



Applied Design Laboratories, Inc.
200 Litton Drive, Suite 215
Grass Valley, CA, 95945
E-Mail: ron2008(at)applied-design.com

COMMUNICATING WITH A BASIC STAMP VIA WIFI AND THE INTERNET

Goals: Remote Internet access to reports, Sending automatic emails and text messages.

I built a new controller based on a Parallax Basic Stamp 2Px for my failing water well to protect the pumps from losing water. I included a Quatech "Airborn" serial to WiFi module to allow internet access and alarms. This note will cover the basics of how I set up the RS232 communication with the Quatech modules to accomplish the goals above.

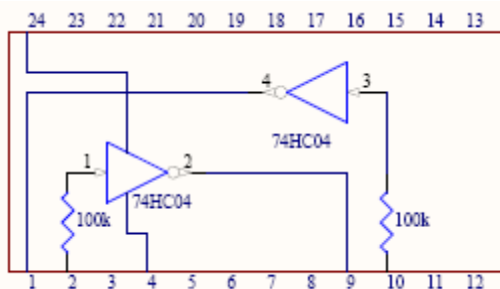
I used a Quatech WLNG-AN-DP101 802.11-to-serial module available from Mouser. It operates on 3.3v instead of the Basic Stamp's 5v so requires a different power supply and some buffering of signals. The connector is miniscule and requires you make a PC board. It talks to my home wireless network.

The overall project of controlling my well pump based on measuring the depth of water in the well is not the subject of this note. I will give the methods to use the Quatech module that worked for me.

Setting up the Quatech module.

For debugging and setup I made a little header that I could plug in my board instead of the Basic Stamp that would interface the serial port normally used to load the Basic Stamp to the Quatech module's RS232 port.

Here is a schematic of the header I made:



Note that my PC board included a resistor and protection diodes on the line from the Basic Stamp's port(5v logic) to the Airborn module's RX port (3v logic).

Pin 1 is Serial Out, Pin 2 Serial In and should be connected on your board to a DB9 receptacle (pins 3 and 2 respectively). This is the normal

connection for the Basic Stamp programming and debug. Note I used pin9 (P4)and pin 10(P5) for the RS232 connections to the Airborn module.

If you connect a terminal or a PC with a serial port running a program like Hyperterminal to the serial connection, you can talk to the Quatech/Airborn module directly. Set it up for 9600 baud, the default rate of the module. Make sure it is working-type a few random keys and you should get an error message.

Here are the setup commands you will have to execute, they can also be executed if you access the module by TCP from your network over the radio. They are detailed in the Quatech manual ***Airborne™ Product Family Command Line Interface (CLI) Reference Guide*** For use with: *WLNG-AN-DP100 Series, WLNG-SE-DP100 Series, WLNG-ET-DP100 Series, ABDG-SE-DP100 Series, ABDG-ET-DP100 Series, Document Number 100-8005-101G.* Follow each command with an enter.

Sign on with the default authentication-you can change this later if you are paranoid.
auth xxxx xxxx
OK

W1-ip xx.xx.xx.xx (the fixed IP address you will use for the module, usually 198.162.xx.xx or something) You can use DHCP if you really want to.

W1-DHCP 0 (disables the DHCP server for a fixed address)

W1-TCP-ip xx.xx.xx.xx (the IP address of your outgoing email SMTP server)

W1-TCP-port 25 (The telnet port of your outgoing email SMTP server)

W1-ssid xxxx (the name of your wireless server if you hide it-recommended)

Bit-rate 9600 (change it if you want to use a different RS232 bit rate)

Serial-default listen (I use this so it will start up in a mode accessible from the internet)

commit (this puts all the values you entered in EE memory)

restart (start over with new values).

RS232 interface between the Basic Stamp and the Quatech module.

This interface only has to deal with the lower voltage for the Quatech. I connected TXD of the Quatech module (pin 28) directly to pin 10 (P5) of the Basic Stamp which is my serial input. The 3.3v logic is sufficient to connect to the Stamp. I connected pin 9 (P4) which I used for my serial output of the Basic Stamp through a 100k resistor to pin 24 (RXD) of the Quatech module, and added a protection diode from that pin to V3.3 to limit the voltage on the serial out signal.

Sending reports from the Stamp that can be monitored from the Internet.

I wanted to monitor my well control and get status reports if I am at work or out of town. This can be done by sending periodic reports out the Stamp's serial port to the Quatech module. I send reports every minute and only send out a line feed (10) character when new things are happening so it normally writes over the same line when I am monitoring the well.

To access your reports from the internet, run a telnet program such as hyperterminal or the Telnet command from a DOS window. Set it up to access your external IP address (if you have a fixed IP address, use that otherwise it will be an address your ISP will have to set up for you). The default port is 23.

If you run the terminal or telnet program, you should be able to sign on to your module with the same sign-on you used locally – "auth xxxx xxxx ␣" If it works, you will get an OK

Then type "pass ␣" and you will get another OK, then you should start getting reports from your Basic Stamp program.

You can also, at this point type commands to your Basic Stamp program and if you are scanning the serial port they will be registered. I use this to allow a me to do remote reset of my well controller from wherever I am.

This is a typical line of Basic in my program that sends part of the report:
SEROUT 4, Baudmode, ["went low ",DEC j," cy,off for ",DEC i/60," min"]

Email text messages for alarms.

Telnet provided the first ways to send email in Unix systems and the commands are still supported in email servers. My goal was to send a message to a remote SMTP server. If you run your own email server, the linkage is slightly different.

See the section on Gotchas for a manual way to test your SMTP/Telnet connection. I recommend that you run the sending email dialog first manually from a Telnet connection (Telnet on MSDos in Win XP, or hyperterminal) without any of the hardware you have built.

To get your hardware working, first you must get the Quatech module set up to have your SMTP server's address in it's W1-TCP-IP address. And 25 for its port-the default SMTP port (see detailed commands above). Your mail server may use

different ports. Then you send "pass" to the module to hook up to the remote SMTP server. A message like this will come back:

220 smtp.perfora.net (mrus1) Welcome to Nemesis ESMTP server

Send "HELO yourmailcarrier.com"

A message similar to this will come back: 250 smtp.perfora.net

Send "auth login"

A message similar to this will come back: 334 VXN1cm5HbWV6

Next you need to send your username in Base64 code – use this tool or similar to convert the text for your username and password into Base64 code

<http://www.opinionatedgeek.com/dotnet/tools/base64encode/>

Enter "your username in Base64 code"

334 xxxxxxxxxxxx

Enter "your password in Base64 code"

235 Authentication successful

IF THIS DOES NOT COME BACK THERE IS SOMETHING WRONG.

Enter "MAIL FROM: your email address"

250 OK

Enter "RCPT TO: yourcellphone#@vtext.com" (Verizon only-use your cellphone company's service)

250 OK

Enter "DATA"

354 Enter mail, end with "." on a line by itself

Enter "Alarm message"

Enter "" (Send a blank line to separate the headers from the message body)

Enter "This is the message"

Enter "."

250 Message blah-lah

Enter "quit"

221 OK

The server will return: "250 Message ----- accepted by mrus1.performa.net" or similar

Enter "quit"

I then wait for a while watching for a remote "r" key then reset the module as the easiest way to get it back to listening for remote access.

Sending messages from the Stamp to email.

Most cell phone carriers have a way to send an email to generate a text message.

My cellphone is on Verizon so I set up the alarm output to send an email to

mynumber@vtext.com and as long as the message is less than 160 characters it will be sent as a text message to my phone. The report following will generate such a text message. For details on other carriers see <http://www.popularmechanics.com/technology/how-to/4318204>

Gotchas.

I wasted an incredible amount of time trying to get the outgoing email feature to work. I found that I could not get Telnet messages to go through my wireless router (I tried Linksys WRT54G with their firmware and later with Tomato firmware, and a D-link DI-624). No matter how I configured it, the telnet message would not come out the WAN port to go to the Internet. I finally left the WAN port empty and hooked my Internet modem to one of the local ports. I had to reconfigure my computers to be able to access the Internet (mostly filling in local gateway and DNS addresses).

I could not get one of my Airborn modules to run at 19,200 baud, the default RS232 speed of the Basic Stamp 2Px's debug command. I could not resolve this problem so I changed all my comm with the Airborn module to 9600 baud. The DEBUG command saves code so is preferable if you can get 19200 to play or know how to change the speed of the DEBUG output.

The Airborn module has an indicator output for the radio link. You should connect this to an LED as you will need it for debugging. Normally it blinks every few seconds when it is trying to connect, then stays on when it is connected. However the LED being on does not necessarily mean there is a strong enough signal for it to communicate. The modules seem to have weak radios that would not reach my wireless router in the house about 50' away, even fooling around with directional antennas. I solved this by putting a Logitech wifi repeater in the pumphouse with the modules. The repeater has no trouble reaching the router, and I can use a simple antenna on the side of my controller for the Airborn module (the antenna and cable that Mouser sells with the modules). Hook your antenna to the ant2 microplug.

There seems to be a mode that the Airborn module gets into where it is totally non-responsive, even to its hardware reset pin. The only way I was able to make it run again was to power down and drain all the power supplies with a ground. The manufacturer was mystified about this and had no suggestions.

Conclusions.

It is indeed possible to talk to a Basic Stamp application over the internet. Just don't take it on as a rush project!