

VB25 - TCP/IP network communications with a Picaxe 28x2

DISCLAIMER

HEREIN "THE CONTENT" SHALL REFERE TO ANY AND ALL VIDEOS, CONTENT, SOFTWARE, CODE, EXAMPLES, CIRCUITS, DOWNLOADS, KNOWHOW, AND/OR INFORMATION.

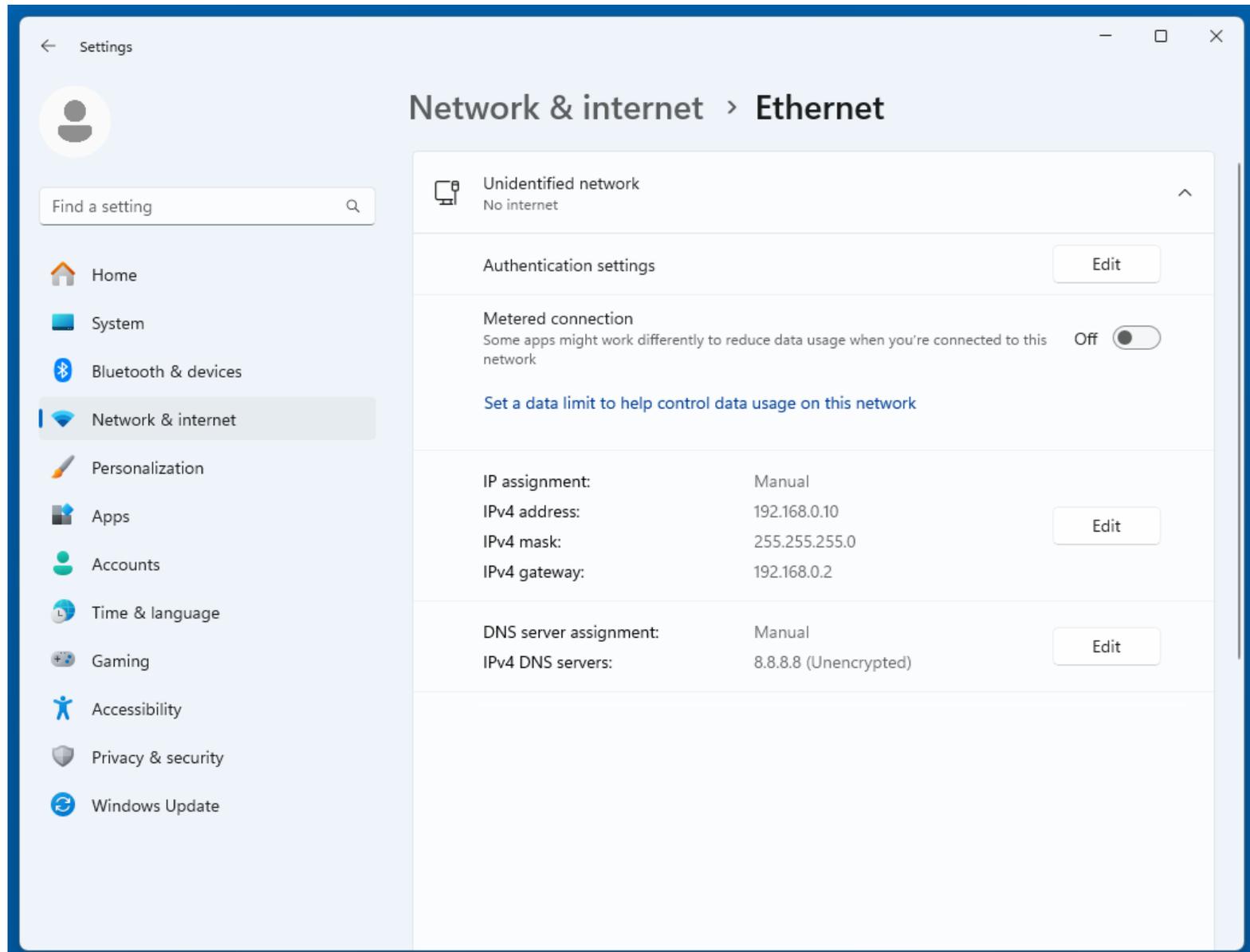
THE CONTENT PRESENTED BY DR. DAN WEDDING IS FOR ENTERTAINMENT PURPOSES ONLY.

DR. DAN WEDDING MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND, EXPRESS OR IMPLIED, AS TO THE ACCURACY OF THE CONTENT AND THE CONTENT IS PROVIDED "AS-IS".

IN EXCHANGE FOR YOUR USE OF THE CONTENT; YOU AGREE INDEMNIFY AND HOLD DR. DAN WEDDING HARMLESS FROM AND AGAINST ANY AND ALL CLAIMS, DAMAGES, COSTS, AND EXPENSES, INCLUDING ATTORNEYS' FEES, ARISING FROM AND/OR RELATED TO THE USE OF THE CONTENT.

REPRODUCTION, REPUBLICATION, DISTRIBUTION, TRANSMISSION, DISPLAY, OR PERFORMANCE, OF THE CONTENT IS PROHIBITED WITHOUT THE EXPRESS WRITTEN CONSENT OF DR. DAN WEDDING.

SETTINGS (COMPUTER) - Youtube.com/@DrWedding VB25



The image shows a Windows Settings window titled "Settings" with a back arrow on the left and window control buttons on the right. The main heading is "Network & internet > Ethernet". On the left is a navigation pane with a search box "Find a setting" and a list of categories: Home, System, Bluetooth & devices, Network & internet (highlighted), Personalization, Apps, Accounts, Time & language, Gaming, Accessibility, Privacy & security, and Windows Update. The main content area shows an "Unidentified network" with "No internet" and an "Edit" button. Below this are "Authentication settings" and "Metered connection" (set to "Off"). A link "Set a data limit to help control data usage on this network" is present. The IP configuration section shows "IP assignment: Manual" with "Edit" button, and "IPv4 address: 192.168.0.10", "IPv4 mask: 255.255.255.0", and "IPv4 gateway: 192.168.0.2". The DNS section shows "DNS server assignment: Manual" with "Edit" button, and "IPv4 DNS servers: 8.8.8.8 (Unencrypted)".

Settings

Network & internet > Ethernet

Unidentified network
No internet

Authentication settings Edit

Metered connection
Some apps might work differently to reduce data usage when you're connected to this network Off

[Set a data limit to help control data usage on this network](#)

IP assignment:	Manual	
IPv4 address:	192.168.0.10	Edit
IPv4 mask:	255.255.255.0	
IPv4 gateway:	192.168.0.2	

DNS server assignment:	Manual	
IPv4 DNS servers:	8.8.8.8 (Unencrypted)	Edit

SETTINGS (USR-TCP232) - Youtube.com/@DrWedding VB25

The image displays two side-by-side screenshots of the USR-TCP232-T24 V5.1.1.20 software interface. Both windows show the same settings, but with different Module IP addresses.

Left Window (Module IP: 192.168.0.7):

- Module work mode: TCP Server
- Module IP: 192.168.0.7
- Subnet mask: 255.255.255.0
- Default Gateway: 192.168.0.1
- Baud Rate(bps): 9600
- Parity/Data/Stop: NONE | 8 | 1
- Module port: 57878 Random
- Destination IP: 192.168.0.10
- Destination Port: 51010

Operate via COM (?): CFG connect to GND

- Select serial port: No serial port (?)
- Buttons: Read via COM, Setup via COM

Operate via LAN (?): Leave CFG pin free

- Buttons: Search in LAN, Set selected item via LAN

Device list in the Net

Module IP	MAC	Ver
192.168.0.7	9CA52584244C	11.1
192.168.0.8	9CA5258B4C8B	11.1

Logs

Parameters has updated to left form.After change Param,click [Set selected item via LAN].

Right Window (Module IP: 192.168.0.8):

- Module work mode: TCP Server
- Module IP: 192.168.0.8
- Subnet mask: 255.255.255.0
- Default Gateway: 192.168.0.2
- Baud Rate(bps): 9600
- Parity/Data/Stop: NONE | 8 | 1
- Module port: 57878 Random
- Destination IP: 192.168.0.10
- Destination Port: 51010

Operate via COM (?): CFG connect to GND

- Select serial port: No serial port (?)
- Buttons: Read via COM, Setup via COM

Operate via LAN (?): Leave CFG pin free

- Buttons: Search in LAN, Set selected item via LAN

Device list in the Net

Module IP	MAC	Ver
192.168.0.7	9CA52584244C	11.1
192.168.0.8	9CA5258B4C8B	11.1

Logs

Parameters has updated to left form.After change Param,click [Set selected item via LAN].

VB FORM - Youtube.com/@DrWedding VB25



VB CODE - Youtube.com/@DrWedding VB25

'Written by: Dan Wedding Youtube.com/@DrWedding

'Date: 2-23-2233

'Description:

'This code sets up TCP/IP communication with a socket between the uControllers and the computer.
'The uControllers and Computer must be on the same subnet.

' The computer is a client
' The computer's IP address is 192.168.0.10
' The computer will accept any port (in case the uControllers are transmitting on different ports)
' The computer is transmitting on 57878

' The uController007 's IP address is 192.168.0.7
' The uController007 is set up as a TCP SERVER
' The uController007 is monitoring port number 57878 for incoming data
' Inside the uController007 it knows to transmit to the TCP CLIENT at IP address 192.168.0.10
' Inside the uController007 it knows to transmit on 51010 but it doesn't matter. The computer will look at
' all of the incoming ports from the uController.

' The uController008 's IP address is 192.168.0.8
' The uController008 is set up as a TCP SERVER
' The uController008 is monitoring port number 57878 for incoming data
' Inside the uController008 it knows to transmit to the TCP CLIENT at IP address 192.168.0.10
' Inside the uController008 it knows to transmit on 51010 but it doesn't matter. The computer will look at
' all of the incoming ports from the uController.

Imports System

Imports System.Net

Imports System.Net.Sockets

Imports System.Text

Imports System.Threading

Public Class Form1

' GLOBAL VARIABLES

'If you need the computer's IP address, this is the command. But it is not used in this code.

'Public ComputerIPAddress As IPHostEntry = Dns.GetHostEntry(Dns.GetHostName())

' ***** uController007 information:

'Load the IP address of the uController into uControllerIPAddress

Dim uController007IPAddress As IPAddress = IPAddress.Parse("192.168.0.7")

'Combine the IP Address of the uController and the Port Number into one IPEndPoint variable

Dim uController007IPandPort As New IPEndPoint(uController007IPAddress, 57878)

'Open a Socket to the data stream

Dim uController007Socket As New Socket(uController007IPAddress.AddressFamily, SocketType.Stream, ProtocolType.Tcp)

' ***** uController008 information:

'Load the IP address of the second uController into uControllerIPAddress

Dim uController008IPAddress As IPAddress = IPAddress.Parse("192.168.0.8")

'Combine the IP Address of the uController and the Port Number into one IPEndPoint variable

Dim uController008IPandPort As New IPEndPoint(uController008IPAddress, 57878)

'Open a Socket to the data stream

Dim uController008Socket As New Socket(uController008IPAddress.AddressFamily, SocketType.Stream, ProtocolType.Tcp)

' ***** Input and output storage

'Input Data Buffer

Dim InMsg(32) As Byte

'Output Data Buffer

Dim OutMsg(32) As Byte

' Used to record how many bytes were transmitted

Dim bytesSent As Integer

' Used to record how many bytes were received from ucontrollers

Dim bytesReceived As Integer

Private Sub Form1_Load(sender As Object, e As EventArgs) Handles MyBase.Load

Try

'Establish a TCP/IP Socket connection to uController007 (IP and Port)
uController007Socket.Connect(uController007IPandPort)

'Establish a TCP/IP Socket connection to uController008 (IP and Port)
uController008Socket.Connect(uController008IPandPort)

Catch ex As Exception

MessageBox.Show("Socket Connection Error: " & ex.ToString, "Socket Error")

End Try

'When reading from the port, should you wait for a response? Or should you not wait?

'This is set so you do not wait - if nothing is there, move on.

'Without this line, the timer will lock up.

uController007Socket.Blocking = False

uController008Socket.Blocking = False

'Check if something came in with a timer (every 100 mSeconds)

Timer1.Interval = 100

Timer1.Enabled = True

End Sub

Private Sub Form1_FormClosing(sender As Object, e As FormClosingEventArgs) Handles MyBase.FormClosing

'Release the resources when you are done.

uController008Socket.Close()

uController007Socket.Close()

End Sub

```
Private Sub btn_7_Red_Click(sender As Object, e As EventArgs) Handles btn_7_Red.Click
```

```
Try
```

```
Toggle_Color(1)
```

```
'put the string you want to send into a byte array
```

```
OutMsg = Encoding.ASCII.GetBytes("R")
```

```
'bytesSent is an int that tells you how many bits got sent (If you need it)
```

```
bytesSent = uController007Socket.Send(OutMsg)
```

```
Catch ex As Exception
```

```
MessageBox.Show("Failed to Send to 192.168.0.7", "Transmit Error")
```

```
End Try
```

```
End Sub
```

```
Private Sub btn_7_Green_Click(sender As Object, e As EventArgs) Handles btn_7_Green.Click
```

```
Try
```

```
Toggle_Color(2)
```

```
'put the string you want to send into a byte array
```

```
OutMsg = Encoding.ASCII.GetBytes("G")
```

```
'bytesSent is an int that tells you how many bits got sent (If you need it)
```

```
bytesSent = uController007Socket.Send(OutMsg)
```

```
Catch ex As Exception
```

```
MessageBox.Show("Failed to Send to 192.168.0.7", "Transmit Error")
```

```
End Try
```

```
End Sub
```

```
Private Sub btn_8_Red_Click(sender As Object, e As EventArgs) Handles btn_8_Red.Click
```

```
Try
```

```
Toggle_Color(3)
```

```
'put the string you want to send into a byte array
```

```
OutMsg = Encoding.ASCII.GetBytes("R")
```

```
'bytesSent is an int that tells you how many bits got sent (If you need it)
```

```
bytesSent = uController008Socket.Send(OutMsg)
```

```
Catch ex As Exception
```

```
MessageBox.Show("Failed to Send to 192.168.0.8", "Transmit Error")
```

```
End Try
```

```
End Sub
```

```
Private Sub btn_8_Green_Click(sender As Object, e As EventArgs) Handles btn_8_Green.Click
```

```
Try
```

```
Toggle_Color(4)
```

```
'put the string you want to send into a byte array
```

```
OutMsg = Encoding.ASCII.GetBytes("G")
```

```
'bytesSent is an int that tells you how many bits got sent (If you need it)
```

```
bytesSent = uController008Socket.Send(OutMsg)
```

```
Catch ex As Exception
```

```
MessageBox.Show("Failed to Send to 192.168.0.8", "Transmit Error")
```

```
End Try
```

```
End Sub
```

Private Sub Timer1_Tick(sender As Object, e As EventArgs) Handles Timer1.Tick

Try

If (uController007Socket.Available > 0) Then

' Did anything come in from controller 7 (on any port number)?

' record how many bytes were received inside bytesreceived / the incoming data was stored in InMsg

bytesReceived = uController007Socket.Receive(InMsg)

Toggle_Color(5)

'convert the bytes in InMsg to chars, one by one. Start at 0 and go to bytesReceived

rtxt_From_7.AppendText(Encoding.ASCII.GetString(InMsg, 0, bytesReceived))

End If

Catch ex As Exception

MessageBox.Show("Receive Error: " & ex.Message, "ERROR from 192.168.0.7")

End Try

Try

If (uController008Socket.Available > 0) Then

' Did anything come in from controller 8 (on any port number) ?

'record how many bytes were received inside bytesreceived / the incoming data was stored in InMsg

bytesReceived = uController008Socket.Receive(InMsg)

Toggle_Color(6)

'convert the bytes in InMsg to chars, one by one. Start at 0 and go to bytesReceived

rtxt_From_8.AppendText(Encoding.ASCII.GetString(InMsg, 0, bytesReceived))

End If

Catch ex As Exception

MessageBox.Show("Receive Error: " & ex.Message, "ERROR from 192.168.0.8")

End Try

End Sub

Sub Toggle_Color(ByRef btn)

If btn = 1 Then

If btn_7_Red.BackColor = Color.FromArgb(255, 255, 255, 225) Then

 btn_7_Red.BackColor = Color.FromArgb(255, 255, 192, 192)

Else

 btn_7_Red.BackColor = Color.FromArgb(255, 255, 255, 225)

End If

Elseif btn = 2 Then

If btn_7_Green.BackColor = Color.FromArgb(255, 255, 255, 225) Then

 btn_7_Green.BackColor = Color.FromArgb(255, 192, 255, 192)

Else

 btn_7_Green.BackColor = Color.FromArgb(255, 255, 255, 225)

End If

Elseif btn = 3 Then

If btn_8_Red.BackColor = Color.FromArgb(255, 255, 235, 190) Then

 btn_8_Red.BackColor = Color.FromArgb(255, 255, 192, 192)

Else

 btn_8_Red.BackColor = Color.FromArgb(255, 255, 235, 190)

End If

Elseif btn = 4 Then

If btn_8_Green.BackColor = Color.FromArgb(255, 255, 235, 190) Then

 btn_8_Green.BackColor = Color.FromArgb(255, 192, 255, 192)

Else

 btn_8_Green.BackColor = Color.FromArgb(255, 255, 235, 190)

End If

Elseif btn = 5 Then

If Encoding.ASCII.GetString(InMsg, 0, bytesReceived) = "B1 " Then

 rtxt_From_7.SelectionColor = Color.Red

Else

 rtxt_From_7.SelectionColor = Color.Blue

End If

Elseif btn = 6 Then

If Encoding.ASCII.GetString(InMsg, 0, bytesReceived) = "B1 " Then

 rtxt_From_8.SelectionColor = Color.Red

Else

 rtxt_From_8.SelectionColor = Color.Blue

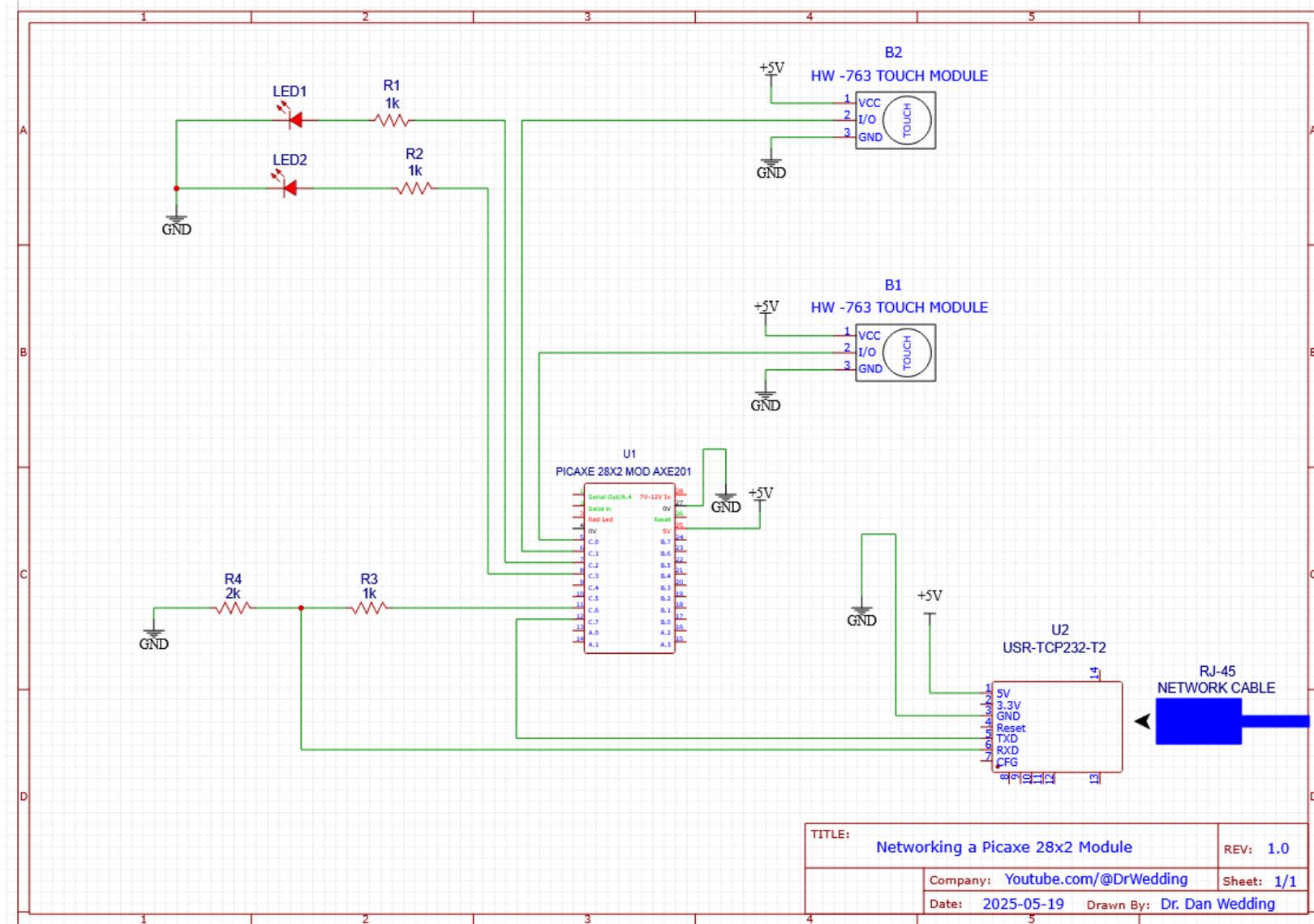
End If

End If

End Sub

End Class

PICAXE 28x2 SCHEMATIC - Youtube.com/@DrWedding VB25



TITLE: Networking a Picaxe 28x2 Module		REV: 1.0
Company: Youtube.com/@DrWedding		Sheet: 1/1
Date: 2025-05-19	Drawn By: Dr. Dan Wedding	

PICAXE BOARD PARTS LIST - Youtube.com/@DrWedding VB25

REFERENCE	QUANTITY	DESCRIPTION	PART	NOTE
U1	1	Picaxe 28x2 Module	Picaxe.com Picaxe 28x2 Module	Note 1
U2	1	Ethernet to serial UART conversion module	USR-TCP232-T2	Note 2
B1-B2	2	Capacitive touch module	TTP223 ALSO CALLED HW-763	
R1-R2	2	LED Current limiting resistor	1k Ω Resistor	Note 3
LED1-LED2	2	LED (Colors optional)	Generic LED	
R3	1	First Half of the voltage divider	1k Ω Resistor	Note 4
R4	1	Second half of the voltage divider	2k Ω Resistor	Note 5

Note 1: A Picaxe 28x2 was used in this video. Any other microcontroller should work, but the code will obviously change.

Note 2: The USR-TCP232-T2 Ethernet to serial UART conversion module does most of the work on the circuit side. This part needs to have its properties configured. The software to configure the part is named: **USR-TCP232-T24-V5.1.1.20.exe** and I downloaded it from the manufacturer's website here: <https://www.pusr.com/support/downloads/Setup-Software-USR-TCP232-T24-V51120.html>

Note 3: The current limiting resistors for the LEDs in this video were 1kΩ. The LEDs in this video were high efficiency, super bright, and low power. I also had the LEDs aimed directly at the camera. Your resistor values may be different depending on your needs. A lower value resistor, such as a 680Ω, 560Ω, 470Ω, or 330Ω will increase the brightness of the LED significantly.

Note 4: The RXD and TXD pins on the USR-TCP232-T2 can only handle 3.3V. See the bottom of page 14 of the user manual found here: <https://www.pusr.com/download/M0/USR-TCP232-T2-User-Manual-V1.1.pdf> The 3.3V TXD pin feeding the 5V RDX Arduino pin is fine. This can be a direct connection. However, the 5V TXD feeding the 3.3V RXD pin on the USR-TCP232 can damage the module. There are specialized modules called voltage levelers that do this. I chose to use two resistors to create a Voltage Divider Circuit. The equation is shown below:

$$5V * \frac{R4}{(R3 + R4)} = 5V * \frac{2k}{(1k + 2k)} = 5V * \frac{2k}{3k} = 3.3V$$

By choosing R4 to be 2kΩ and R3 to be 1kΩ, the voltage is exactly 3.3V.

Note 5: If you do not have a 2kΩ resistor, you can make one out of two 1kΩ resistors in series. You can use a potentiometer set to 2kΩ. Some people use a 2.2kΩ resistor in place of the 2kΩ. This will increase the voltage feeding the USR-TCP232-T2 to 3.44V and the maximum voltage the pin should be supplied is 3.45V (*See page 9 of the user manual*).

PICAXE CODE - Youtube.com/@DrWedding VB25

NOTE: Your code may be different if you use a different Picaxe.

```
;RUN ONCE CODE GOES HERE
#no_table
#no_data

symbol once0 = b1
once0 = 1

symbol once1 = b2
once1 = 1

'Button Pins
input c.0, c.1

'LED Pins
output c.2, c.3
low c.2
low c.3

'Serial Pins
output c.6
input c.7
high c.6

hsersetup B9600_8, $9 ;This sets the mode bits (most times it is $9 or $f)
hserptr = 0
hserinflag = 0
;Received values will be moved into B10, so clear it out
b10 = 0x00
```

main:

```
'Check to see if the first button is being pressed
if pinc.0 = 1 and once0 = 1 then
    serout c.6, t9600_8, ("B1 ")
    once0 = 0
elseif pinc.0 = 0 and once0 = 0 then
    once0 = 1
end if
```

```
'Check to see if the second button is being pressed
if pinc.1 = 1 and once1 = 1 then
    serout c.6, t9600_8, ("B2 ")
    once1 = 0
elseif pinc.1 = 0 and once1 = 0 then
    once1 = 1
end if
```

```
'Check to see if a value showed up in the UART while
the uController is doing something else
'The incoming information was stored in the scratchpad
if hserinflag != 0 then
```

```
    pause 10 ;In case it has not finished reading yet
    get 0, b10
```

```
    ' Toggle the selected LED
    if b10 = "R" then
        'if I got an R from VB - toggle c.2
        toggle c.2
    else if b10 = "G" then
        'if I got a B from VB - toggle c.3
        toggle c.3
    end if
```

```
    'Reset the UART values
    hserptr = 0
    hserinflag = 0
    b10 = 0x00
```

```
end if
```

```
goto main
```