 0, 1023, 0, 450);
                Serial.print(CurrentDir);
                Serial.println(MotorDirection);
                rxbuflen = 0;
                break;
            case 'E': // set direction and rotate automatically
                // stop rotation
                RotSTOP();
                rxbuf[rxbuflen+1] = 0; // append nul char
                TargetDir = atoi(rxbuf);
                rxbuflen = 0;
                sensorValue = analogRead(analogInPin);

```

```

CurrentDir = map(sensorValue, 0, 1023, 0, 450);
delta = abs(CurrentDir - TargetDir);
if (delta >= 360) { delta -= 360;}
if (delta > 5){
  if (TargetDir <= 75){
    if (CurrentDir >= 360) {
      if (TargetDir > (CurrentDir - 360)) {
        RotCW();
        Preset = 1;
      } else{
        RotCCW();
        Preset = 1;
      }
    }else if (CurrentDir > 270) {
      RotCW();
      Preset = 1;
    }else if (CurrentDir < TargetDir) {
      RotCW();
      Preset = 1;
    }else{
      RotCCW();
      Preset = 1;
    }
  }else if (TargetDir < 360) {
    if (CurrentDir > 360) {
      RotCCW();
      Preset = 1;
    }else{
      if (CurrentDir < TargetDir){
        RotCW();
        Preset = 1;
      }else{
        RotCCW();
        Preset = 1;
      }
    }
  }
}
break;
default:
  if (recvByte >= '0' && recvByte <= '9'){
    rxbuf[rxbuflen++] = recvByte;
    if (rxbuflen > 4) {
      rxbuflen = 0;
    }
  }
}
}else if (Preset){
  sensorValue = analogRead(analogInPin);
  CurrentDir = map(sensorValue, 0, 1023, 0, 450);
  delta = abs(CurrentDir - TargetDir);
  if (delta >= 360) { delta -= 360;}
  if (delta < 5) {
    RotSTOP();
    Preset = 0;
  }else if ((CurrentDir > 450) || (CurrentDir < 5)){
    RotSTOP();
    Preset = 0;
  }
}
}
}

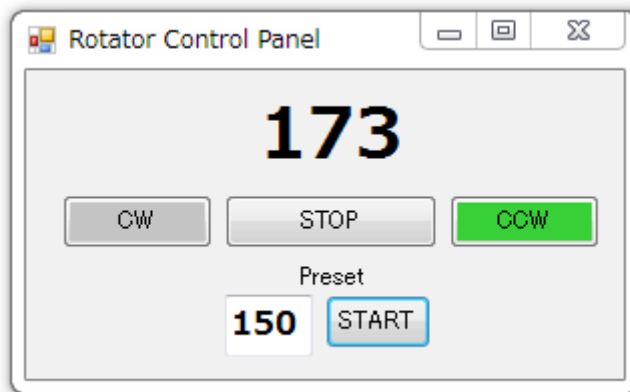
```

## 5 パソコン側アプリ

パソコン側のアプリは Visual Basic 2010 Express で作成しました。ZIP 形式でプロジェクトファイルごと配布しています。

COM ポート番号は COM8 にプログラムで固定していますので、使用されるパソコンの COM ポートに合わせて変更して、ビルドしてください。

アプリの画面は次のようなものです。



上段の少し大きめの数字（173 と表示）は方位角を示します。

CW ボタンを押せば時計回り（右回り）に回転し、CCW ボタンを押せば反時計回り（左回り）に回転します。

Preset の角度を数値（0～359）で設定して START ボタンを押せば、ローテータが指定された方位に向かって回転し始めます。Preset の角度と方位角との誤差が±5度以下になると停止します。もしも、Preset の角度と現在の方位角との誤差が±5度以下であれば、START ボタンを押しても回転しません。

STOP を押すと、回転を停止します。

回転中は、CW または CCW ボタンの色がグリーンになります。停止すると、これらの色がシルバーになります。

パソコン側アプリのソースコードを以下に示します。

```
'Title: Rotator Control Panel
'Author: H.NAMVA
'Original issued on: F 2013.11.18 ver 1.00
'Last updated on: 2013.12.09 ver 1.01
Public Class Form1
    Private RxBuf(1024) As Byte
    Private RxBufLength As Integer
```



```

Private strAzimuth As String = ""

Private Sub Form1_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
    Dim availableSerialPorts As
System.Collections.ObjectModel.ReadOnlyCollection(Of String) =
My.Computer.Ports.SerialPortNames
    Try
        If availableSerialPorts.Contains(COMPort.PortName) Then
            If COMPort.IsOpen Then
                MessageBox.Show(COMPort.PortName & " is already OPENED",
"error", _
                    MessageBoxButtons.OK, MessageBoxIcon.Error)
                Me.Close()
            Else
                COMPort.Open()
            End If
        Else
            MsgBox(COMPort.PortName & " is not available")
            Me.Close()
        End If
    Catch ex As Exception
        Debug.Print(ex.Message & " in Form1_Load")
        Me.Close()
    End Try
End Sub
Delegate Sub ModifyAzimuthLabel()

Sub ModifyAzimuth()
    Dim temp As String = ""
    Dim direction As Integer
    Dim overlap As Integer
    Dim motor As Integer

    Try
        direction = Integer.Parse(strAzimuth)
        motor = direction Mod 10
        direction = direction \ 10
        lblAzimuth.Text = direction.ToString
        If direction >= 360 Then
            overlap = direction - 360
            lblAz2.Text = "(" & overlap.ToString & ")"
            lblAz2.Visible = True
        Else
            lblAz2.Visible = False
        End If
        If motor = 1 Then 'CW
            btnCW.BackColor = Color.LimeGreen
        ElseIf motor = 2 Then 'CCW
            btnCCW.BackColor = Color.LimeGreen
        Else
            btnCW.BackColor = Color.Silver
            btnCCW.BackColor = Color.Silver
        End If

        TxbRx.AppendText(strAzimuth & vbCrLf)
        If Len(TxbRx.Text) > (TxbRx.MaxLength - 1000) Then
            'サライズYが大きく-なEりe過さるた?らc、Aデフ-[タ^をd半?分a消A去?すEるe
            temp = Mid(TxbRx.Text, TxbRx.MaxLength / 2, Len(TxbRx.Text))
            TxbRx.Text = temp
        End If
    Catch ex As Exception
        Debug.Print(ex.Message & "in ModifyAzimuth")
    End Try
End Sub

```

```

End Sub

Private Sub COMPort_DataReceived(ByVal sender As System.Object, ByVal e As
System.IO.Ports.SerialDataReceivedEventArgs) Handles COMPort.DataReceived
    Dim ModAzi As New ModifyAzimuthLabel(AddressOf ModifyAzimuth)
    Try
        strAzimuth = COMPort.ReadLine()
        Debug.Print("data received: " & strAzimuth)
        lblAzimuth.Invoke(ModAzi)
    Catch ex As Exception
        Debug.Print(ex.Message & vbCrLf & "RxBufLength = " &
RxBufLength.ToString)
    End Try
End Sub

Private Sub Timer1_Tick(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Timer1.Tick
    COMPort.Write("D")
End Sub

Private Sub btnCW_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnCW.Click
    COMPort.Write("A")
End Sub

Private Sub btnCCW_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnCCW.Click
    COMPort.Write("B")
End Sub

Private Sub btnSTOP_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnSTOP.Click
    COMPort.Write("C")
End Sub

Private Sub btnStart_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnStart.Click
    Dim preset As Integer
    Try
        preset = Integer.Parse(TxbPreset.Text)
        COMPort.Write(preset.ToString("D3") & "E")
    Catch ex As Exception
        MsgBox("Invalid Preset Value")
        Debug.Print(ex.Message & "in btnPrest")
    End Try
End Sub
End Class

```

#### 4 調整

パソコンに表示される方位角とローテータコントローラに表示される方位角が一致するように調整する必要があります。

次の手順で調整します。

- 1)ローテータコントローラを操作して、アンテナの方位を 360 度（ゼロではなくて、丁度一回転したところ）に合わせます。
- 2)ローテータコントローラの後ろにあるトリマ（OUT VOL. ADJ）を回して、パソコン側アプリの表示が 360 になるようにします。

調整はこれで OK です。