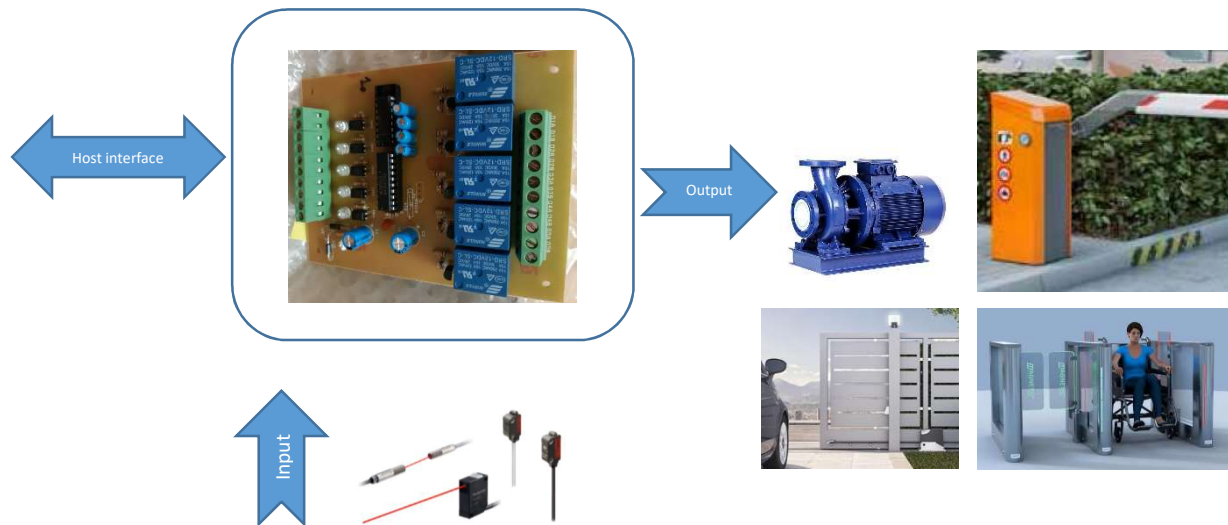


BO MẠCH GIAO TIẾP ĐIỀU KHIỂN TÍN HIỆU DIGITAL

PTC.RS232 10DIO



A/ GIỚI THIỆU CHUNG

Đây là bo mạch thực hiện giám sát và điều khiển thiết bị giao tiếp với máy tính qua ngõ giao tiếp RS232 (cổng COM – Serial port).

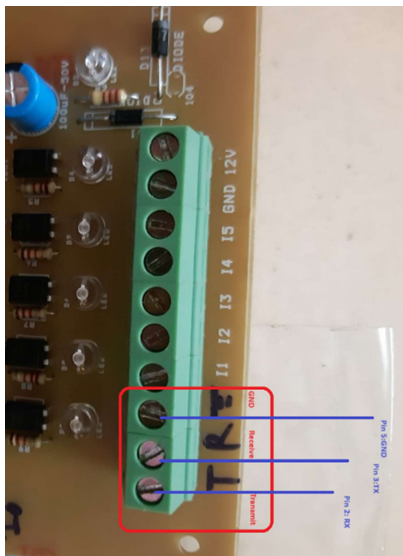
Thông số kỹ thuật:

- Mã SP: PTC.RS232 10DIO
- Nguồn: 12 VDC, 2A
- **Địa chỉ bo mạch: Chỉ có ở phiên bản PTC.RS232 10DIO ADD**
- 05 digital input (Kích GND)
- 05 digital output (5 relays of dry contact 10A)
- Protocol: RS 232 (3 wires: TX, RX, GND). Sample viết bằng Visual Basic 6.0
- Môi trường làm việc: 0^oC – 80^oC, không lắp đặt trong nước.
- Ứng dụng:

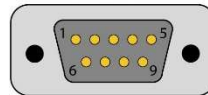
- Điều khiển đóng/ ngắt các thiết bị điện có công suất tối đa 10A (Nếu cao hơn thì lắp bổ sung relay)
- Giao tiếp với các loại cảm biến có ngõ output là DRY CONTACT (Nếu là tín hiệu điện thì phải lắp bổ sung relay)

B/ ĐẦU NỐI THIẾT BỊ

Bộ mạch được đấu nối với nguồn 12 VDC (Chân GND và 12V)



DB9M Connector

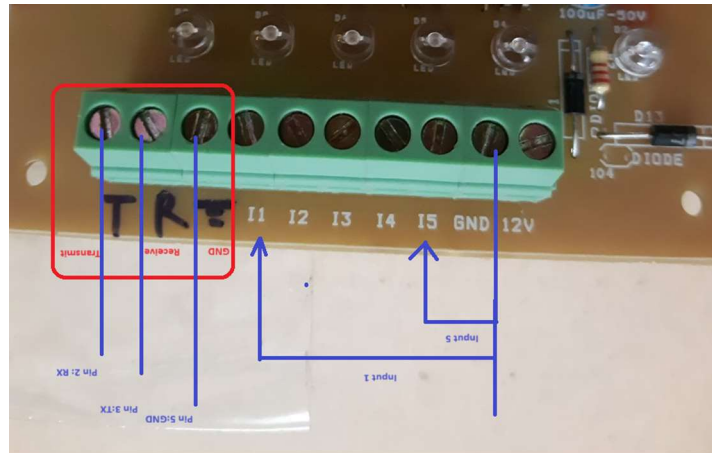


RS232 Pin Out

Pin #	Signal
1	DCD
2	RX
3	TX
4	DTR
5	GND
6	DSR
7	RTS
8	CTS
9	RI

Bộ mạch được đấu nối với máy tính (hoặc các thiết bị điều khiển giao tiếp RS 232): Ngõ ký hiệu chữ T (Transmit) trên bộ mạch được đấu vào chân 2 (pin 2) của cổng COM DB-9 trên máy tính; Ngõ ký hiệu R (Receive) trên bộ mạch được đấu vào chân 3 (pin 3) của cổng COM DB-9 trên máy tính; còn lại là GND đấu vào chân số 5 (pin 5)

Đấu nối Input: Tùy theo thiết kế của hệ thống của bạn, nếu bạn dùng cảm biến hoặc nút nhấn thì tín hiệu kích ngõ Input phải là tín hiệu khô (DRY CONTACT). Để kích ngõ Input 1 (I1) thì nối chân GND và chân I1 trên bộ mạch, tương tự cho các Input 2 đến Input 5. Mỗi chân Input đều có 01 đèn LED màu trắng tương ứng nằm ngay vị trí của chân Input đó.



Đầu nối Output:

Mỗi ngõ Output gồm 2 chân, là các cặp tiếp điểm khô, độc lập, các bạn có thể đấu với nhiều loại nguồn điện khác nhau.



Output1 gồm 01A và 01B, khi Output 1 ON: 01A và 01B được thông nhau

Khi Output 1 OFF: 01A và 01B ngắt kết nối

Tương tự, các ngõ Output2, 3, 4, 5 cũng như Output 1.



C/ NGHI THỨC GIAO TIẾP (PROTOCOL)

I/ ĐIỀU KHIỂN 5 OUTPUT

Để điều khiển 05 relay output của board điều khiển cần xuất chuỗi lệnh xuất ra cổng COM có cấu trúc như sau:

<STX>0255AA`YY`<ETX>

Trong đó: <STX> là ký tự có mã ASCII là 2, <ETX> có mã ASCII là 3. Nếu bạn lập trình Visual Basic thì <STX> chính là Chr(2), <ETX> chính là Chr(3).

I.1 – Lệnh điều khiển output

0255AA LÀM COMMAND

I.2 – Điều khiển output

YY LÀ BYTE SỐ HEX, có giá trị từ Hex("00") đến Hex("1F"), THỂ HIỆN TRẠNG THÁI MỞ/ ĐÓNG RELAY. Trong đó: Hex("00") = Binary("00000000"), Hex("1F") = Binary("00011111")

08 BIT trong Byte xuất ra ĐƯỢC MÔ TẢ Ở DẠNG NHỊ PHÂN LÀ:
[.][.][.][D5][D4][D3][D2][D1]. Trong đó [.] là không dùng đến nên có giá trị 0 hay 1 gì cũng được.
(Tức là: BIT 0 ứng với trạng thái Out put cho cổng D1, BIT 1 ... cho cổng D2, BIT 2 ... cho cổng D3, BIT 3 ... cho cổng D4, BIT 4 ... cho cổng D5).

Đối với BIT0: điều khiển D1

BIT0 =1 RELAY D1 ON

BIT0 = 0 RELAY D1 OFF

Như vậy, nếu chỉ muốn ON mỗi Relay D1, còn các Relay còn lại OFF thì BYTE gửi đi có giá trị mô tả ở dạng nhị phân là: "00000001", tức 0x01 ở hệ Hexa

Chuỗi phải gửi ra cổng COM là: Chr(2) & "0255AA01" & Chr(3)

(Dấu "", & ở trên là dấu công chuỗi thôi nha các bạn, không gửi)



TƯƠNG TƯ CHO BIT 1 - BIT 4. ĐỂ ĐIỀU KHIỂN CÙNG LÚC NHIỀU CHÂN THỰC HIỆN NHƯ SAU:

BYTE gửi đi phải chứa tất cả các trạng thái D1 đến D5.

Ví dụ: cần điều khiển D1 = ON, D2 = OFF, D3 = OFF, D4 = ON, D5 = ON thì Byte gửi đi được mô tả ở dạng nhị phân là: “00011001”, ở dạng Hexa là “0x19”, ở dạng thập phân là “25”

(Các bạn không tin thì vào link này để convert nhé:

<https://www.rapidtables.com/convert/number/binary-to-hex.html>)

Khi đó, chuỗi phải gửi ra cổng COM là: STX & “0255AA19” & ETX

Nếu bạn sử dụng VB 6.0 thì có thể CODE như sau:

Click vào link <https://www.vbforums.com/showthread.php?156035-convert-binary-to-Hex> để copy hàm BinaryToDecimal vào Project của mình (Hoặc xem phần Phụ lục bên dưới). Rồi viết hàm Send_to_Com như sau:

```
Public Sub Send_To_Com ()
```

```
...
```

```
Dim ss as String
```

```
Dim strHex as String
```

```
Dim sBinary as String
```

```
'Sample: D1 = ON, D2 = OFF, D3 = OFF, D4 = ON, D5 = ON
```

```
'Khi nhúng vào phần mềm thì bạn Build cho mình sBinary theo yêu cầu điều khiển
```

```
sBinary = “00010011”
```

```
strHex = Hex(BinaryToDecimal(sBinary))
```

```
ss = chr(2) & “0255AA” & Right("00" & strHex, 2) & chr(3)
```

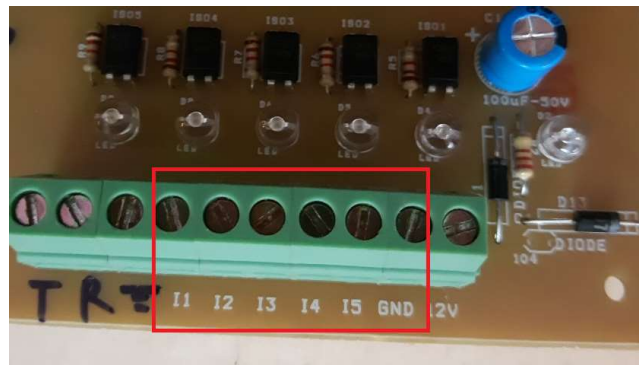
```
'Gửi chuỗi ra cổng COM. Bạn nhớ add OCX MSComm vào From của mình và cấu hình đúng COM PORT.
```

```
MSComm.Output = ss
```

```
...
```

```
End Sub
```

II/ THU THẬP 05 NGÕ DIGITAL INPUT



II.1/ GÓI DỮ LIỆU

Khi có sự thay đổi trạng thái các ngõ vào từ I1 đến I5, gói dữ liệu được gửi về máy tính (Host) qua chân T (Transmit) có cấu trúc như sau:

<STX>0155AANN<ETX>

Trong đó: <STX> và <ETX> đã được trình bày ở phần C.I ở trên.

II.1.1 – Lệnh thu thập input

0155AA LÀM COMMAND

II.1.2 – Thu thập trạng thái Input

NN: Là Byte trạng thái 05 cổng Input, mô tả dạng nhị phân là: [..][..][..][I5][I4][I3][I2][I1] LÀ BÍT TƯƠNG ỨNG CÁC NGÕ VÀO.

0155AA LÀ COMMAND

Để biết trạng thái của I1 đến I5 đơn giản chỉ convert NN sang dãy Binary sẽ biết được trạng thái của từng Input: **1 là OFF (không kích), 0 là ON (Kích GND).**

Ví dụ: NN = “19” (tức là 25 hệ thập phân), khi convert sang Binary sẽ là “00010011”. Điều này có nghĩa là I1 OFF, I2 OFF, I3 ON, I4 ON, I5 OFF

Lập trình: Nếu lập trình Visual Basic 6.0 có thể dùng hàm Hex2Bin để convert số Hex NN sang chuỗi Binary. Xem hàm này ở phần phụ lục hoặc tại website:
<https://www.vbforums.com/showthread.php?93709-Convert-Hex-to-Bin>

Khi lập trình giao tiếp với cổng COM (Serial), bạn cần kỹ năng xử lý nhận dữ liệu qua sự kiện Oncom – bởi bo mạch gửi 1 chuỗi qua nhiều sự kiện Oncom, bắt đầu là STX, các gói dữ liệu, kết thúc bằng ETX. Bạn có thể tham khảo Sample viết bằng Visual Basic 6.0



PHỤ LỤC

CODE MẪU VB6

Public Function BinaryToDecimal(BinaryString As String) As Long

```
If (Not Len(BinaryString) > 1023) Then
    Dim retVal As Double
    If (BinaryString <> "") And (Replace(Replace(BinaryString, "1", ""), "0", "") = "") Then
        Dim i As Long
        BinaryString = StrReverse(BinaryString)
        For i = 0 To Len(BinaryString)
            If (Mid(BinaryString, i + 1, 1) = 1) Then
                retVal = retVal + (2 ^ (i))
            End If
        Next i
    End If
    BinaryToDecimal = retVal
Else
    MsgBox "Overflow would occur"
    Exit Function
End If
```

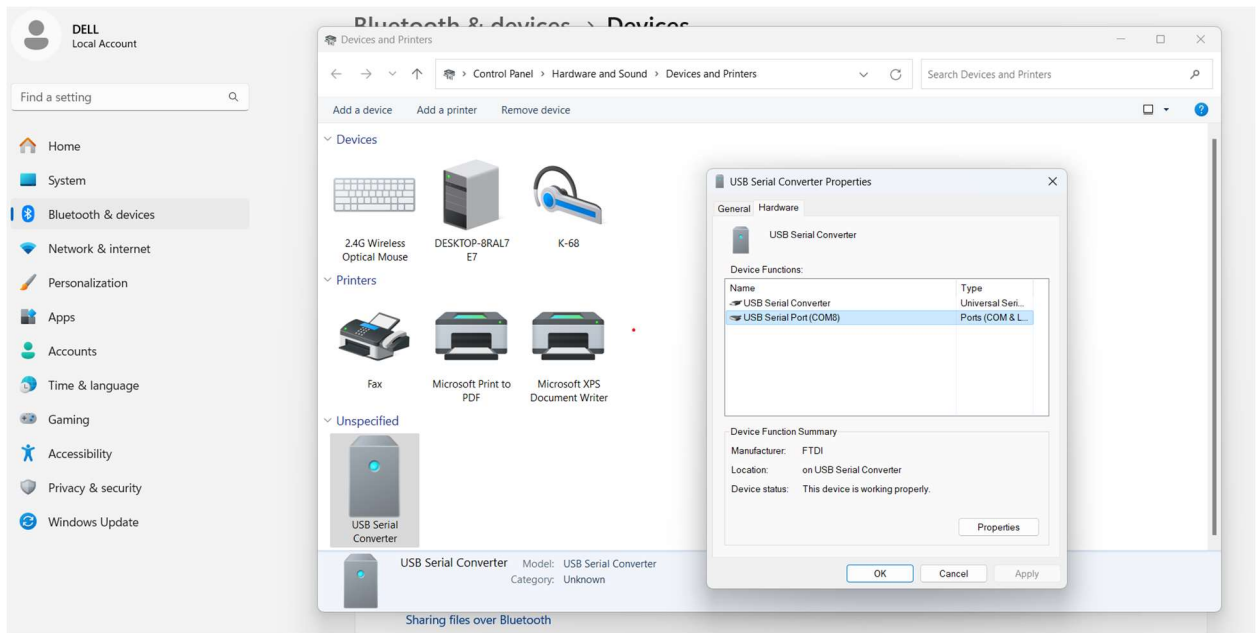
End Function**Public Function Hex2Bin(ByVal dwHex As String) As String**

```
dwHex = CStr(dwHex)
Dim sTemp As String
For l = 1 To Len(dwHex)
    sTemp = Mid(dwHex, l, 1)
    Select Case sTemp
        Case "0": Hex2Bin = Hex2Bin & "0000"
        Case "1": Hex2Bin = Hex2Bin & "0001"
        Case "2": Hex2Bin = Hex2Bin & "0010"
        Case "3": Hex2Bin = Hex2Bin & "0011"
        Case "4": Hex2Bin = Hex2Bin & "0100"
        Case "5": Hex2Bin = Hex2Bin & "0101"
        Case "6": Hex2Bin = Hex2Bin & "0110"
        Case "7": Hex2Bin = Hex2Bin & "0111"
        Case "8": Hex2Bin = Hex2Bin & "1000"
        Case "9": Hex2Bin = Hex2Bin & "1001"
        Case "A": Hex2Bin = Hex2Bin & "1010"
        Case "B": Hex2Bin = Hex2Bin & "1011"
        Case "C": Hex2Bin = Hex2Bin & "1100"
        Case "D": Hex2Bin = Hex2Bin & "1101"
        Case "E": Hex2Bin = Hex2Bin & "1110"
        Case "F": Hex2Bin = Hex2Bin & "1111"
    End Select
Next l
```

End Function

Sử dụng bộ chuyển đổi UBS to RS232

Hiện nay phần lớn máy tính thương mại đã khai tử cổng RS232, trừ các dòng máy tính công nghiệp. Do vậy các bạn cần phải mua bộ chuyển đổi USB to RS232 để có cổng giao tiếp với bo mạch PTC.RS232 10DIO. Khi lắp đặt, đối với Windows 11, bạn vào Control Panel/ Hardware and Sound/ Devices and Printers để kiểm tra cổng COM là bao nhiêu nhé (xem hình bên dưới)



MỌI THẮC MẮC BẠN VUI LÒNG LIÊN HỆ ĐẠI LÝ CUNG CẤP HÀNG
XIN CHÂN THÀNH CẢM ƠN