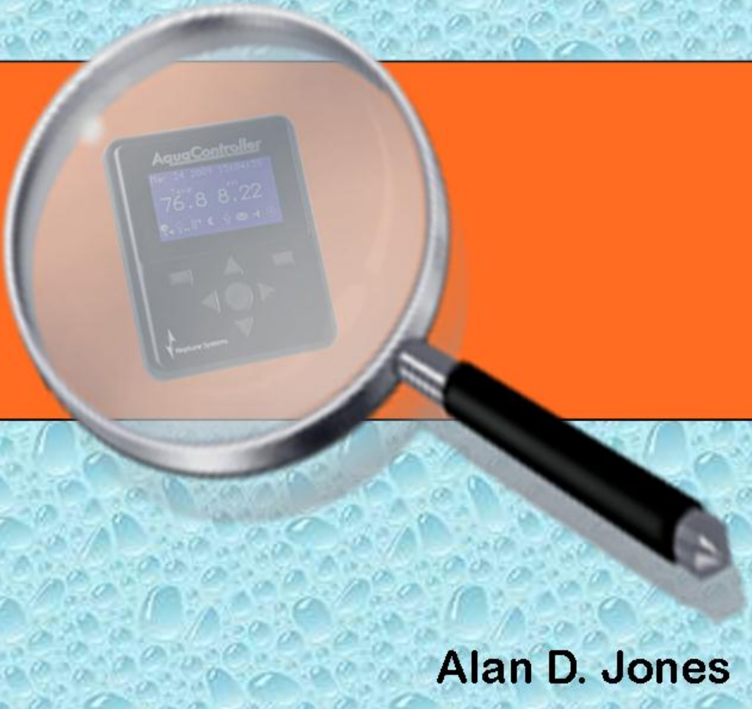
The Unofficial New User Guide

**A Comprehensive Guide
to the Installation and Operation
of the AquaController Apex
and the AquaController Apex Lite**



Alan D. Jones

The Unofficial

New User Guide

A Comprehensive Guide to the Installation
and Operation of the AquaController Apex
and the AquaController Apex Lite

By Alan D. Jones

Version 16

Contents

Acknowledgements

This would not have been possible without the input and ideas from a number of people. I am grateful to Russ Michel for his input on networks and network troubleshooting and who kept me honest with most of the “geek speak”. Ken Argo who has immense experience with these controllers and all things Windows reviewed the initial draft and corrected some of my many errors. He’s a programming genius. And Curt Pansegrau the owner of Neptune Systems who found time in his busy day to proof this and who has had the patience to answer my endless questions and emails. Finally, all the members and contributors on the ReefCentral Neptune Systems forum whose ideas and experiences I’ve leveraged for input and solutions. To all of you I offer my sincere appreciation.

Preface

A few short years ago the extent of aquarium automation was some home light timers given new duty managing the metal halide lights over the tank. Eventually someone cobbled together some controls and released the first dedicated aquarium controller. When the X10 standard was developed for home automation, it was natural to extend it to the aquarium. This dramatically lowered the entry cost but still left a lot to be desired. The Internet with its promise of access anywhere, anytime led to the integration between controller and web. Now you not only control the timing and activation of devices but through probes and sensors you can take a real-time view of your tank and adjust equipment as necessary, even sending yourself email or text messages as conditions warrant. Neptune Systems has brought all of these capabilities together into the Apex line of controllers.

When I first opened the box containing my brand new Apex AquaController, I was absolutely stumped – where do I start? Through trial and error and with the help of many others going down the same road as myself along with fantastic Neptune support, I managed to get the controller installed and configured. Now, I don't know what I would do without it. It really is a marvelous piece of electronics. This document is intended to help ease the process for you and hopefully answer some of the many questions you might have.

If you have comments or suggestions for improvements to this document, please let me know at www.reeftech.webs.com .

Good Luck and Happy Reefing
Alan Jones

Chapter 1: Getting Started

Apex vs. Apex Lite

The AquaController Apex and the AquaController Apex Lite share the exact same cases, run the exact same software and are compatible with the exact same accessories. There are two differences between the Apex and the Apex Lite:

1. The Apex base unit supports 3 probes (Temp, pH, and pH/ORP). The Apex Lite only supports 2 (Temp, pH).
2. The Apex has 4 variable voltage ports that can be used for controllable pumps (i.e. Tunze) or dimmable lighting (i.e. VHO ballasts). The Apex Lite has no variable voltage ports.

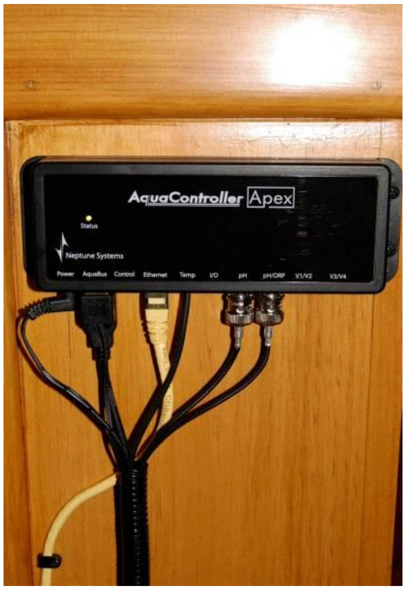
To give the Apex Lite similar capabilities as the Apex, you would need to purchase the AquaSurf module (will give you 2 (vs. 4) variable speed ports but does NOT support dimmable lighting) and a probe expansion module (PX1000, PM1 or PM2 depending on the probes you want). The Apex is typically bundled with the base unit, a display module, a single EB8 and two probes (temp and pH). The Apex Lite is bundled the same except with a single temp probe. Of course, you can purchase any of the modules or probes separately as well.

Throughout this document "Apex" refers to either the Apex or Apex Lite unless noted otherwise.

Apex and the Mac

Unlike its predecessors, the Apex uses a browser for PC control. It does not use a stand-alone application (i.e. Aquanotes) like the AC and AC III did. As such, the Apex will work just fine with a Mac and Safari (also Firefox, Internet Explorer). However prior to version 3.3 of the firmware loader application, you could not apply firmware upgrades from a Mac. Version 3.3 of the loader was distributed with release 4.02 of the firmware. With version 3.3 that restriction has been removed and you can apply firmware upgrades with your Mac (see *Chapter 11: Firmware Updates* for more information).

Installation



This picture shows how I mounted the base module to the side of the stand. The cables from left to right are: 12V dc supplemental power, AquaBus to Display module (behind this cable is a second AquaBus cable going to the EB8, Ethernet cable (yellow) going to your home router, Temperature probe, pH probe and ORP probe (these have the BNC style connectors). There is no AC power cable going to the base module – its primary power source is the AquaBus.

I recommend mounting the base module in an easily accessible location where you can get at the various connections and cycle the power when you need to. I mounted mine to the outside of my stand. I also used corrugated wire wrap to consolidate and organize the many cables going to the base unit. It trims it out nicely.

This photo shows the Display module mounted on the side of the canopy at eye level. The Coralife temperature probe is in the sump showing a slightly cooler temperature. With expansion modules you can add multiple probes to the Apex system.

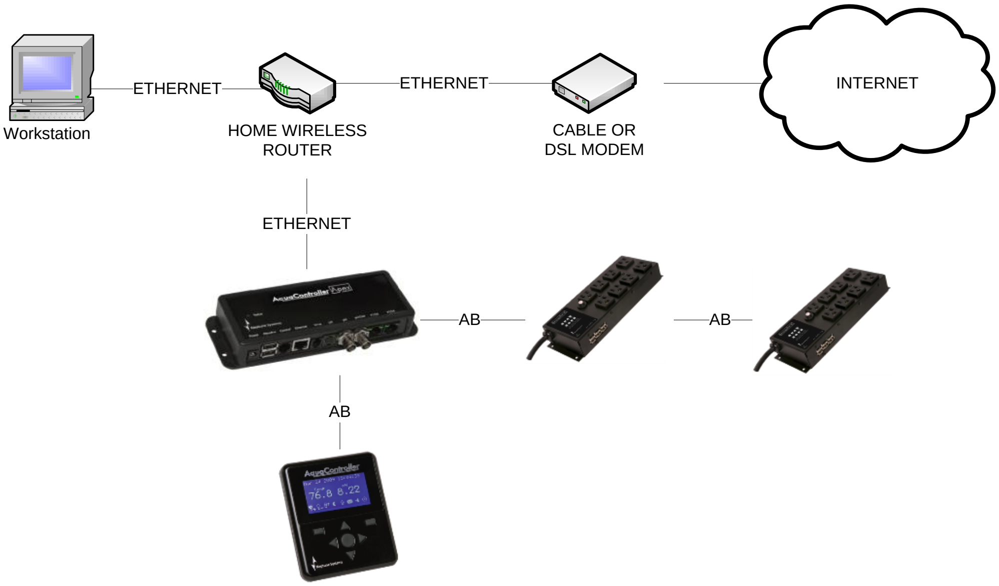
The display module comes with a 10' AquaBus cable and can connect to the display or any other AquaBus module. You can connect multiple display modules if you want.

The EB8 is mounted inside the stand where the various power cables can be easily connected.

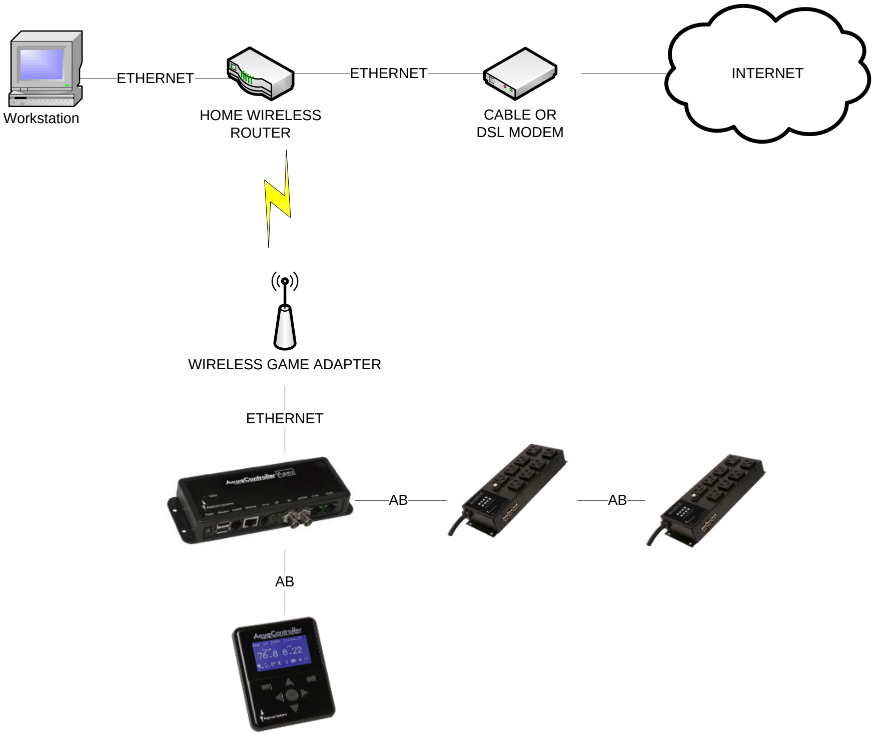
The AquaBus can be a total of 200' in length giving you ample opportunities to locate the modules in convenient locations. Most of the bus and power cables are 6' but you can source longer USB cables on the Internet. You have to make sure there are no powered repeaters on long cables as this will damage the base unit.

You can connect the EB8's AquaBus cable to the base unit or any other AquaBus type module including another EB8, PM1, PM2, PM3 or WXM.

The following diagram depicts how a typical connection might look. 'AB' represents a USB style AquaBus cable. Ethernet is a straight through CAT-5 or CAT-6 Ethernet cable, available at any electronics retailer or Wal-Mart.



If you were to connect the Apex via wireless instead of a wired connection, it would look like this. **However it is highly recommended that you first establish a wired connection and use that to configure your local network access, Internet access and email. Doing it in this order eliminates a lot of startup troubleshooting and headaches.**



Modules and Outlets – an overview

The Apex system is conceptually built around modules and outlets. You purchase modules but program outlets. There are three basic modules with every Apex system, the Apex Base module (manages Processing and Storage), the display module (user interface) and the energy bar module called the EB8. You can add more modules but this will get you the basics including 8 receptacles, a remote display unit with push button operation and the control unit which houses the CPU, communications, memory, variable speed pump receptacles, I/O switches, and where you can connect your probes. Neptune makes additional modules for I/O switches, probe expansion, high amperage outlets, etc. These "control port" modules (DC8, DC4HD, LunarSim) don't use the newer 'AquaBus' communications channel (AquaBus modules use a USB style connector) – but they're still 100% compatible with the Apex. The EB8 module has 3 pairs of AquaBus ports for a total of 6 ports. You can connect any AquaBus module to any other AquaBus module.

The Apex base module is 'plug and play' and will automatically recognize AquaBus modules as soon as they are connected to it and in the case of the EB8 it will automatically create 8 outlets for you with various default programs (pump, light, heater, etc.). It will also create 4 variable speed ports to be used with 0-10v controlled powerheads (i.e. Tunze powerheads that are compatible with the 7091 or 7095 Tunze controllers) or dimmable lights and 3 virtual outlets for audible alarms and email alerts (the Apex Lite does not have the variable speed ports). You can change these as you wish during programming. Modules are automatically numbered as they're added to the AquaBus beginning with 1. Sometimes there will already be modules declared from the factory as a result of Neptune's final QC steps. If this is the case, you might find your EB8 numbered 4 or 5. If so, you can run the *Setup → Module Setup → Delete Module* command from the display module.

Outlets can be virtual or physical. The physical outlets are the receptacles on the energy bar. Virtual outlets (sometimes called "dummy outlets") can be defined almost as "working storage" variables that you can set on/off based on conditions, then later test those outlets and perform even more complex operations. Virtual outlets also include email and alarms. Default outlets are configured automatically when the base module first "sees" your EB8. These default outlets can be used as-is or changed. You can change the name and programming of an outlet via the outlet setup screen. You can add/delete outlets via the module setup screen. The name of an outlet is completely up to you but is limited to 12 characters, no spaces, is case-sensitive and must be unique. The default Neptune naming convention is "device_x_y" where X = the AquaBus address and Y = the receptacle number (1-8) of that EB8. You don't have to follow that convention. You can name an outlet "Lights" if you want.

A couple of notes about the EB8 module. These come with 8 receptacles, numbered 1 - 8. Outlets 4 and 8 are activated with mechanical relays. These have a distinct "click" whenever they switch on/off. Mechanical relays have been around for a long time, are inexpensive and very reliable. They handle both high and low current devices well but are large and bulky compared to solid state alternatives. But, they are known to 'arc' and have their contacts 'pit'. And, as far as switches go, they're relatively slow which is a problem in some applications.

Outlets 1-3 and 5-7 are TRIAC (Triode for Alternating Current) solid state outlets. They are silent, compact and capable of hundreds of thousands of on/off cycles. They are very, very fast and can switch in fractions of a second and do that all day long without wearing out like a mechanical switch/relay would. Another advantage that TRIAC switches have over mechanical switches is that they are very good at switching high current, inductive (lighting) loads because they "time" their switch to the point when the current sine wave is at or near zero (what Neptune calls 'soft start'). This virtually eliminates the possibility of “arcing” that can happen with mechanical relays that are indiscriminate about where in the sine wave they open/close.

One problem with TRIAC controlled outlets however is that they sometimes cannot detect very low power factor devices and as a result may not switch off correctly. They fail in an ‘open’ state. This only happens with very low power devices (i.e. < 4-5 watts). However, if you have any device like that, you might consider putting them on outlets 4 or 8.

When you consider that Neptune has to design switching outlets that control everything from dosing pumps to metal halide lights, be able to switch in tenths of a second and last for years and not cost an arm and a leg, you realize that there is no one perfect type of switchable outlet. Both the mechanical relays and the solid state TRIAC have strengths and weaknesses. Understanding how they work will enable you to make an informed choice on how you layout your devices and outlets.

The total amperage draw of all outlets for an EB8 is 15A which is the maximum load an average household circuit can handle anyway. TRIAC outlets are designed for a maximum of 5A, the relay outlets 10A. If you need more outlets that can handle large loads, Neptune makes the DC4HD module for that.

Also, the EB8 module has a virtual “probe” that reports total amperage load – displayed on the status screen in real time and logged in your data logs. This is the total load for the EB8 - it can't currently tell you the load by outlet. However, by turning on the outlets one by one, you can deduce how much each device draws. You can then shuffle them around to different outlets if necessary. The probe is named “Amp_n” where “n” is your EB8 AquaBus address.

One final comment regarding modules. Modules are connected to the base module and each other via a communications/power bus called the “AquaBus”. The cable looks like an ordinary USB cable with type “A” male connectors at each end.   However, unlike USB which operates with 5v signaling, the Apex uses 12v on the AquaBus. This is the way Apex modules including the base unit get their power. The total distance of the AquaBus “chain” from Apex Base to final module must be < 200’. You could do this in eight 25’ sections if you wanted (the supplied cable length is 6’). If you use a long USB cable (i.e. > 15’) you must insure it does not include any boosters or repeaters that are typically included to make long USB cables work. Here is one source (www.usbfirewire.com) but do a little searching as there are others. **Do not plug normal USB devices into the AquaBus – something is going to get fried, probably your Apex since it’s probably the most expensive!**

Probes

A probe is a device that monitors and reports something back to the Apex like temperature or pH. The Apex comes with a temperature probe and a pH probe as part of the standard bundle (also included is the base module, display module and an EB8 module). The Lite only includes a temperature probe. You can add a second pH probe or an ORP probe as part of the base control module (the Apex Lite does not permit this third probe).   In addition, there is a virtual probe created with every EB8 called “Amp_n” where “n” is the number of your EB8. This is the probe that reports your EB8 amperage. You can also add additional probes including Dissolved Oxygen and Conductivity via a probe expansion module.   All of Neptune’s probes can be completely submerged without damage.

The temperature probe can be used right out of the box. Plug it into the base module and it will begin reporting your temperature immediately. However, if you suspect it is inaccurate you can calibrate it against a known reference (Neptune recommends using a thermometer that's 'NIST Traceable', available on the web for under $60). Calibrate using *Setup → Temp Setup → Temp Calibrate* using tenths of a degree +/- to add/subtract to whatever reading it has.

The pH probe has to be calibrated before you can use it. Again, the manual has instructions on how to do this. Basically you use two reference solutions to establish the low and high points for the probe. These solutions are pH 7.0 and 10.0 respectively. One common mis-conception is that during calibration, the pH probe should eventually read 7 or 10. It may, or it may not, that's why you're calibrating it. It is immaterial what the probe reads during calibration, only that the calibration numbers stop changing. When you are finished calibration, you can then test the probe by putting it back into either the 7.0 or 10.0 solution after first returning the display to the Home screen. It should read 7 or 10 after a minute. If it doesn't, then you should go through the calibration steps again.

Don't FORGET!

The probe does not average if you calibrate it multiple times - each calibration writes over top of any prior data. Also remember to rinse the probe in plain tap water between solutions when calibrating or testing. You can easily transfer solution and distort the readings.

You have an option when calibrating or using your pH probe to include temperature compensation or not. There are a couple of things you should consider. First, practically speaking the compensation is negligible given the very narrow range of temperatures in most reef tanks. A pH of 8.30 at 77 degrees would be 8.29 at 75 degrees and 8.31 at 79. Also, if you do compensate, you have to be careful of what happens if your temperature probe reports a strange value – do you immediately respond with dosing if your temperature probe blips? (See the Defer command for a way to get around this). On the other hand, the capability is there and you can decide if you want to use it or not. You activate it via the display module using *Setup → pH Setup → Temp Comp.*

The ORP probe comes calibrated from the factory and should not require calibration in normal use. If however you feel you do need to recalibrate it, there are instructions in the manual on how to do it. It is not as simple as the pH probe calibration and requires a special powder called *quinhydrone* along with pH 4 and pH 7 solutions. There is much information on the web as to this process and suppliers of the necessary solutions.

pH and ORP probes do require cleaning as algae and oxidation build up on the glass electrodes. Using a soft bristle brush and gently (stress gently) scrub the bulb will remove most residue. You can also soak the probe in vinegar for several hours if plating is on the probe. Make sure you rinse the probe in regular tap water when done cleaning.

Finally, there's one other type of a probe that bears mention – that's the serial switches that are part of the I/O Breakout Box (see *Switches*). You cannot alter the value of a switch (like you can with *outlets*) – they report a condition, in this case OPEN or CLOSED so they behave more like a probe. Switches can be referenced in any outlet program.

Probe Placement

Probes can be placed anywhere in your tank or sump. However, it is best practice to keep probes at least 6" apart. It is not necessary to keep your temperature probe next to your pH probe as was the requirement with the older AC line of controllers. Probes can be fully or partially submerged as they are all fully sealed against water, however the cable insulation can become brittle with constant immersion in salt water. Also, it is important to ensure that any of the bulb type probes (pH, ORP) stay wet at all times as they can be damaged with exposure to air.

Probe Interference

All current Neptune probes are electrically isolated from other controller electronics (Galvanic Isolation) and insulated against electrical interference (EMI). However, high voltage MH ballasts can still occasionally cause erratic readings for brief periods. Here are some things you can do to minimize probe interference if you experience sporadic problems with probe readings:

1. Physically separate probes and ballast wiring as much as possible.
2. If you have to cross wires, cross them at 90 degrees to minimize interference.
3. Coil any extra probe wiring to reduce any antenna effect.
4. Install 'chokes' on your ballast wiring (available at Radio Shack).
5. Take advantage of the 'Defer' command on your probe statements to eliminate brief value swings (doesn't solve the interference problem but does minimize its symptoms).

Chapter 2: Configuring your new controller

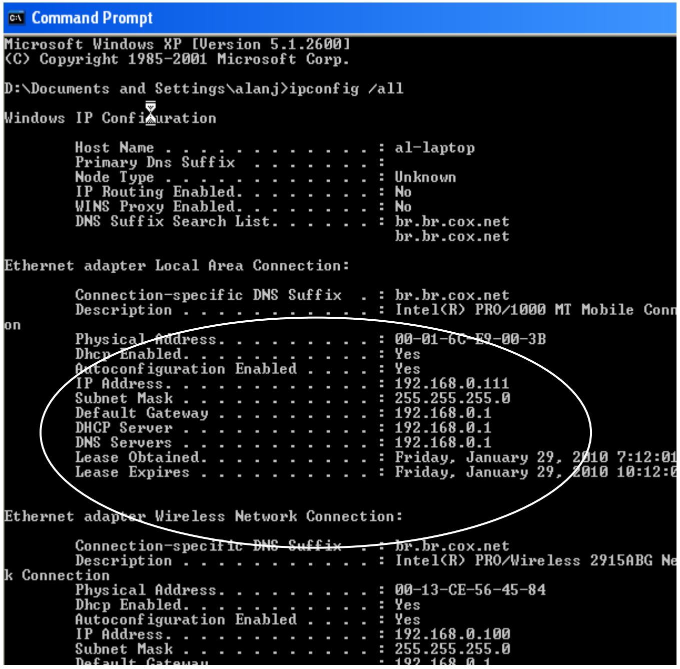
Before you start configuring your Apex, there are a couple of things you should do. First, you really need to have a router in your home if you want to use a PC to control it – while you can connect the Apex directly to a PC without a router, it becomes more complicated. Second, do not attempt to configure wireless unless you have communications working via a hard wired connection – you'll just make everything harder for yourself. Finally, throughout the process of configuring your network, email and even loading firmware updates, it's helpful if you have and know how to use the DOS command prompt and Telnet for troubleshooting. If you don't know what these are or how to use them, see the FAQ section. Also there are plenty of resources on the web.

The steps you should follow in summary are:

1. Find a workspace near your PC and router where you can connect everything together – you don't need to be near your tank.
2. Plug the 3 modules together with their AquaBus cables but don't plug anything into the outlets yet.
3. Connect the EB8 into a wall outlet to power it up.
4. Verify you have the right number of modules (especially EB8's) declared (see "*I have more modules declared than I physically have*" if you do).
5. Verify your Apex's network setup using the display module.
6. Connect the Apex to your router and make sure you can access the Apex from your PC.
7. Verify your firmware – having old firmware will just make it more difficult.
8. Configure email and make sure that's working.
9. Configure your Apex for external Internet access. Do this up front and Neptune support can frequently telnet directly into your controller to help support you.
10. Now you can program some outlets.

Network setup

The first thing you want to do when connecting your Apex is establish communications with it.   For most home networks running Windows, simply connect an Ethernet cable between the Apex and your router and power up the Apex. You should see a yellow blinking light next to the Ethernet port on the base module. The Apex should request an IP address from your router, your router will respond and after a brief period, the Apex will be registered on your network. You should be able to access it using http://apex in your browser's address bar. If not, it's easy to modify for your network.



```
Microsoft Windows XP [Version 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

D:\Documents and Settings\alanj>ipconfig /all

Windows IP Configuration

        Host Name . . . . . . . . . . . . : al-laptop
        Primary Dns Suffix  . . . . . . . :
        Node Type . . . . . . . . . . . . : Unknown
        IP Routing Enabled. . . . . . . . : No
        WINS Proxy Enabled. . . . . . . . : No
        DNS Suffix Search List. . . . . . : br.br.cox.net
                                            br.br.cox.net

Ethernet adapter Local Area Connection:

        Connection-specific DNS Suffix  . : br.br.cox.net
        Description . . . . . . . . . . . : Intel(R) PRO/1000 MT Mobile Conn
on
        Physical Address. . . . . . . . . : 00-01-6C-E9-00-3B
        Dhcp Enabled. . . . . . . . . . . : Yes
        Autoconfiguration Enabled . . . . : Yes
        IP Address. . . . . . . . . . . . : 192.168.0.111
        Subnet Mask . . . . . . . . . . . : 255.255.255.0
        Default Gateway . . . . . . . . . : 192.168.0.1
        DHCP Server . . . . . . . . . . . : 192.168.0.1
        DNS Servers . . . . . . . . . . . : 192.168.0.1
        Lease Obtained. . . . . . . . . . : Friday, January 29, 2010 7:12:01
        Lease Expires . . . . . . . . . . : Friday, January 29, 2010 10:12:0

Ethernet adapter Wireless Network Connection:

        Connection-specific DNS Suffix  . : br.br.cox.net
        Description . . . . . . . . . . . : Intel(R) PRO/Wireless 2915ABG Ne
k Connection
        Physical Address. . . . . . . . . : 00-13-CE-56-45-84
        Dhcp Enabled. . . . . . . . . . . : Yes
        Autoconfiguration Enabled . . . . : Yes
        IP Address. . . . . . . . . . . . : 192.168.0.100
        Subnet Mask . . . . . . . . . . . : 255.255.255.0
        Default Gateway . . . . . . . . . : 192.168.0.1
```

If you know the IP address of your home network, great. If not, either log into your router's administration function and see what address range is configured or from your PC's DOS do "ipconfig /all". It will list the IP address of every adapter. You're looking for the "Ethernet adapter Local Area Connection". Note the 4 pieces of the IP address (they're called "octets"). It should be "192.168.0.xxx" or "192.168.1.xxx".

Go into the display module and look at the IP address (*Setup* → *Net Setup* → *IP Address*). It must match the first 3 octets of your home address. If it doesn't, then back out and check the DHCP setting (*Setup* → *Net Setup* → *DHCP*). It should be "on".   Check your port (*Setup* → *Net Setup* → *Http Port*). It should be "80" which the default is. Save and exit. Your base module will reboot. Again, from your PC browser, enter http://apex. If that doesn't work, try manually entering your controller's IP address from your browser address screen as http://192.168.1.50 or whatever the address in your Apex is. You should get the login screen. If you don't, go back in and double check the settings again – look at your IP address, DHCP setting and port number. Also, sometimes you can have everything configured properly but it still doesn't work – rebooting the Apex and your router can sometimes clean it up. Do the router first, then the Apex.

Note, if you have a Mac, the http://apex command will not work. This is called NBNS (NetBIOS Name Service) and is strictly a Microsoft capability. Enter your IP address in Safari manually as http://192.168.1.50 (example).

Network Troubleshooting

If you're still having difficulty establishing communications with the base unit, take a deep breath and start eliminating what's not the problem. The Apex should connect to most home networks straight out of the box but frequently something very simple is configured incorrectly. The assumption here is that you're connecting via a wired Ethernet connection to an existing, working router.

Step 1 – Verify your physical connection. Do you have a solid green and a flashing yellow light at the Ethernet connection on the Apex base unit and the same (or at least a flashing light on the front panel) of your router? If you don't, stop. No sense going further. You don't have a proper physical connection to your router for some reason. Check the cable, perhaps try a different cable. You have to have the green and yellow lights before proceeding.

Step 2 – Verify your network settings. From your desktop PC, start the DOS and do the 'ipconfig /all' command or go to your Network Properties, select your Ethernet adapter and get the data from your TCP/IP properties. Complete the following table for your Home PC (using the *Ethernet adapter Local Area Connection* values from your PC):

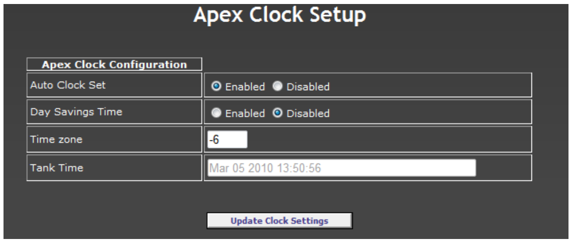
Network Parameter	Home PC	Apex
Subnet Mask / Net Mask		
Default Gateway		
DHCP Server		
DNS Server		
Alt DNS Server		

Now go into the Apex via the display module and select the *Net Setup* menu (from either the *Setup* or *System* menu depending on your software version). Select *Net Mask, Gateway, DNS Server* and *Alt DNS Server* one by one, recording their values in the above chart under 'Apex'. It's possible that your home PC does not have an entry for Alt DNS Server. If not, leave it blank (0.0.0.0). Also, the Apex uses the same value for DHCP Server as it does for DNS Server so complete them accordingly. For most home networks the PC's Default Gateway, DHCP Server and DNS Server should have the same IP address and it will be the IP address of your router.

All of the Apex values in the above chart must match the Home PC values. If they don't, go into the Apex via the display menu and change them so they match. Reboot the Apex when you are finished. Give it a minute to obtain its IP address, and then try again to access it via http://apex . You should get a login prompt. If you still cannot connect, go back and double check your values. If they all appear to be correct, go back into the Apex via the display menu and go to the IP Address option. Record what that is and exit. Then from your PC try to access the Apex by typing the following in your browsers URL address bar: http://192.168.0.50 or whatever your Apex's IP address is.

Step 3 – Eliminate your browser. If you still can't access the Apex, go back to your DOS command window and try to telnet to that address using the following syntax: 'telnet 192.168.0.50' again, using your Apex's IP address. If you still can't access it, then email Neptune support (support@neptunesys.com) or post a thread in the ReefCentral.

Clock Setup



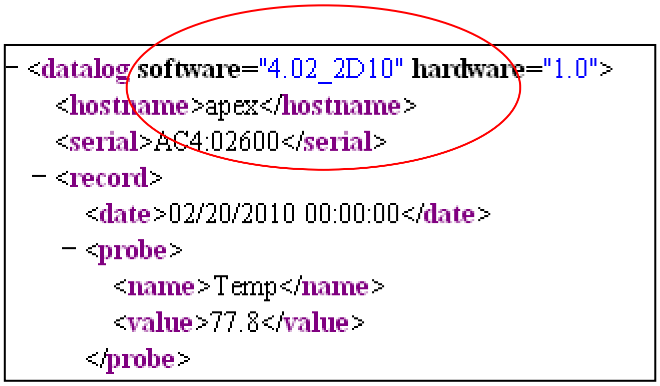
Apex Clock Setup

Apex Clock Configuration	
Auto Clock Set	● Enabled ○ Disabled
Day Savings Time	○ Enabled ● Disabled
Time zone	-6
Tank Time	Mar 05 2010 13:50:56

Beginning with version 4.02, the clock is configured differently. In the past you simply entered the date/time on the *Configuration – Network Setup* page. With 4.02 the clock is given its own configuration page because of a couple new options. First, the clock now gets synchronized regularly against a group of NTP servers (Network Time Protocol). This insures that any inaccuracies in the clock are erased every 18 hours. Second, as a result of using NTP, you now must enter your proper time zone against Greenwich Mean Time (GMT) since that's the time NTP uses. Negative numbers are for locations West of Greenwich England, positive numbers for locations East of Greenwich, England. In the U.S., Eastern is -5, Central -6, Mountain -7, etc. You also can select Daylight Savings Time which will adjust your time offset for DST (GMT itself does not change in the summer). The NTP servers are part of the NTP Pool Project. Visit the NTP Pool website to learn more about NTP and running it on your own PC.

Verify your firmware

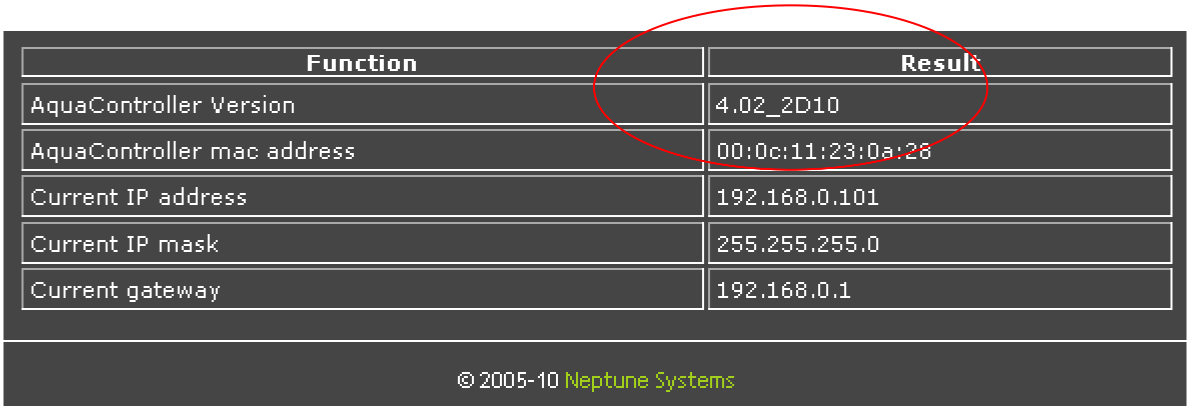
Your Apex should ship with current (not Beta) firmware. Now that you have communications working, it's easy to check. Just go to the XML page and select either the outlet report, data log report or the program report. The first line of the report will list your firmware. Compare to the chart later in this document and follow the steps to upgrade if it's not correct.



```
- <datalog software="4.02_2D10" hardware="1.0">
    <hostname>apex</hostname>
    <serial>AC4:02600</serial>
  - <record>
      <date>02/20/2010 00:00:00</date>
    - <probe>
        <name>Temp</name>
        <value>77.8</value>
      </probe>
```

You can confirm the version of your firmware that's loaded by using the Display module *Self Test* option from the main menu. It won't tell you the 4 character version (2D10 in the above example) but it will tell you the release and build (for example, 4.02, Build #2). Either way, make sure your firmware has loaded the way you think it has.

You can also view your current firmware version at the bottom of the *Configuration - Network Setup* screen



Function	Result
AquaController Version	4.02_2D10
AquaController mac address	00:0c:11:23:0a:28
Current IP address	192.168.0.101
Current IP mask	255.255.255.0
Current gateway	192.168.0.1

© 2005-10 Neptune Systems

Configure e-mail

Email can be one of the most frustrating pieces to setup. Most of the issues come from having the wrong port, userid or password declared. In addition, some of the biggest email services (Gmail, Hotmail for example) won't work with the Apex because they require SSL (Secure Socket Layers) which the Apex doesn't support. Try it first with your ISP's email server and your account information. More often than not, that will work. If it doesn't work, it could be the port or SSL (see below).

One email service that is free and doesn't require SSL is GMX (www.gmx.com) and you can use their email server to send email directly to your SSL email account like Gmail or Hotmail. You have to first establish your GMX account and get an email address. You will use this to access the GMX SMTP server. Configure your email in the Apex as below using your GMX email address as the "From" address.

Note that the following configuration options are for GMX. If you are using another email service, your options will be similar but may not be identical. For example, with AT&T you need to 'disable' email authentication. Most email services have instructions for configuring their service to work with Microsoft Outlook. You can look to that for specifics on server name, port, etc.

Email enable:	enabled
SMTP Server:	mail.gmx.com
SMTP port:	587
From:	your fully qualified GMX email address (i.e. **myname@gmx.com**)
To:	See *Note* below
Alt:	can use for text message (i.e. phonenumber@vtext.com) or leave blank
Re-email delay:	60 (this is the minutes delay the controller will use to send another email if the condition is still true, so you don't get 300 emails in a 5 minute period)
Time zone:	Your time zone, Eastern is -5, Central -6, etc.
Email Authentication:	Enabled
User name:	your fully qualified GMX email address (i.e. myname@gmx.com)
Password:	your GMX password, not your Apex password

Don't FORGET!

Note: if you want to use GMX as just a server to send email to another account like Gmail, use your fully qualified Gmail email address (i.e. **myname@gmail.com**) If you want to use GMX as your full service email provider, use your fully qualified GMX email address (i.e. **myname@gmx.com**) . In either case the "From" address must be your GMX address as is shown.

Carrier text messaging list:

phonenumber@tmomail.net	T-Mobile (US)
phonenumber@vmobl.com	Virgin
phonenumber@cingularme.com	Cingular
phonenumber@messaging.sprintpcs.com	Sprint
phonenumber@vtext.com	Verizon
phonenumber@messaging.nextel.com	Nextel
phonenumber@txt.att.net	AT&T
phonenumber@email.uscc.net	US Cellular
phonenumber@myboostmobile.com	Boost
phonenumber@message.alltel.com	Alltel
phonenumber@voicestream.net	VoiceStream
phonenumber@tms.suncom.com	SunCom
phonenumber@sms.bellsouth.com	Bell South
phonenumber@phone.cellone.net	Cellular One
phonenumber@qwestmp.com	Qwest

phonenumber@pcs.rogers.com	Rogers (CN)
phonenumber@msg.telus.com	Telus (CN)
phonenumber@orange.net	Orange (UK)
phonenumber@o2.co.uk	O2 (UK)
phonenumber@t-mobile.uk.net	T-Mobiile (UK)

Other email services will be set in a similar fashion. Be careful of the port. Some email services do not use the default SMTP port of 25 due to spam. You can find their settings on their support page or with a little bit of searching on the web. Along the same line, some ISP's do not permit you to use port 25 except with their own email service. You can research your ISP's policy on port 25 and see if the email services you want to use support inbound ports other than 25.

Email services known to work with the Apex controller:

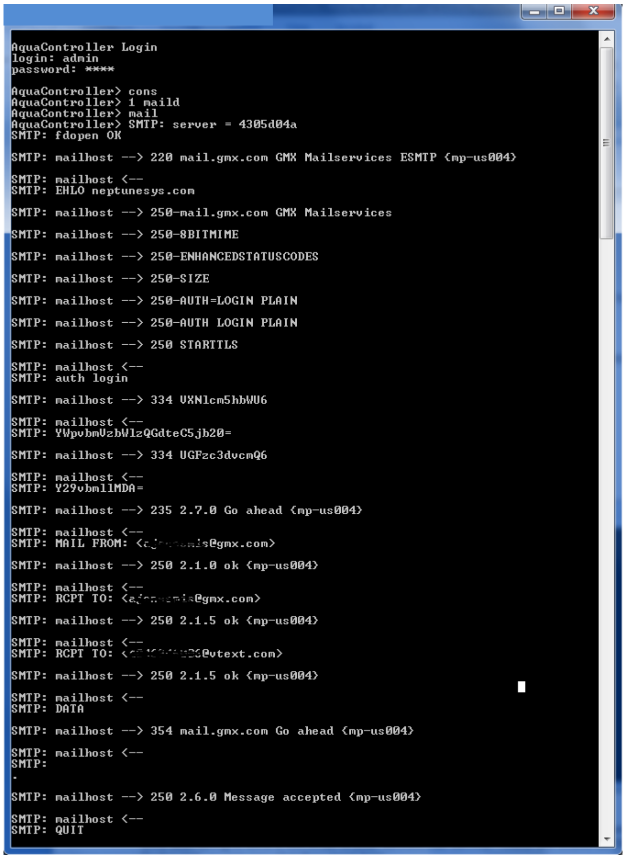
Service	SMTP Server	Port
GMX (free)	mail.gmx.com	587
AT&T – Yahoo	smtp.att.yahoo.com	25 or 465
Verizon	outgoing.verizon.net	587
Knology	smtp.knology.net	25
Gawab (free)	smtp.gawab.com	587
Cox	smtp.cox.net	25
AOL	smtp.aol.com	587

To test your email you have a couple of choices. There's a "test" checkbox at the bottom of the network/email setup screen. You can also go to the status screen and change the status for the email outlet from "auto" to "manual on" and update (remember to change it back). And of course, it can be activated through a program statement on the email outlet, something like "If time 01:00 to 23:00 then ON" (obviously that's not what you want to keep but it will do the job for testing).

The Apex has an email tracing/logging tool you can use if you're having trouble. The steps are as follows:

1. From the DOS, telnet into the controller and login
2. Issue the following commands:
    a. cons
    b. 1 maild
    c. mail

The commands and output will look something like this. Note at the bottom the "message accepted" statement indicating a successful completion.

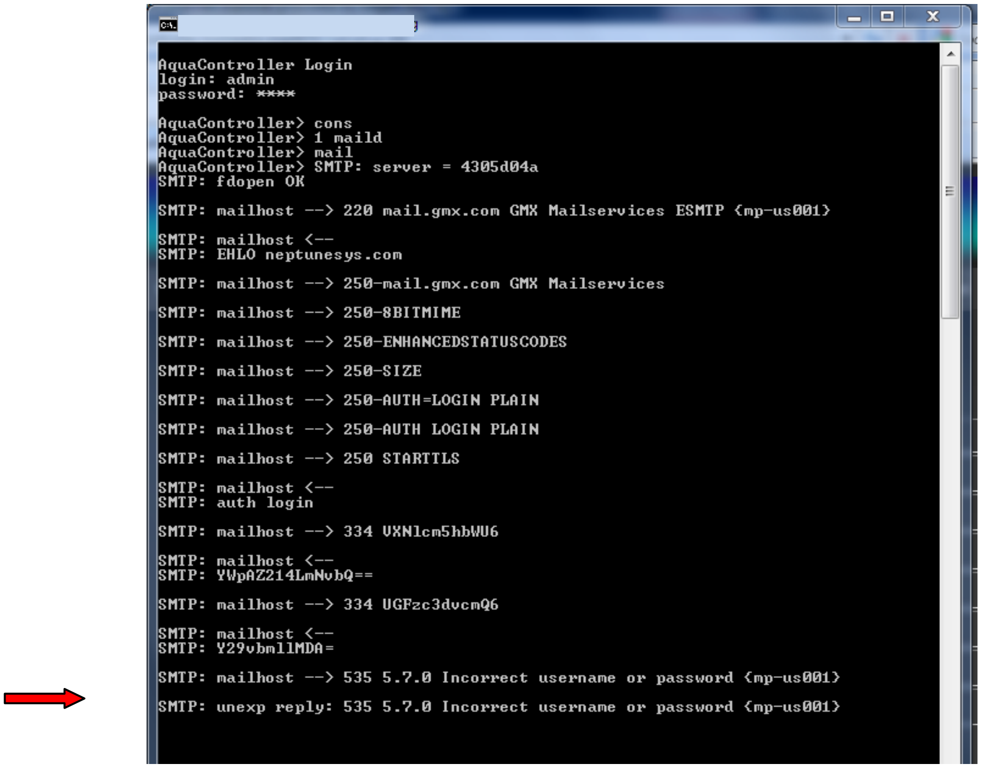


```
AquaController Login
login: admin
password: ****

AquaController> cons
AquaController> 1 maild
AquaController> mail
AquaController> SMTP: server = 4305d04a
SMTP: fdopen OK

SMTP: mailhost --> 220 mail.gmx.com GMX Mailservices ESMTP {mp-us004}

SMTP: mailhost <--
SMTP: EHLO neptunesys.com

SMTP: mailhost --> 250-mail.gmx.com GMX Mailservices

SMTP: mailhost --> 250-8BITMIME

SMTP: mailhost --> 250-ENHANCEDSTATUSCODES

SMTP: mailhost --> 250-SIZE

SMTP: mailhost --> 250-AUTH=LOGIN PLAIN

SMTP: mailhost --> 250-AUTH LOGIN PLAIN

SMTP: mailhost --> 250 STARTTLS

SMTP: mailhost <--
SMTP: auth login

SMTP: mailhost --> 334 VXNlcm5hbWU6

SMTP: mailhost <--
SMTP: YWpvbmVzbWlzQGdteC5jb20=

SMTP: mailhost --> 334 UGFzc3dvcmQ6

SMTP: mailhost <--
SMTP: Y29vbmllMDA=

SMTP: mailhost --> 235 2.7.0 Go ahead {mp-us004}

SMTP: mailhost <--
SMTP: MAIL FROM: <[redacted]@gmx.com>

SMTP: mailhost --> 250 2.1.0 ok {mp-us004}

SMTP: mailhost <--
SMTP: RCPT TO: <[redacted]@gmx.com>

SMTP: mailhost --> 250 2.1.5 ok {mp-us004}

SMTP: mailhost <--
SMTP: RCPT TO: <[redacted]@vtext.com>

SMTP: mailhost --> 250 2.1.5 ok {mp-us004}

SMTP: mailhost <--
SMTP: DATA

SMTP: mailhost --> 354 mail.gmx.com Go ahead {mp-us004}

SMTP: mailhost <--
SMTP:
.

SMTP: mailhost --> 250 2.6.0 Message accepted {mp-us004}

SMTP: mailhost <--
SMTP: QUIT
```

In the telnet window you will see a log of all the commands and responses the Apex issued and received while attempting to send a test email to your defined mail service. These are known as SMTP Response Codes and you can see what they mean here:

http://email.about.com/cs/standards/a/smtp_error_code_2.htm .

In the example below, I've deliberately entered an incorrect email address in my email configuration screen. Note the error message at the end.



```
AquaController Login
login: admin
password: ****

AquaController> cons
AquaController> 1 maild
AquaController> mail
AquaController> SMTP: server = 4305d04a
SMTP: fdopen OK

SMTP: mailhost --> 220 mail.gmx.com GMX Mailservices ESMTP {mp-us001}

SMTP: mailhost <--
SMTP: EHLO neptunesys.com

SMTP: mailhost --> 250-mail.gmx.com GMX Mailservices

SMTP: mailhost --> 250-8BITMIME

SMTP: mailhost --> 250-ENHANCEDSTATUSCODES

SMTP: mailhost --> 250-SIZE

SMTP: mailhost --> 250-AUTH=LOGIN PLAIN

SMTP: mailhost --> 250-AUTH LOGIN PLAIN

SMTP: mailhost --> 250 STARTTLS

SMTP: mailhost <--
SMTP: auth login

SMTP: mailhost --> 334 VXNlcm5hbWU6

SMTP: mailhost <--
SMTP: YWpAZ214LmNvbQ==

SMTP: mailhost --> 334 UGFzc3dvcmQ6

SMTP: mailhost <--
SMTP: Y29vbmllMDA=

SMTP: mailhost --> 535 5.7.0 Incorrect username or password {mp-us001}

SMTP: unexp reply: 535 5.7.0 Incorrect username or password {mp-us001}
```

You can copy/paste this (right click in the DOS command window, click on "select all", then right click again and this time select "copy") into a message to Neptune support (support@neptunesys.com) if you can't figure out what the error is but it's usually pretty obvious.

Configure Display

You don't need a PC to manage the Apex, in fact everything you can do with the PC you can do directly on the display and there are certain functions that are ONLY available on the display like setting feed interval times or calibrating probes for example (a complete list of the Display module's menu items including those exclusive to the Display module can be found in the Menu Hierarchy located in the Appendix). The Display module is an AquaBus module which means it can be connected to any AquaBus port anywhere on the AquaBus chain. In fact you can have multiple Display modules connected on the bus and use them to display probes monitoring different tanks or use them as multiple displays monitoring a single tank. Typically it's connected to the Base module's AquaBus port but you could connect it to an EB8 module just as well. That in itself is a significant design advantage over other controllers on the market.

Looking at the picture to the right, the Display module has 7 keys. The large round center key is the SELECT key. The 4 arrow looking keys are for navigation. The remaining 2 keys are function keys – their functions are context sensitive based on the transaction. Valid functions like 'EXIT' or 'HOME' will be displayed above them. Holding a button for ¾ second will cause it to repeat – hold it longer and the repeat rate will increase.

The display module has the capability to display 4 different status screens that you can configure. This enables you to use the Apex on multiple tanks where each one has its own screen or a single tank with up to 4 screens showing different data (i.e. virtual outlets on one screen, lights on another). You decide how you want it. You access the different screens with the left/right arrow buttons on the display module. In the top right of each screen is a series of 4 small blocks – each represents the respective numbered screen.

From the HOME screen, pressing either the UP or the DOWN arrow keys will activate one of the four feed modes. Each press of the UP/DOWN key will toggle through the feed cycles, A – D. This frequently confuses people because this sequence is different than the other menu driven commands in that there is no menu – you immediately are presented with the feed cycle choices and you don't have to select an option – simply stopping on the appropriate letter is sufficient to activate it. If you get here by mistake, simply let it select a feed cycle then immediately press the CANCEL key (right function key).

There are 6 lines of information available per screen and you can configure this in a variety of ways using either the display itself or via the browser application (*configuration →display setup*). The rows of information are:

- Clock line
- Probe line 1
- Probe line 2
- Blank line
- Status line 1
- Status line 2

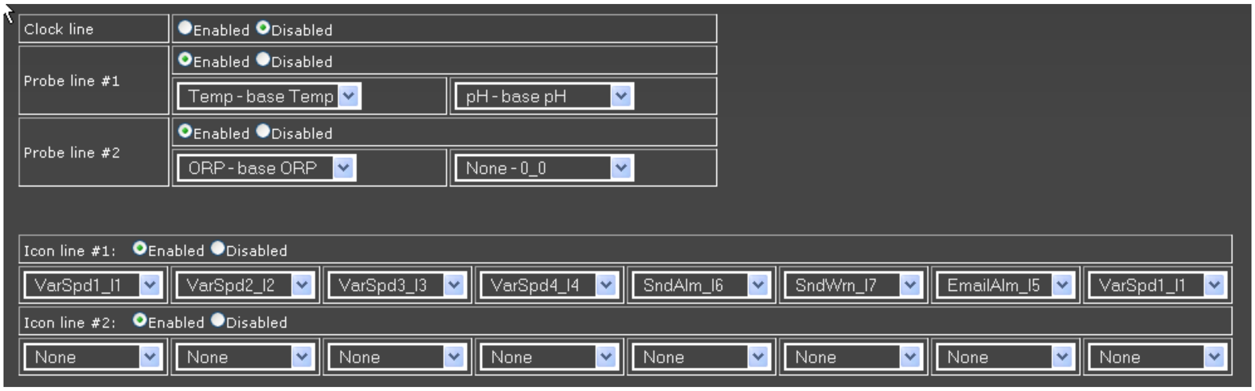
The clock line displays the date and time. The probe lines display up to 4 probe values, 2 per line. This can be used for any combination of temperature, pH, ORP, Conductivity/Salinity, Dissolved Oxygen or Amp (EB8) probes.

The status lines are a series of 8 icons each that represent your outlets – enough that each status line can handle an EB8. In the program setup for each outlet you can define an icon to represent that outlet. Each icon has a on and off version and it is these icons that are displayed in the status lines. You assign an icon to an outlet in the outlet's programming screen. The words 'ON' or 'OFF' will be displayed if you have manually turned that outlet on or off – the icons indicate the outlets on/off status when in AUTO. So, you have three possible things that can be displayed for an outlet – the icon in either its on or off version, the word 'ON' or the word 'OFF'.

You don't have to use all lines. For example, if you only use a single status line, your probe line 1 will be in large font. If you don't use either status line then both probe lines will be in large font. And if you use all lines then all are in small font.

One complaint users had was not being able to get rid of outlets you don't use and clean up the status display. Beginning in version 4.02 the display configuration you select will carry over to the web Status page. You now can select "all" status or pages 1 – 4. The devices you select in your configuration to display as icons on the display module screens will have their status (Auto, Manual On, Manual Off) represented on the corresponding web page.

Selecting the 3 probes, 4 variable speed ports, 2 alarms and email:



Clock line	○Enabled ●Disabled	
Probe line #1	●Enabled ○Disabled	
	Temp - base Temp	pH - base pH
Probe line #2	●Enabled ○Disabled	
	ORP - base ORP	None - 0_0

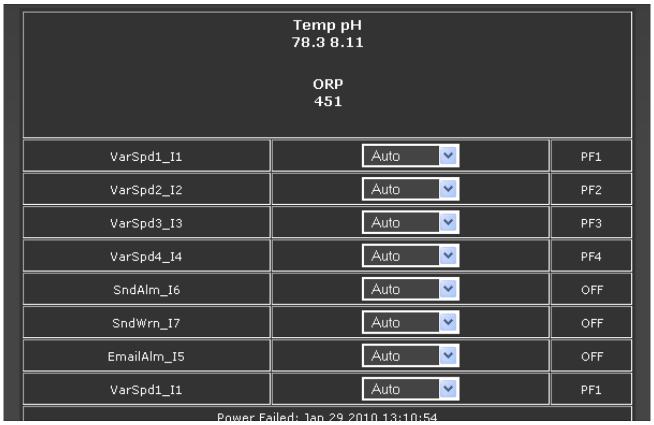
Icon line #1: ●Enabled ○Disabled

VarSpd1_I1	VarSpd2_I2	VarSpd3_I3	VarSpd4_I4	SndAlm_I6	SndWrn_I7	EmailAlm_I5	VarSpd1_I1

Icon line #2: ●Enabled ○Disabled

None	None	None	None	None	None	None	None

Will result in a status page that looks like this:



Temp pH
78.3 8.11

ORP
451

VarSpd1_I1	Auto	PF1
VarSpd2_I2	Auto	PF2
VarSpd3_I3	Auto	PF3
VarSpd4_I4	Auto	PF4
SndAlm_I6	Auto	OFF
SndWrn_I7	Auto	OFF
EmailAlm_I5	Auto	OFF
VarSpd1_I1	Auto	PF1

Power Failed: Jan 29 2010 13:10:54

No unused ports are listed making for a very clean display.

Given all this, you could build a single Apex system that monitors and controls multiple tanks. You would run an AquaBus between all the tanks, locating the base unit in some convenient location and keeping the total length of the bus to under 200'. Each display can monitor that tank so practically speaking, you could manage 4 tanks if you wanted to devote a display page to each. Each page can monitor up to 4 probes and up to 16 outlets. Of course, you can physically have more probes and outlets than that but the display can only represent that many.

Display Switch Status

Currently you cannot display the status of switches. To get around this limitation, create a virtual outlet specifically for the switch you want to monitor the status of. Name it something appropriate and program it such that 'If Switch1 OPEN Then ON'. Then display the status of that outlet on the status screen.

Configure Internet access

OK, so you've now got your controller working off your in-house network and you're wondering "Can I get this thing to work across the web?". Well of course you can - that's one of the reasons you bought the Apex. Before you do though, there's a couple of things you have to know about networks, private and public. And if you already know this, skip on down to "Putting it all together".

Configuring Internet access is where most people get stuck. Stop for a moment and think about what you are doing. You're basically configuring a computer to work as a web server – except it's not a PC and you're doing this on a residential network. But if you decided to build a web server in your home, you would need the following components:

- A server or servers
- Web server software
- Dedicated IP addresses
- Build your own web pages
- Supplemental software like Java and Flash
- Security software

With the Apex, here's what you need:

- The Apex controller

The Apex is amazingly robust, coming complete with all the software you need. It has its own web server software, the web pages are already loaded, it has security (OK, maybe a little light there) and it has the Java and Flash that you require. All you have to do is get the IP addressing working and since you're doing this on a residential network, there are a couple of little twists you have to get around.

Still game? Read on.

Static vs. Dynamic IP addresses

Your Apex can have a dynamic IP address, in fact by default from the factory it has dynamic addressing enabled (DHCP = ON). This means every time you plug in the base module or reboot it, it will ask your router for an address. If your PC and Apex are both doing this, you can't be sure what address each will get. Most of the time you really don't care and don't realize that your PC may come up with a different address than it had yesterday. But since we're going to be directing traffic from the Internet to the Apex, we have to know what the Apex's IP address is. To do this, we assign an address and turn off DHCP in the Apex so it will only use this address. So for example, suppose your router on your home network uses addresses 192.168.0.100 to 199. To be safe assign it an IP address out of that DHCP range – give the AquaController an IP address of 50. The router will never give that to any other device and every time the Apex attaches to the network it will use address 192.168.0.50.

Most home networks are built around what are known as private IP address schemes. Typically they will use something like 192.168.0.100 through 199. Your third and fourth octets may vary (3-Com and Linksys routers use "1" for the third octet, D-Link and Netgear use "0") but if you have a PC your first two are probably 192.168. These are private IP address ranges but are not usable (or routable) over the Internet. They work fine for your home but to operate over the Internet, you need a range that's unique. Your cable or DSL provider supplies this with their modems. This is done by "proxy". They will have a public (or WAN wide-area network) address on their side of the modem, you have your private (or LAN local-area network) address on the other (to find out what your current external WAN IP addres is go to www.whatismyip.com) . The twist is that your cable or DSL provider doesn't want you to tie up that address forever so it expires every so often (its dynamic and called a 'lease'). As soon as you go to use your connection, your modem will ask your provider for a new address which your modem will keep for some time until it too expires.

So as you can see, you can have static or dynamic addresses on your Apex as well as your cable or DSL modem. This dynamic IP addressing causes a problem. How do you access your network from the Internet if the address is changing all the time?

Domain Name Services

To deal with dynamic IP addresses on the external (WAN) side and simplify things a little, a number of Domain Name Services exist. These services will keep track of your modem's constantly changing public IP address for you and give you a static name you can use to access your home network. The name will never change but the IP address it points to will. One of the best known of these services is DynDNS (www.dyndns.com). It's a free service that's very easy to setup. Pick a name for your router, input your modem's current public IP address and they take it from there. A small application that you download to your PC from DynDNS constantly monitors your WAN IP address and updates the DynDNS server. You never have to think about it.

Many routers support dynamic DNS (DDNS) internally – there's no software required to be installed on your PC making this solution far more reliable. When you enable DDNS support in your router, you specify the domain service to use (sometimes you're limited to just those your router vendor includes). Your router then takes care of determining when the WAN IP address changes and notifies the domain name service. If your router supports DDNS, that should be your first choice.

Other DDNS services are (DynDNS is still probably the best):

- changeip.com
- DNSomatic.com
- easyDNS.com
- EuroDynDNS.org
- no-ip.com
- ods.org
- ovh.com
- regfish.com
- tzo.com
- zoneedit.com

Ports

Another network issue you'll have to deal with is the issue of ports. A port is basically a way for you to get into an application from the Internet. Ports are to programs as IP addresses are to computers. A computer has one IP address but each program running has its own port (or ports). A port can only belong to one program at a time but a program can use multiple ports. Many applications have historically used specific ports to access their services. For example, most internet browser traffic uses port 80, telnet uses port 23, email uses 25, etc. The problem is that hackers long ago figured this out and look for these ports to exploit their dirty deeds. As a result, many Internet providers now block specific ports, forcing you to use something different. If your ISP is blocking port 80, read on.

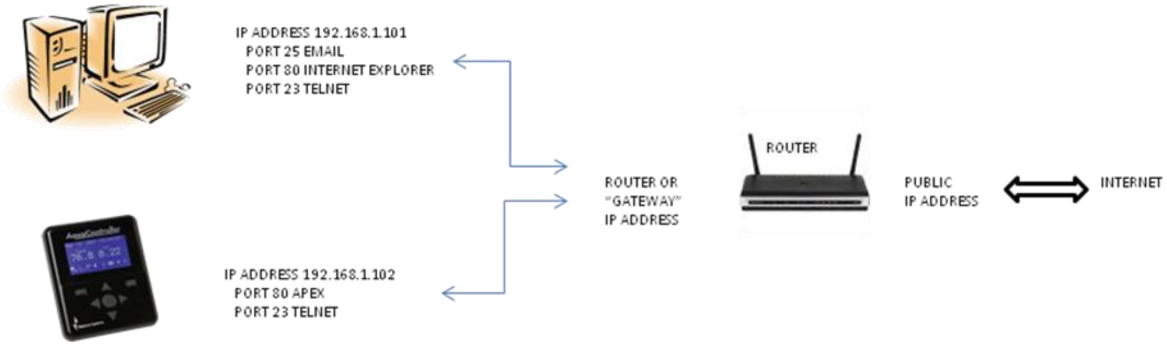
Fortunately for us, the Apex controller deals with this. By default it uses port 80. You may have noticed this in the network setup screen. However, you can use any port number you want (as long as it's not being used by something else on your network). You just have to remember that port number and use it in the Apex, your router and with your Domain Name Service. All three must match. Since the port number is 4 digits, you have plenty of choices. Tip: Since setting the port address on the display controller requires scrolling to the port number, using a high number like '9975' enables you to scroll down past '1' and into the 9000's. Of course, setting the port via the web interface 'configure network' has no such issue since you can directly type in the number. So, first try port 80 but if it's blocked then you'll have to assign a new port to the Apex and complete the port forwarding configuration process.

If you change the port number of your controller and use the http://apex command to access it by hostname, you must add the new port number to the end. So, if your new port number was '9000' then you would use http://apex:9000 to access the controller by name.

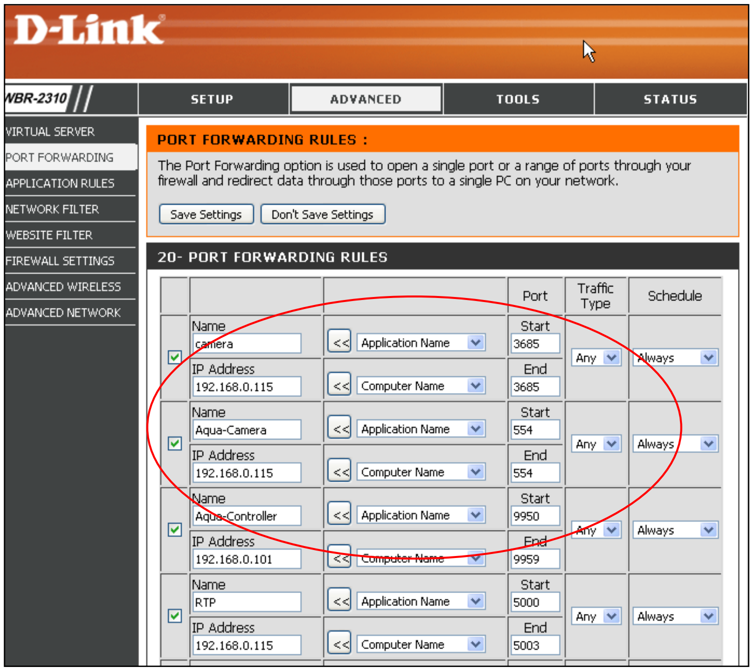
Port Forwarding

Your router is tasked with taking traffic to/from the internet and "routing" it to the appropriate device or application on your network. By default, the Apex uses port 80 so you would configure your router to forward port 80 messages to the Apex's IP address. If port 80 is not being blocked by your ISP, then you simply use this default. However, if you have to change your port, you're going to have to configure your router to push particular traffic to the IP address and port of your Apex controller, whatever you had set that to. This process is called "port forwarding" whether you're using the default port 80 or a custom port number. Basically you're telling the router "take any traffic that comes in looking for port 1234 and send it to IP address 192.168.0.105, port 1234 (or whatever the address of your Apex is). A good tutorial and screen shots of popular routers can be found at www.portforward.com.

You can visualize IP addresses, ports, routing and gateway like this:



Here’s a sample from a D-Link router’s port forwarding configuration screen.



PORT FORWARDING RULES :

The Port Forwarding option is used to open a single port or a range of ports through your firewall and redirect data through those ports to a single PC on your network.

20- PORT FORWARDING RULES

			Port	Traffic Type	Schedule
☑	Name camera	<< Application Name	Start 3685	Any	Always
	IP Address 192.168.0.115	<< Computer Name	End 3685		
☑	Name Aqua-Camera	<< Application Name	Start 554	Any	Always
	IP Address 192.168.0.115	<< Computer Name	End 554		
☑	Name Aqua-Controller	<< Application Name	Start 9950	Any	Always
	IP Address 192.168.0.101	<< Computer Name	End 9959		
☑	Name RTP	<< Application Name	Start 5000	Any	Always
	IP Address 192.168.0.115	<< Computer Name	End 5003		

Example shows a D-Link WBR-2310 screen. Other routers will be similar.

Verizon FIOS

Verizon’s FIOS (Gaelic for “knowledge”) is an “all-in-one” bundle of services – telephone, internet and TV. As such, it configures like a router, a firewall and a content-filtering device. HenryH has written an excellent tutorial on how to configure FIOS to work with the Apex.

You can find it at ReefCentral in the Neptune forum (thread # 1760163):

Putting it all together

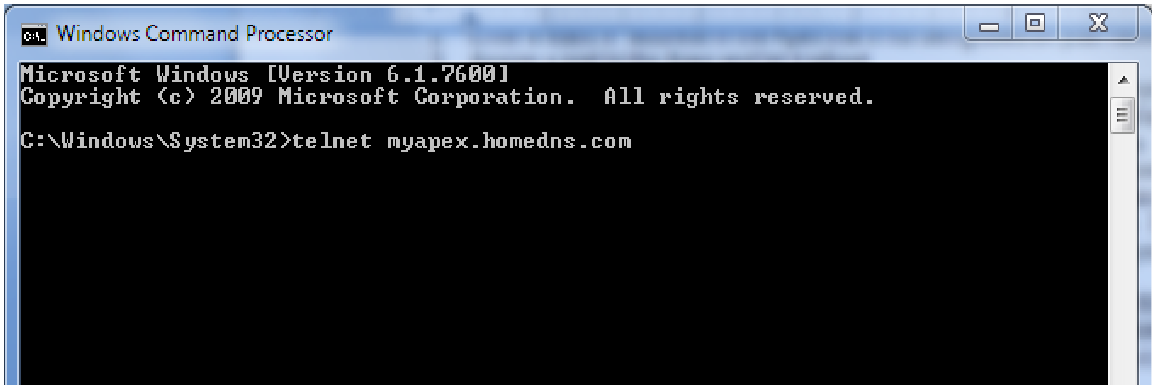
If you're still game, here's how to put it all together:

1. Turn off DHCP on the Apex
2. Enter a static IP address in the Apex that's not being used on your network
3. Assign a port to the Apex and let it reboot.
4. Make sure you can still access the controller. You now access it from your browser with this syntax: http://192.168.0.110:8000 (assumes address 110 was picked and port 8000).
5. You can? Good. Move on. If not, stop and recheck what you’ve done. If you think you’ve entered everything correctly, reboot your router (first) then base unit (second).
6. Setup an account with DynDNS (or whomever you select). Pick a hostname, register and enter your router’s external IP address. Download and install the updater software if you’re using DynDNS. DynDNS will track this from now on and change it as necessary. You will just use the name you selected and could care less about the IP address after this.
7. Log into your router and find the menu for port forwarding. Enter port 8000 (or whatever you selected as your port in #3), select "all traffic" and input the now static IP address of your Apex

from #2, 192.168.0.110. Sometimes there's a check box to activate/deactivate it. Make sure the box is checked.

8. Now go to a PC that's not on your home network (time to call a friend you haven't talked to in years). Have them try accessing your Apex via the hostname you created in step #6. Remember to include the port from step #3. So if your DynDNS hostname was "myapex.homedns.com" and you had selected a port number of 8000 you would enter "http://myapex.homedns.com:8000" in the browser address bar. You should get a small window with the Apex login prompt. Enter your Apex userid and password and voila! You should be in.
9. You will want to change the userid and password on the Apex after this to be safe. Remember, your Apex is now accessible from the Internet.
10. If you're successful to this point you might want to repeat step 7 for telnet access to your controller. This way Neptune can get into your system for remote support. Simply add another port forwarding statement in your router but this time the port must be "23". Point it to the exact same IP address, that of your Apex controller.

The test of this is a little different. Still must be done from a different computer but this time instead of testing with Internet Explorer, your friend must start the DOS command window. From the command prompt they would enter (without "www" or a port number):



```
Microsoft Windows [Version 6.1.7600]
Copyright (c) 2009 Microsoft Corporation.  All rights reserved.

C:\Windows\System32>telnet myapex.homedns.com
```

You should receive the AquaController 'login' prompt. You would login as normal. Once you have tested this, you need to decide if you want to keep it active or not. Typically it's just a check box in the router – you would leave all the information in place. As Russ Michel pointed out to me, telnet is one of the oldest and most basic protocols – low hanging fruit for nefarious individuals. A four digit password is easily susceptible to a brute force attack. It's not like you have important financial information in your Apex but they could screw it up just to be nasty.

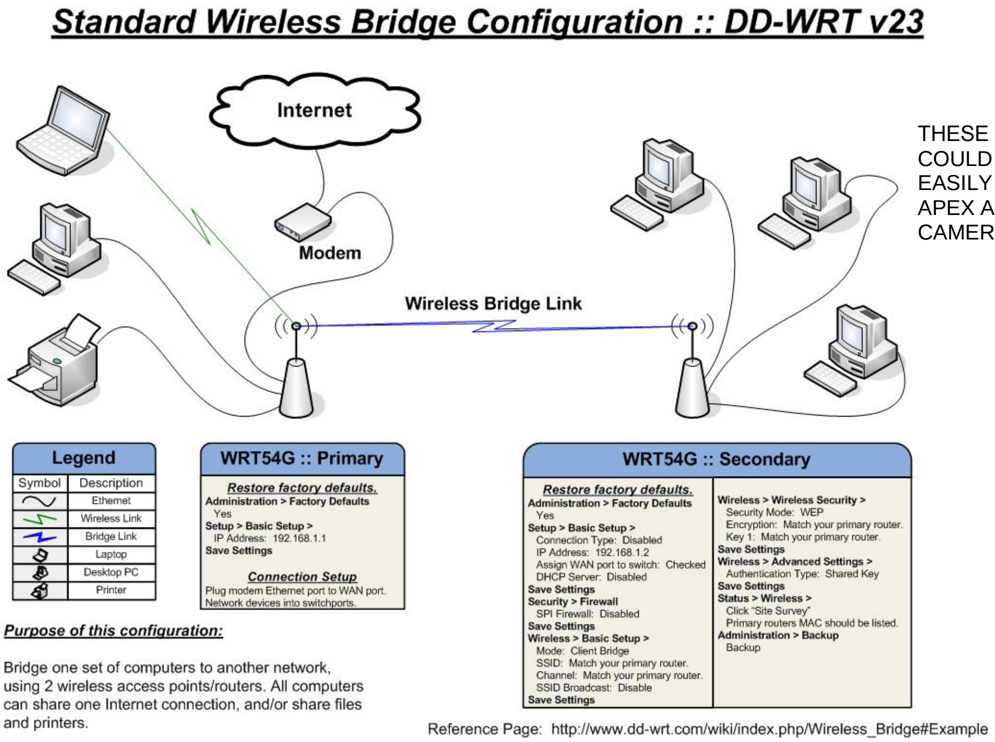
Chapter 3: Wireless

Overview

The Apex does not have wireless built-in. You can spend a lot of time debating whether it should or should not but given various design and market constraints Neptune decided not to incorporate wireless. However, you can add wireless capability to the Apex controller and you have a number of choices of how to do it. <u>It's important to note however that currently firmware updates cannot be applied via wireless and any attempt to do so is not supported. You can turn your Apex into a brick</u>. So, you've been warned!

Most home networks have either DSL or cable broadband that comes into the house and terminates into a wireless router. These routers also have a number of LAN connections on the back where you can physically connect a PC or a printer via Ethernet. A computer running Windows or Mac O/S can easily connect to this type of a network with a wireless adapter either external as with USB or internal via a dedicated network card or on the motherboard. But the Apex does not have a wireless NIC and there's no way to add one, internally or externally like a PC.

The solution is to <u>extend</u> your existing wireless network by 'bridging'. A bridge is a method of connecting together two network segments. In our case this "second" network is a network of a single device – the Apex. The easiest way to bridge the Apex to your network is with gaming adapters. These were originally developed to allow people to connect their Xbox and other game consoles to a wireless network – that's why they are still called gaming adapters today. A wireless gaming adapter has a single Ethernet jack and a wireless network interface. After it is configured, the wireless part of the gaming adapter will connect to your wireless router, just like a laptop or other wireless client. It is an inexpensive solution to the problem of not having a wireless NIC card in the Apex – most gaming adapters retail for under $100.



THESE THREE PC'S COULD JUST AS EASILY BE YOUR APEX AND TWO CAMERAS

Legend

Symbol	Description
∼	Ethernet
⚡	Wireless Link
⚡	Bridge Link
💻	Laptop
🖥	Desktop PC
🖨	Printer

WRT54G :: Primary

<u>Restore factory defaults.</u>
Administration > Factory Defaults
Yes
Setup > Basic Setup >
IP Address: 192.168.1.1
Save Settings

<u>Connection Setup</u>
Plug modem Ethernet port to WAN port.
Network devices into switchports.

WRT54G :: Secondary

<u>Restore factory defaults.</u>
Administration > Factory Defaults
Yes
Setup > Basic Setup >
Connection Type: Disabled
IP Address: 192.168.1.2
Assign WAN port to switch: Checked
DHCP Server: Disabled
Save Settings
Security > Firewall
SPI Firewall: Disabled
Save Settings
Wireless > Basic Setup >
Mode: Client Bridge
SSID: Match your primary router.
Channel: Match your primary router.
SSID Broadcast: Disable
Save Settings

Wireless > Wireless Security >
Security Mode: WEP
Encryption: Match your primary router.
Key 1: Match your primary router.
Save Settings
Wireless > Advanced Settings >
Authentication Type: Shared Key
Save Settings
Status > Wireless >
Click "Site Survey"
Primary routers MAC should be listed.
Administration > Backup
Backup

<u>Purpose of this configuration:</u>

Bridge one set of computers to another network, using 2 wireless access points/routers. All computers can share one Internet connection, and/or share files and printers.

Reference Page: http://www.dd-wrt.com/wiki/index.php/Wireless_Bridge#Example

Image from dd-wrt.com

Wireless Connection Options

The following is a list of ways you can connect the Apex to your home network (in order of preference):

1. Gaming Adapter (examples: Linksys WGA54G, TrendNet TEW-647GA*).
2. Wireless Bridge Adapter also known as Wireless Ethernet Bridge (examples: D-Link DWL-G810, Linksys WET54G). Where the gaming adapter will only give you a single Ethernet jack, a wireless bridge adapter will give you multiple Ethernet jacks.
3. Using open source firmware (i.e. DD-WRT) to convert a router into a dedicated bridge (if you insist, you'll impress your local Geek Squad guys but be a real bummer at parties). You can get more info on DD-WRT and a list of compatible routers at www.dd-wrt.com . As with the wireless bridge adapters, this will give you multiple Ethernet jacks.
4. Powerline Ethernet – note that you cannot put these on a UPS for power failure protection. Consider carefully whether that risk is worth their easy-to-connect capability.

*- There are some reports that the TEW-647 has issues with losing communications with the Apex base unit. The causes could be DHCP, QoS options and/or reception.

What won't work is:

1. Wireless Travel Routers
2. USB Wireless Adapters
3. Wireless Access Points
4. Switches and Hubs
5. Another Router
6. Two tin cans and a string

Wireless Configuration

Actual step by step configuration instructions for all these possible alternatives is beyond the scope of this guide. There is however common tasks that you must accomplish in order for one of these devices to work and we can detail them here.

1. Make sure your Apex is working in a hardwired, Ethernet mode. You can take the base and display unit (you don't need the EB8) and place them next to your existing wireless router and configure/test them. Only when you're certain you have hardwired network connectivity to the Apex should you attempt wireless.
2. Record your important network parameters. The first three will come from doing an ipconfig /all from the DOS and will probably be the address of your router. Your wireless security would be found in your router's configuration.
   a. Gateway address
   b. Subnet mask
   c. DNS server
   d. Record any wireless security key or passphrase
   e. Record your wireless router's SSID
3. Plug your bridge device into your current router to configure it, using an Ethernet cable.
4. Follow the vendor's directions for configuration. Typical steps will include:
   a. Configure the adapter to use the same SSID (wireless network name) as your wireless router.
   b. Configure wireless security matching the WEP, WPA, WPA2 key of your router – this is a common problem area. If you have trouble getting things working, try disabling wireless security on your primary router and your bridge (gaming adapter).
   c. Setting a unique IP address that matches your router's subnet (first 3 octets the same)
   d. Input the gateway address and subnet mask
   e. Letting the bridge 'find' available networks and connect
5. Once you get this far you can take your bridge device / gaming adapter to your Apex and connect it.

Chapter 4: Programming

Introduction to Outlet programming

One of the most intimidating aspects of the Apex is getting your first program going. What you need to remember is that fundamentally the Apex programming language is just a bunch of simple true/false statements that result in an outlet being either ON or OFF at the end.

With the Apex, statements are evaluated by outlet (called "timers" in other Neptune controllers), first statement to last statement. This is done once per second for any outlet that is not set to Manual ON or Manual OFF. As statements are processed the controller is updating an internal register for that outlet. The register can change from ON to OFF for every command but whatever the outlet register's state is after all program statements are evaluated, that is what the Apex sets the outlet physically to. So the order of conditional statements is important. A true OFF statement followed by a true ON statement will result in an "ON" condition. Move the OFF statement to the end and the mode is reversed. Having this internal register for each outlet is going to be important as you will see with some of the more advanced program statements. Non-conditional statements like 'Defer' and 'Min Time' are not affected by where they are in the outlet program.

Take these examples where the statements are identical but the order is reversed; assume the time is 1pm (13:00) and the tank temperature is 81 degrees:

If Time 08:00 to 19:00 Then ON	If Temp > 80.5 Then OFF
If Temp > 80.5 Then OFF	If Time 08:00 to 19:00 Then ON
This outlet would be OFF	This outlet would be ON

The Apex program statements have two parts. The first part establishes the true/false condition and starts with "IF" (except Oscillate, Min Time and Defer where no "IF" is required – more on these later). Examples are "If Time", "If Outlet", "If DoW", "If Temp", "If pH" and so on. This first part gets evaluated as either true or false based on an operand and an operator, an "ON" or "OFF" or an implied "true" or "false" depending on the statement.

```
If pH >  8.30
If Temp  > 84
If Time  08:00 to 19:00
If DoW  SM-WT-S
If Outlet MH_Right = ON
If Sun 020/040
If Power EB8_3 ON
If Switch1 OPEN
```

These each evaluate to True or False

The second part of every statement begins with "Then" and is followed by either "ON" or "OFF" (caps are for readability and are not required for statement execution). The "Then" action is only taken if the statement evaluates to "true".

```
Then ON
Then OFF
Then [PROFILE NAME]
```

When you assemble the pieces you have the complete program statements

```
If pH > 8.30 Then ON
If Temp > 84 Then OFF
If Time 19:00 to 08:00 Then Night_Mode
```

Some variables you can test for in your program statements:

- Time of Day
- Temperature
- pH (requires pH probe)
- ORP (requires ORP probe)
- Switch values (requires I/O box)
- Day of Week
- Power (on/off) for either the base module or the EB8
- Seasonal Temperature
- Sunrise/Sunset
- Lunar Cycle
- Another outlet's (physical or virtual) on/off state
- How long has this outlet been in this state

Know which EB8 is which

If you have more than one EB8 declared in your base module, you want to make sure you know which one is which. A simple way to do this is to get a regular table lamp and turn it on. Plug it into the first outlet on an EB8. Now, go to the status screen and find that outlet. It will be numbered something like "name_x_1" where name is the outlet name established by the base module, x is the EB8 number and 1 is outlet #1. At this point, you don't care if it's a pump, a light or a heater. It doesn't matter. From the status screen, set that outlet state from auto to manual on. The light should turn on. Then set the outlet state to manual off. The light should go out. If it doesn't try the same test on the other EB8. You will quickly figure out your EB8 numbering scheme. Now you can program.

The 'Fallback' and 'Set' Statements

Almost every physical outlet is going to need a Fallback and a Set statement regardless of whatever program statements you might have (virtual outlets don't need a Fallback statement). These will normally be the first and second statements of the outlet program.

The Fallback statement establishes what state the outlet should be in if the EB8 loses communication with the base module, which is why it doesn't apply to virtual outlets. It's only for physical outlets. In case of communications failure, what do you want this outlet to do? For a return pump, you probably want the fallback to be ON since having your main pump shut down would be a bad thing. On the other hand, if your outlet controls your dosing pump, perhaps fallback OFF is more appropriate because the pump could fatally dose your tank. Many people set their lights to Fallback OFF but I prefer to have them ON in this case so I can manually control them via the light ballast's on/off switch. If you set them to Fallback OFF, then they don't get any power and you have to pull wires and find a wall outlet to plug them into while you resolve whatever controller issue you might have.

The Set statement is used to establish the starting point for this outlet's register state for this run through the program - it initializes the outlet. If you have a pump that's supposed to run 7x24 without any other conditions, then all you need is these two statements...Fallback ON and Set ON. That's it. The pump will run all the time even if the EB8 loses communications with the base module.

Be careful with the SET statement. If you're having a problem with an outlet program working the way you want, take a look at SET. Consider this simple example (from my own tank):

```
Fallback OFF
Set OFF
If Temp > 78.5 Then ON
If Temp < 78 Then OFF
```

What will the state of the outlet be if the temperature is 78.3 degrees? It will be OFF. In fact, any temperature that is lower than 78.6 (because the stmt is > 78.5, i.e. 78.6) and the outlet will be OFF. The If Temp < 78 Then OFF statement is meaningless. This is because of the Set OFF statement.

Remove that statement and what is the outlet state at 78.3? It depends on whether the temperature is coming up from 78 (in which case it would be OFF) or coming down from 78.5 in which case it would be ON.

SET is one of those little “gotcha” commands that you always have to look at, especially if something isn’t working the way you think it should. In general do not use the set statement if you use the probe values to turn on/off the outlet.

Outlet Naming

Neptune has a convention where the outlet name combines a short device name, the EB8 number and the receptacle number on the EB8. This is a handy nomenclature to use especially if you have a lot of outlets. However, you are not restricted to that. **You can use any name you want, upper and lower case (it’s case-sensitive), maximum of 12 characters, no spaces and it must be unique.** So if "Front_MH" means more to you than "MH_3_2", then that's what you can use.

Here's a sample table you can use to help organize what you want your outlets to do:

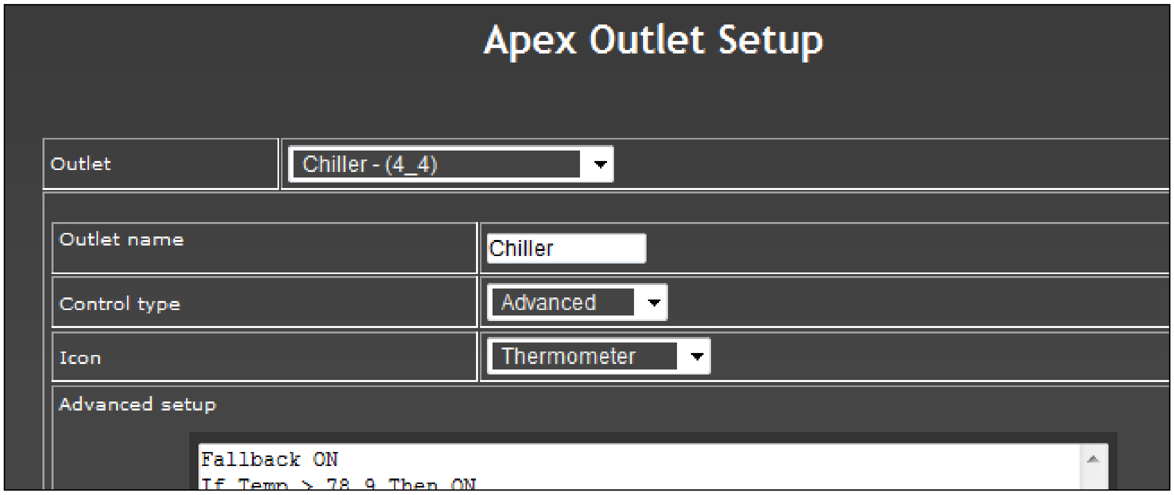
Outlet Name	EB8 Loc	Fallback	Set	ON Conditions	OFF Conditions
Heater	3_1	Off	Off	If Temp < 75.3	If Temp > 76
Return_Pump	4_5	On	On		If Water_Low* If FeedA If FeedB
Halides	4_6	Off	Off	If Time 08:00 to 19:00	If Water_Low* If Temp > 80
Email	Virtual	Off	Off	If Temp > 80 If Temp < 75 If Water_Low*	

* - In the above chart note that "Water_Low" is a virtual outlet that would have been set in its own outlet program, probably from a float switch opening or closing. You can define as many virtual outlets as you want and they can be set or tested like any physical outlet.

Outlet Programming (Setup)

The Apex web software comes with some "fill in the blanks" setup screens for programming basic device types like lights, pumps, heaters or chillers. To use these, just select the appropriate outlet type when you go into the *Configuration → Apex Outlet Setup* page. If you want to enter program statements manually, set the Control Type to "*Advanced*". You can use both methods to see what the program statements are that the Apex generates. Just use the wizard then switch the mode to advanced. You can then tweak the statements or add additional statements as you like.

There are 6 Control Types plus *Advanced*. They are: *Light, Pump, Heater, Chiller, pH Control and ORP Control*. Each has a unique group of setup statements appropriate to that device type.



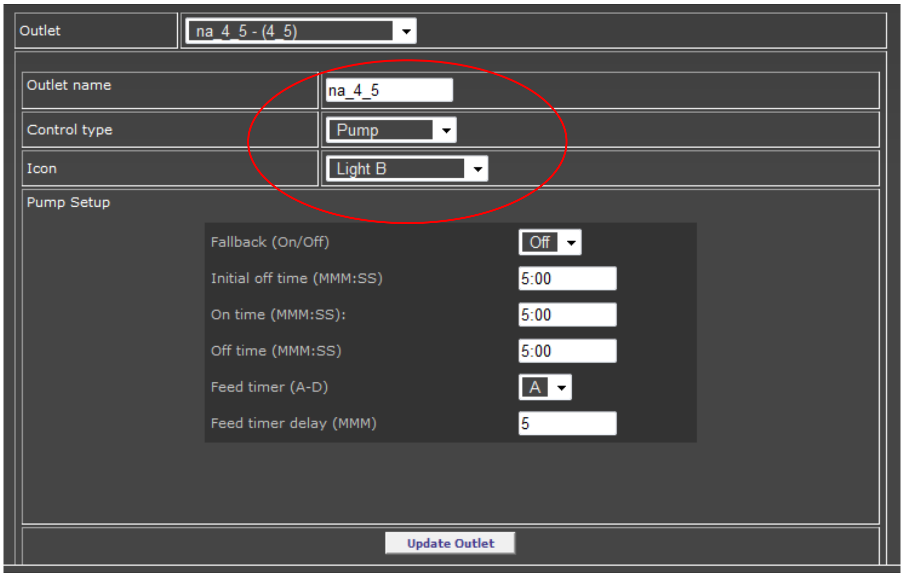
Apex Outlet Setup

Outlet	Chiller - (4_4)
Outlet name	Chiller
Control type	Advanced
Icon	Thermometer

Advanced setup

```
Fallback ON
If Temp > 78.9 Then ON
```

Here’s an example of what the setup screen looks like when using the *Pump* Control Type:

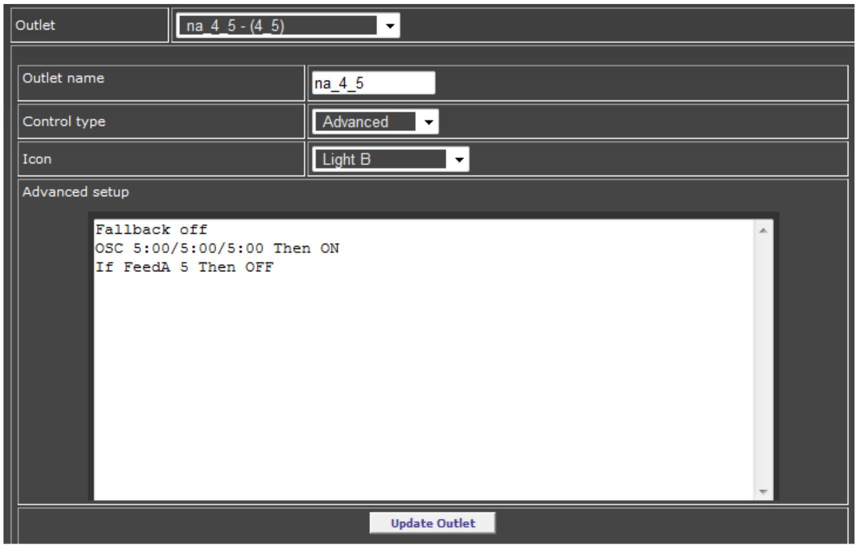


Outlet	na_4_5 - (4_5)
Outlet name	na_4_5
Control type	Pump
Icon	Light B

Pump Setup

Fallback (On/Off)	Off
Initial off time (MMM:SS)	5:00
On time (MMM:SS):	5:00
Off time (MMM:SS)	5:00
Feed timer (A-D)	A
Feed timer delay (MMM)	5

Here's the same outlet using the *Advanced* control type. Note the oscillate and feed statements have been completed for you:



Outlet: na_4_5 - (4_5)

Outlet name: na_4_5

Control type: Advanced

Icon: Light B

Advanced setup

```
Fallback off
OSC 5:00/5:00/5:00 Then ON
If FeedA 5 Then OFF
```

As you can see you can either use the menus to complete your outlet program or the advanced mode or both. You can start with the menu, update your results then return to the outlet program and add additional statements as need.

As you become comfortable with the programming of the Apex you will likely use the *Advanced* method of programming all the time. For an example of various outlet programs including some simple and complex, visit Ken's Reef at http://www.kenargo.com/apexProgram/index.php

Chapter 5: Basic Programming Statements

Time related programming variables

There are three common time programming variables that will be referenced in the next two chapters. They are summarized here for clarity and are referenced as needed in detail.

Variable	Format
TIME	HH:MM in a 24 hour format
DURATION	MMM:SS
DELAY	MMM

Time

If Time [TIME] to [TIME] Then [ON/OFF/PROFILE]

The "If Time" statement is one of the simplest to use statements. There are only two things you need to know to use it properly. First, the time is entered as HH:MM in a 24 hour format. Second, the Apex controller is smart enough to know when a time range spans midnight. If you want a time statement that runs from 9pm – 3am the next morning, you can do it with a single statement like

```
If Time 21:00 to 03:00 Then ON
```

You cannot use '<' or '>' in the Time statement – only 'to'. Also, since the time is evaluated as true if <u>between</u> X and Y, if you want something to end at precisely 3:00 am then you would enter 2:59 in your time statement otherwise it would be 3:01 before being 'false'.

Probe Testing (Temp, pH, ORP)

If [PROBE NAME] [</>] [VALUE] Then [ON/OFF/PROFILE]

The standard Apex probes have reserved programming names of Temp, pH and ORP. If you have a PM expansion box, these names get a 'Xn' suffix where n=expansion box number (for example, pH becomes pHX5 with a PM1 at address #5). Temperature value goes to tenths of a degree; pH can be tested to hundreds. ORP is a whole number. The syntax is very simple:

```
If Temp > 78.6 Then ON
If pH < 8.23 Then PF3
If ORP > 375 Then OFF
```

Testing pH to two decimal places can result in a condition known as "hysteresis" where the outlet is bouncing back and forth as the pH changes ever so slightly. Consider testing pH just to tenths (i.e. 8.20 instead of 8.23 in the example). You can use the "DEFER" command to smooth this hysteresis but I still question whether two decimal places on a logarithmic scale really matters in our aquarium environment.

Testing against other outlets (Outlet State Statements)

If Outlet [OUTLET NAME] = [ON/OFF] Then [ON/OFF/PROFILE]

The Apex programming language lets you take the status of an outlet, either virtual or physical and test it against another outlet, either virtual or physical (technically you're testing the state of one outlet in another outlet's program). Let's take a simple example of two physical outlets. One controls your lights, the other controls your fans. You want to run your fans any time your lights run. You could duplicate all of the conditions for your lights in your fan program. That would work but every time you change your lights program, you have to remember to change your fans program. Here's a simpler way. In your fans outlet program add the following. Whenever your outlet "MH_Right" is on then your fans will be on.

```
Set OFF
If Outlet MH_Right = ON Then ON
```

Note that the = sign is required with the If Outlet statement – it's the only statement that does. Also, the outlet names are case sensitive.

Day of Week

If DoW [SMTWTFS] Then [ON/OFF/PROFILE]

The Day of Week (DoW) lets you tweak your outlet program based on the day of the week. It's a little different from other statements in how it works. It has 7 placeholders, each represent a day of the week.

SMTWTFS

If you substitute any of the days with a hyphen, then that day evaluates as false

S-TW-FS	Sunday, Tuesday, Wednesday, Friday & Saturday are True, Monday and Thursday are false

<u>If you happen to use the wrong letter for a day position, a hyphen will automatically be substituted</u>.

Consider the following. In this example we want the outlet OFF on Saturday and Sunday, ON for all other days:

```
Set OFF
If DoW –MTWTF- Then ON
```

The above statement would evaluate to ON for every day except Sunday and Saturday.

Now consider this example where I want the lights in the tank to come on early on Saturday and Sunday by an hour:

```
Fallback ON
Set OFF
If Time 07:00 to 08:00 Then ON
If DoW -MTWTF- Then OFF
If Time 08:00 to 19:30 Then ON
```

The logic above is as follows (disregard the Fallback and Set). Every day, if the time is 7 – 8am then turn the register ON. But if the DoW is not Saturday or Sunday (i.e. Mon – Fri), then turn it back OFF. However, every day, if the time is 8:00 – 19:30 turn the register ON.

The order of the above statements is critical. The DoW has to be between the two time statements. Put it any other place and the statements will not evaluate as you wanted. Remember, the outlet is not turning ON/OFF during the statement evaluation – only the outlet's register is changing. Whatever the register is after the last statement is evaluated is what the outlet is physically set to.

Creating Conditionals (and, or)

You can't directly link two or more statements together to form 'and' but you can arrive at the same point if you think about how the controller processes statements and phrasing some conditions in the negative. Here's an example. Suppose you wanted the day to be Tuesday and the Time to be 4pm – 5pm in order for this outlet to be on.

```
Set OFF
If DoW –T---- Then ON
If Time 16:00 to 17:00 Then ON
```

```
Set OFF
If DoW –T---- Then ON
If Time 17:00 to 16:00 Then OFF
```

The statement on the left won't work. It's essentially an 'or' – if either the DoW is Tuesday or the Time is 16:00 to 17:00 then the outlet would be ON. The statement on the right will work however and becomes an 'and' condition. The register is set OFF, turns ON if the day is Tuesday then turns back OFF if the time is other than 16:00 – 17:00 (the 'negative' condition referenced above). Since the register at the end after all statements are processed is ON, the physical outlet is turned ON. This isn't the only way to solve the problem. Keep in mind that you want your statements to go from the 'widest' condition to the 'narrowest' condition.

Power

If POWER [POWER PROBE] [ON/OFF] [DURATION] Then [ON/OFF/PROFILE]

There are two different virtual probe values you can test to determine if there has been a power interruption, "Apex" and "EB8_x" (x = the AquaBus position). The first probe, Apex is used to monitor power to the base unit. In addition to detecting a power failure to the base, this probe will also turn 'ON' during a reboot so keep that in mind.

The duration value is used to delay the probe value from being reset for a certain amount of time after the power returns, similar to how the feed timer delay works. For example

```
If Power Apex OFF 005 Then ON
```

This statement detects if power to the base is off (or a reboot) and holds this value for 5 minutes, setting the outlet register to ON.

- The If Power [power probe] [on/off] portion is evaluated first for true/false.
- If true, the value is held for the amount of minutes in duration after power is restored. 0 means it's instant and is not held past this iteration of the program statements
- Finally the 'Then ON/OFF/PROFILE' is processed to set the outlet state register

A good example of the value of the DURATION can be seen with a skimmer. If the power fails you want to delay starting the skimmer once power is restored. This can be accomplished easily with the command:

```
If Power EB8_3 OFF 003 Then OFF
```

If Power to EB8 #3 is OFF and for 3 minutes after power is returned, set the skimmer outlet state register to OFF.

The most common usage of the 'If Power' statement where no duration delay is used is for an Email alert. In this case, you typically don't want power failure notifications AFTER power is restored. In that case, a '0' delay is used in the Email outlet:

```
If Power EB8_5 000 Then ON
```

This sets the Email outlet register to ON as soon as power goes out. When power returns, the outlet state goes to OFF since there is no delay present.

There are two power sources available to the base module – via the EB8 and via a 12v adapter. Beginning with release 4.03, a new option has been added, enabling the base module to detect a power failure on the 12v port in addition to the EB8 AquaBus port. You can read more about this feature and how to provide uninterrupted power to your controller in the FAQ section at the back of this document.

Feed (and Maintenance) Cycles

If [FEED CYCLE] [DELAY] Then [ON/OFF/PROFILE]

Feed cycles are a special type of sub-program. The Apex has 4 “feed” cycles, A - D and although they’re called “feed” you can use them for any task where you need to have selected outlets on or off for a period of time.

You activate them via either the display or the web interface and they are configured a little different from normal outlet programs. First, using the display module you establish the duration of the individual feed cycles using the "*setup →outlet→feed interval*" menu. You can set a different time in seconds for each cycle. Once that is done, you go into the individual outlet programs you want as part of a particular feed cycle and add the statement "If FeedA 000 Then OFF" at the end. What that means is that "*if Feed cycle A is active and for 000 minutes after it's ended, this outlet is OFF*" .

Note – the feed cycle is not an outlet – it is a program statement you ADD TO AN EXISTING OUTLET to include that outlet in the feed cycle program.

If you had a skimmer that you wanted OFF for a feed cycle and keep it off for an additional 5 minutes after the cycle ends (to enable your sump level to stabilize if the return pump is off for example) then you would use "If FeedA 005 then OFF" as the last statement in your skimmer outlet program. Also, an outlet can belong to multiple feed cycles, each with a different ON/OFF state or delay time. It's completely up to you how you want to use this powerful feature. Again, the Feed statements are in addition to whatever else you want this outlet to do.

Feed cycle programming can also be used for maintenance cycles where you want a number of outlets to turn OFF or ON for some time. You can always cancel a feed cycle early so set your maintenance cycle to some high value, like 60 minutes.

Note that if you cancel a feed cycle, the delay variable (005 in the above example) is currently ignored - all outlets are immediately processed and set to whatever state their programs dictate.

Here's an example of how you could configure your different feed cycles. This is just an example, you can setup your feed cycles any way you wish or not use them at all.

Cycle	**Purpose**	**Duration (min)**	**Outlets ON**	**Outlets OFF**	**Extra Time ON or OFF**
FeedA	Feeding - Short	15	Heater Chiller Canopy Lights Fans	Return Pump Skimmer	000 005

FeedB	Feeding - Long	90	Heater Chiller Canopy Lights Fans	Return Pump Skimmer	000 005
FeedC	Skimmer Cleaning	30	All but Skimmer	Skimmer	005
FeedD	Maintenance	90	Canopy Lights Sump Lights Fans	Return Pump Skimmer Heater Chiller	000 005 000 010

Note that if outlets that you want ON are normally going to be ON during the times you would select those particular Feed cycles, you don't have to do anything with them – they're ON already. If the outlet would normally be OFF (like your Sump Light in the FeedD cycle) and you wanted it ON, then you would use the statement:

```
Set OFF
If FeedD 000 Then ON
```

Otherwise you're normally programming outlets to turn OFF during Feed/Maintenance cycles. Here's an example of a Skimmer outlet program with two different Feed cycles being referenced at the end:

```
Fallback ON
Set ON
If FeedA 010 Then OFF        ← if Feed cycle A, turn OFF and keep OFF for 10 more minutes
If FeedB 000 Then OFF        ← if Feed cycle B, turn OFF
```

If you have Tunze or VorTech pumps, you can reference an already created profile in your Feed statement. Just like the 'ON' or 'OFF', that profile will be activated for the duration of your Feed cycle. For example, suppose you had a profile called 'Slow' which ran your powerhead at a constant 20%. The statement would look like this:

```
If FeedA 000 Then Slow
```

At the completion of the Feed cycle, the pump would resume normal programming.

Alarms and Warnings

The Apex has the ability to trigger three different types of Alarms and Warnings: *Email Alarm, Sound Alarm* and *Sound Warning*. These virtual outlets exist by default with every Apex base module. They are called *EmailAlm_I5, SndAlm_I6 and SndWrn_I7*.

For the two sound outlets select your sound via the Display module *System – Sound Setup.* To program them, select the outlet from the Outlet Setup page. Select 'Advanced' for the control type. Then in the Advanced Setup box you can enter your alarm statements. You can test for any valid condition in your email alarm, sound alarm or sound warning. They can be different from each other or the same. An example warning condition might be If Temp > 83 Then ON and the alarm might be If Temp > 85 Then ON.

Note that the speaker on the Apex base module is not that loud. For a solution that will really get your attention look in the FAQ for the 'Failed Circuit Alarm'.

Chapter 6: Advanced Programming

Oscillate

OSC [DURATION / DURATION / DURATION] Then [ON/OFF/PROFILE]

The oscillate, or OSC statement is one of the most confusing of all the Apex commands. But it is a versatile command, capable of being used in a number of different scenarios. The main purpose of this command is to control multiple pumps to create artificial waves in your tank and while you don't have to use it with variable speed pumps, doing so gives you the ability to vary the intensity at the same time. But the oscillate command can be used anytime you have a repetitive event to program. Some common 'non-wave generation' uses for the oscillate command are:

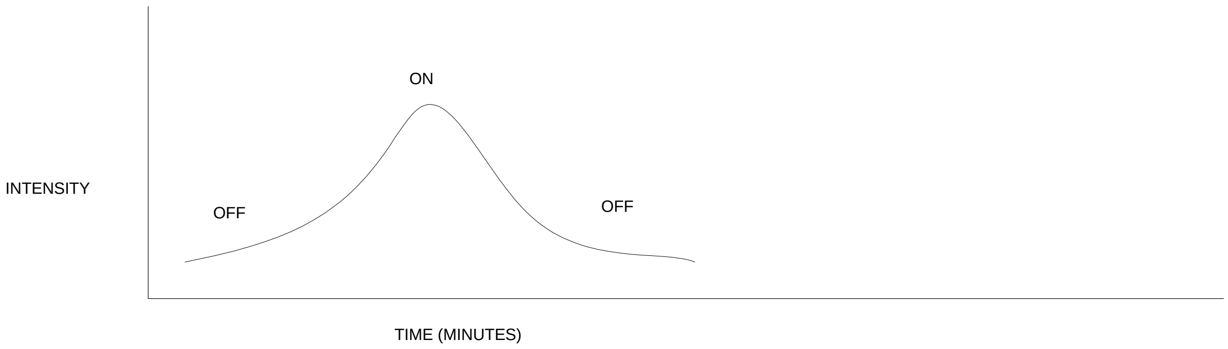
- Controlling dosing pumps
- Limiting when ATO pumps can run regardless of switch states
- Controlling water changes
- Limiting how much time reactors run (calcium, carbon, phosphate, etc.)

To explain how it works for waves, consider the following graphs:

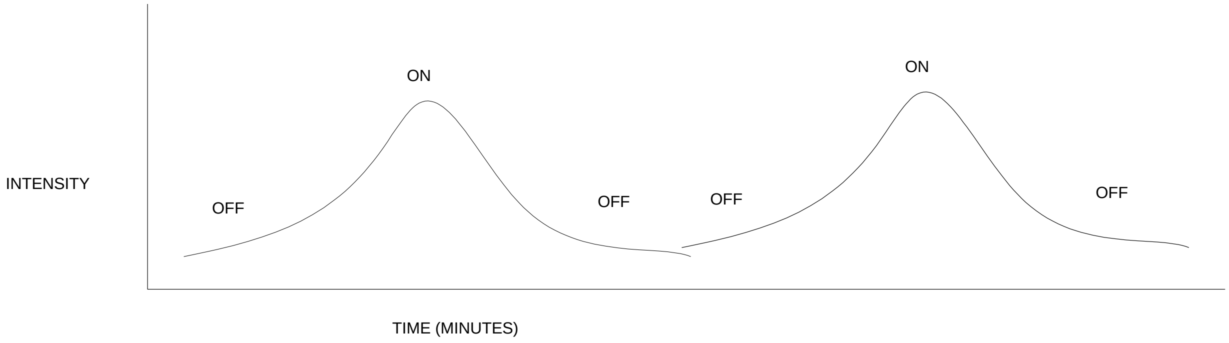
In this graph, a single pump using the OSC statement has a 3-phase cycle – OFF – ON – OFF.

Don't FORGET!

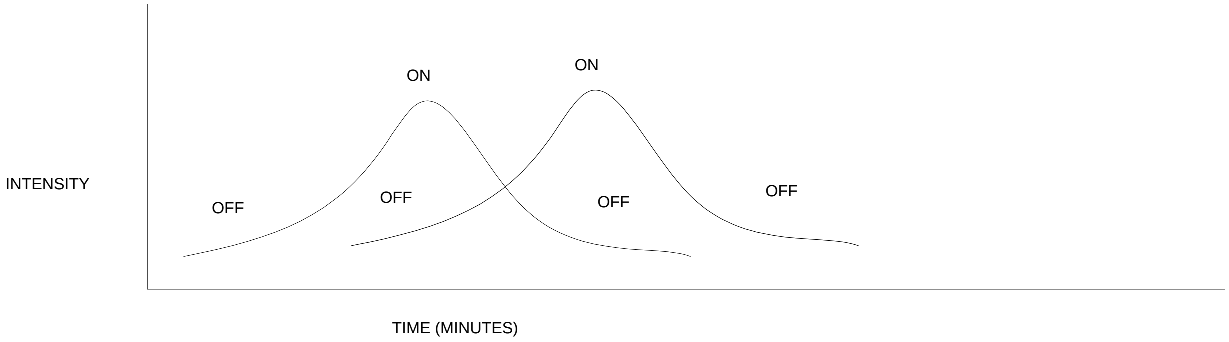
Note that in the following graphs, if you were not using the profiles to vary intensity and were only using the oscillate command, the graph would look more like a bar chart than a bell curve but the OFF-ON-OFF pattern would be the same.



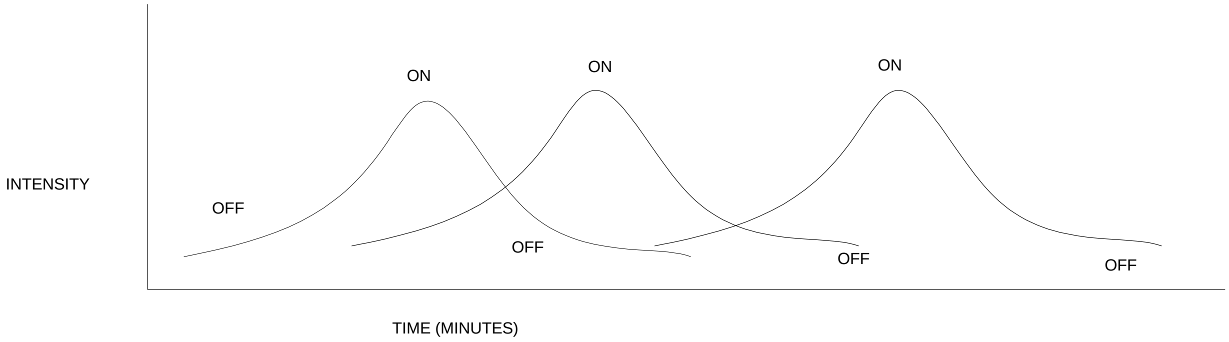
Now look what happens over time. A very rhythmic pattern is created but its little more than just sloshing back and forth.



But, if you add a second pump and take advantage of the offset timings that the OSC statement has, you begin to develop a pattern where you can overlap the begin/end of each pump phase



With 3 pumps of varying durations you can create patterns similar to what's depicted below – short – short – long.



The Oscillate syntax is as follows (Tip: there is no "If" conditional)

```
OSC MMM:SS/MMM:SS/MMM:SS Then ON
```

If you had multiple pumps, you would stagger the pumps OFF/ON/OFF times. There are some examples in the Neptune manual of how this would work.

You don't have to use all three parameters if you don't want. Consider a dosing pump that you want to run for 1 minute every hour, only at night. That pump program would look like this:

```
Fallback OFF
Set OFF
```

OSC 059:00/1:00/0 Then ON (note that leading zeros in minutes are not required)
If Time 07:00 to 18:00 Then OFF (overrides the OSC during the day and turns outlet OFF)

Now, if you really want to get wild and crazy with this statement you can reverse the OFF/ON/OFF cycle to ON/OFF/ON. To do this, substitute "Then OFF" for "Then ON" at the end of the command. So in other words your two choices are (substitute time in the format MMM:SS for the off or on):

OSC off/on/off Then ON
OSC on/off/on Then OFF

One final comment on the oscillate command. The 3 time values don't have to add up to exactly 1 hour or even 24 hours. They can be any combination of minutes and seconds you desire. The controller calculates the times starting at midnight but then resets every night at midnight. Even if you create a program using the OSC command in the middle of the day, the ON/OFF/ON cycle starts from midnight of that day. **So when testing your program, make sure you consider the current time of day when you attempt to verify whether your oscillate statement is working properly or not.**

Example 1 - Consider the following example (remember the times are OFF/ON/OFF in minutes and seconds):

OSC 0/15:00/45:00 Then ON

The above will be ON for 15 minutes beginning at midnight, then OFF for 45 minutes. So at 3pm it would be ON from 3pm – 3:15pm then OFF until 4pm and repeats through the day.

Example 2 - However, if we change it just a little:

OSC 45:00/15:00/0 Then ON

It will be OFF for 45 minutes at midnight, then ON for 15 minutes. So at 3pm it's OFF until 3:45, then it is ON from 3:45 – 4:00pm and continues to repeat that way through the day.

Example 3 - Or, reverse the OFF/ON/OFF to be ON/OFF/ON but keep the same 15 minutes out of the hour, however now you have two ON times per cycle instead of one:

OSC 7:30/45:00/7:30 Then OFF

All of the above examples run for 15 minutes out of the hour for a total of 6 hours per day – the only difference is when the 15 minutes happens – 1) at the top of the hour, 2) at the end of the hour or 3) a little up front and a little at the end! So keep this in mind when you're testing the command at 8pm and wondering why it doesn't work.

Finally, A future firmware release will change this if the sum of the ON and OFF times is not evenly divisible by 1440 (# minutes in 24 hours). In that case it will NOT reset the on/off/on timer at midnight each day.

Hysteresis - Minimum Time and Defer

No, not hysterics – that's what happens to your wife when your tank leaks. Hysteresis in the context of the Apex controller is a condition where the outlet state rapidly changes between ON and OFF based on the event you are testing. Consider the following program:

If Temp > 78.5 Then ON
If Temp < 78.5 Then OFF

As you might imagine, the outlet will be bouncing between ON and OFF as the temperature rises and falls by 1/10 of a degree. That's an artificial hysteresis – one easily solved by modifying those statements. But what about this example with a top-off float switch called Switch1:

```
Set OFF
If Switch1 OPEN Then ON
```

This would work if the switch tolerance was broad. What if the switch tolerance was only 1/16"? What if the water was sloshing around as in a sump? You would have a situation where the outlet was turning ON/OFF constantly as drips of water went into the sump.

To compensate for this Apex has two statements that perform similar but not identical functions – the Minimum Time and the Defer statements (Defer was introduced with 4.01).

Min Time

Min Time [DURATION] Then [ON/OFF]

The "Min Time" statement prevents the outlet from turning ON or OFF again for a certain amount of time that you specify (MMM:SS). It keeps it in its current state and prevents it from changing state for a certain amount of time. It doesn't stop it from turning ON/OFF now nor does it regulate the amount of time an outlet runs. It just forces the outlet to stay in its current state.

A common use is with metal halide lights. If you have a brief power failure that kills your lights, you don't want them trying to turn back on immediately. You want them to cool down for 5 or 10 minutes first, to avoid damaging the bulbs. You can use the Min Time statement for that. Or consider a top-off where you don't want the top-off happening again for a certain amount of time even if the outlet wants to. The Min Time statement is perfect for that. You could use it as follows at the end of your MH outlet statements:

```
Fallback ON
Set OFF
If Time 08:00 to 19:00 Then ON
Min Time 010:00 Then OFF
```

Defer

Defer [DURATION] Then [ON/OFF]

The Defer statement is slightly different from the Min Time. The Min Time prevents the outlet from turning ON or OFF again – but it does nothing to prevent the outlet from turning ON/OFF now. In other words, the Min Time keeps the outlet from changing state for a prescribed amount of time. The Defer statement on the other hand, delays the current state of the outlet. This is typically used to be certain the outlet is in the state you want, that it's not just a sporadic condition.

Consider the situation where the outlet state could change rapidly – a brief temperature spike caused by interference or a float that's cycling between OPEN/CLOSED due to water sloshing around. The Defer statement will insure the outlet (actually the outlet state register) remains in that state for an amount of time before actually changing that outlet physically.

For example, consider this email alarm outlet program:

```
Fallback OFF
Set OFF
If Temp < 76 Then ON
Defer 002:00 Then ON
```

The Defer command will ensure the email alarm outlet register stays in the ON state for a full 2 minutes before actually setting the physical outlet state to ON. If within that 2 minute period the outlet register was to be OFF after processing all the outlet commands, then the Defer command would not be satisfied, the outlet state would not be set to ON and in this case, no email would be sent. In the above example, if at 1:59 the outlet register went to OFF, the Defer timer would start all over again. The outlet must remain in the Defer state for the full Defer time. Note that the outlet's register might go ON/OFF based on various commands that are being processed but after all the commands are processed, the outlet register must remain in the Defer state or the Defer timer will start over.

The Defer command tests the outlet register to be the same state as its associated "Then" statement initially. So in the above example, if the temp was >= 76 initially, the outlet register would be OFF, the Defer statement would be skipped and the outlet's physical state gets set to OFF immediately. The Defer is not processed in that case.

You can stack two Defer statements together so long as they have different "ON/OFF" conditions. Here's an example where you want high/low temperature actions but want to make sure the temperature is not just a spike:

```
Fallback OFF
If Temp > 78.9 Then OFF
If Temp < 78.8 Then ON        These pair up
Defer 000:10 Then OFF          These pair up
Defer 000:10 Then ON
```

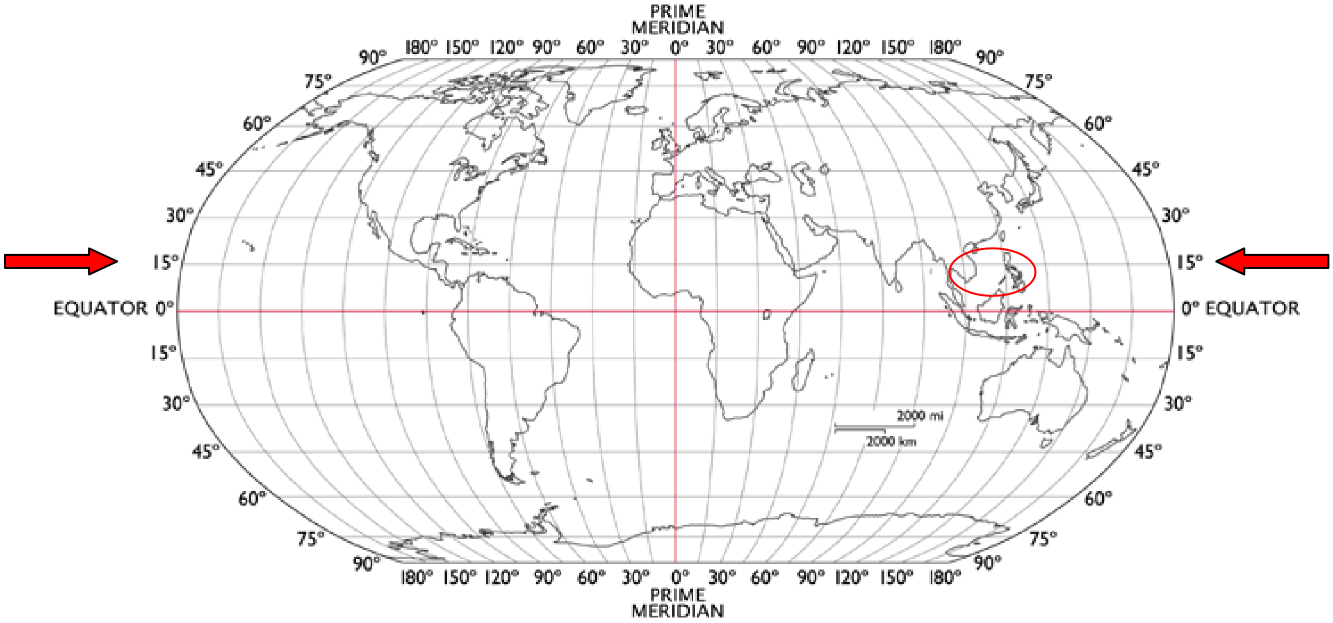
The first Defer will only apply if the outlet register is OFF which would be the "If Temp > 78.9" statement. The second Defer only applies if the register is ON which would be the second temperature statement. They're not physically related to each other – the Defer applies to the outlet state not any particular preceding command. In either case, the temperature statement must be true for 10 seconds in order for the physical state of the outlet to change.

Finally, it doesn't matter where in the outlet program the Defer statement is located as long as it is after the statement that sets the outlet register to the Defer test condition. In the above example, the Defer with the 'Then OFF' could have been immediately after the 'If Temp > 78.9' statement or it could have been at the end of the program. It will work either way. I find it easier to conceptualize when it's at the end but you can do it differently if you want to.

Device Type	**Statement to use**	**Purpose**
Lights	Min Time	Prevent lights from turning on again before cooling off
Top Off	Min Time	Limit how frequently the top can run
Float Switch	Defer	Make sure it's really in the proper state, not just sporadic
Alarm	Defer	Same
Email Alert	Defer	Same

Seasonal Features

The Apex controller has the ability to control your temperature and lighting to simulate the varying hours of light over the course of a year. You can view the seasonal table containing temperature, sunrise/sunset and moon rise and set at the *Seasonal Table* page (these are not user maintainable). The actual times are less important than how the hours of sunlight automatically change over the course of the year. Although the tables themselves are not user maintainable, you can +/- the times with a parameter (see below).



Temperatures

If Temp [</>] RT [+/-] [DIFFERENTIAL DEGREES] Then [ON/OFF/PROFILE]

You can adjust your temperature by using the seasonal temperature table. This table contains average water temperatures based on the first day of the month throughout the year. Other day's temperatures are interpolated between months.

To adjust your programming for regional temperatures (RT) you replace references to degrees in your temperature statements with the letters "RT" essentially saying "if my tank temperature is < > the regional temperature for this day then…" Further, you can add or subtract a differential to the RT temperature to tailor the temperature test. For example, consider this heater program:

Normal programming logic

```
Fallback OFF
Set OFF
If Temp < 76 Then ON
```

You can +/- a differential:

```
If Temp < RT-0.4 Then ON
```

Seasonal programming logic

```
Fallback OFF
Set OFF
If Temp < RT Then ON

If Temp < RT+0.4 Then ON
```

Sunrise/Sunset

If Sun [+/-] [DELAY] / [+/-] [DELAY] Then [ON/OFF/PROFILE]

You can adjust your lighting to reflect the seasonal differences in daylight hours by using the SUN command. As with the temperature, it can be tailored +/- within the statement. This is especially useful if you want to stagger multiple lights (for example Halides and Actinics) to simulate the gradual brightening and darkening of the sky yet still reference them to a changing lighting table. You can also use the adjustment to make the ON/OFF earlier or later while still observing the daily changes in the total hours.

Normal Programming Logic

```
Fallback OFF
Set OFF
If Time 07:00 to 18:30 Then ON
```

Seasonal Programming Logic

```
Fallback OFF
If Sun 000/000 Then ON
```

You can +/- a differential:

```
If Sun 060/000 Then ON   (sunrise delayed 60 min)
If Sun 000/-045 Then ON  (sunset hastened by 45 min)
If Sun 045/045 Then ON   (both delayed by 45 min)
If Sun 0/0 Then OFF (outlet OFF during daylight)
```

Remember, the time is MMM and can be positive (default) or negative (include "-" sign). First parameter is sunrise, second is sunset. And, there isn't a "not Sun" type logic that lets you test for night vs. testing for daylight. If it's daylight (the Sun command) then you can deem the outlet ON or OFF, not the other way around. To get around this, establish your "nighttime" outlet state first and then use the Sun command to override it, as in the pump statement below:

```
Set PF_Night                (activates the profile PF_Night which has a night pump profile defined)
If Sun 000/000 Then PF_Day  (overrides this during the seasonal day with a different pump profile)
```

The Sun command (and the following Moon command) have an implied "ELSE [ON/OFF]" function. Normally, you need a statement preceding your light statement that sets your outlet state to OFF. Then your statement to turn on your lights simply turns them ON – the preceding statement turns them OFF. That's not required with the Sun or Moon statements. Review again the 'Normal' example to the 'Seasonal' example above. Note that in the normal you have the 'Set OFF' then the 'If Time'. You need that pair for the outlet to function correctly. But you don't need the 'Set OFF' in the Seasonal example – it's implied. If the clock time is outside the boundary of your 'If Sun' statement, then the implied 'else OFF' or 'else ON' takes effect.

Moon Rise/Set

If Moon [+/-] [DELAY] / [+/-] [DELAY] Then [ON/OFF/PROFILE]

The moon rise and moon set is identical to the sunrise sunset command. However, one thing you have to be aware of is that as in real-life, some days the moon actually rises or sets during daylight hours. The Apex faithfully replicates this behavior with the moon tables. So don't be surprised if you see your moon lights on during the day – it's the way nature intended!

Normal Programming Logic	Seasonal Programming Logic
Fallback OFF	Fallback OFF
If Time 18:30 to 07:00 Then ON	If Moon 000/000 Then ON
You can +/- a differential:	If Moon 060/000 Then ON (rise delayed 60 min)
	If Moon 000/-045 Then ON (set hastened by 45 min)
	If Moon 045/045 Then ON (both delayed by 45 min)

Remember, the delay time is MMM and can be positive (default) or negative (include "-" sign). First parameter is moon rise, second is moon set.

If you want to vary the *intensity* of the moon lights and not just the rise/set, you can purchase the Lunar Simulator module with LED lights. There is a value in the seasonal tables called *"Int"* which roughly equates to the waxing and waning (Intensity) of the moon as it goes through its monthly phases (note that this is not intended to match any current moon phases – it's designed to simulate the moon's monthly brightness). This integrates with the Lunar Simulator module providing a seamless lunar solution and functions similar to the ramp function with variable speed ports. For more information on the LunarSim module, refer to Chapter 7.

Gravitational pull will have to wait for a future release.

Variable Speed Pumps and Dimmable Lights

The Apex (not the Lite) comes with four variable speed outlets that can be used for controllable pumps like **Tunze Streams** or dimmable light ballasts from **Sylvania** and **Phillips**. The outlets are named VarSpd1_I1, VarSpd2_I2, VarSpd3_I3 and VarSpd4_I4 by default. You configure these outlets by defining profiles then referring to the profile using normal program statements.

Profiles

A profile contains the specific commands to vary the voltage and duration that these devices use. Profiles have default names of PF1, PF2, up to PF16 but as with modules and outlets, you can name them anything you like (limit of 8 char). Also note that while you can vary the voltage on these ports (0 – 10vdc) there is minimal current, something < 10ma. This is for signaling only and is insufficient to power anything.

Once configured, you refer to a profile instead of the "ON" command when programming.

Traditional program statements	Using a Profile instead
Set ON	Set PF1
If Time 08:00 to 19:00 Then ON	If Time 08:00 to 19:00 Then PF1

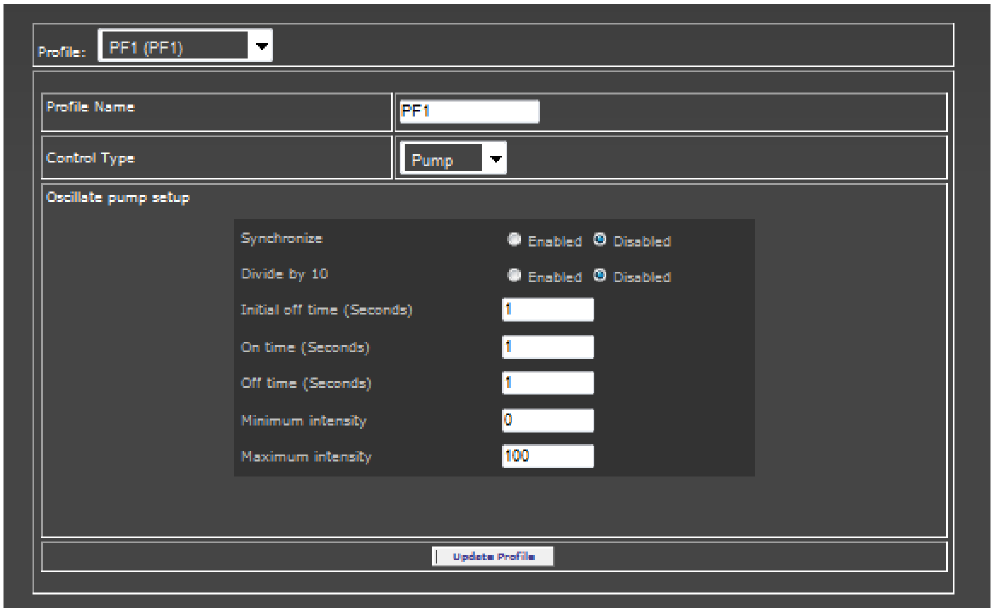
Profiles can be used in the following commands:

- Day of Week
- Feed
- Moon Rise/Set
- Oscillate
- Outlet Testing
- Power
- Probe Testing
- Regional Temperature
- Set
- Sun Rise/Set
- Switch Testing
- Time

To create a profile, use the *Configuration – Profiles Setup* page. Similar to configuring an outlet, you select the type of profile you want (*Pump, Ramp or VorTech*), then fill in the blanks. Your intensity parameters can either rise (i.e. go from 30% - 100%) or fall (i.e. go from 100% to 30%).

Variable Speed Power heads (PUMP)

Below is the configuration screen for the PUMP control type. The fields for *Initial Off, On Time* and *Off Time* should look familiar – they're the arguments you need for the *oscillate* command. The *Minimum* and *Maximum Intensity* fields (percentages) define how the pump speed is going to change when going from off to on, essentially each 10% of intensity equals 1 volt DC for a total of 100% or 10vdc. For now, let's set aside the remaining two fields – *Synchronize* and *Divide by 10*.



Profile: PF1 (PF1)

Profile Name	PF1
Control Type	Pump

Oscillate pump setup

Synchronize	○ Enabled ● Disabled
Divide by 10	○ Enabled ● Disabled
Initial off time (Seconds)	1
On time (Seconds)	1
Off time (Seconds)	1
Minimum intensity	0
Maximum intensity	100

The off/on/off times (oscillate) can be used to either stagger multiple pumps or they can be used to synchronize the pumps in such a fashion that they build, one off the other. The Apex manual has some wave test results you can refer to for various size tanks and numbers of pumps.

Connecting Tunze power heads

Neptune Systems sells a cable specifically designed to connect the Tunze controllable powerheads (i.e. 6055, 6105, 6205 and 6305 power heads that are compatible with the 7091 or 7095 Tunze controllers) to the base unit called a "*2 Channel Apex to Stream Cable*". You can purchase it from Neptune here: http://www.neptunesys.com/index.php?page=shop.product_details&flypage=shop.flypage&product_id=70&category_id=17&manufacturer_id=0&option=com_virtuemart&Itemid=26

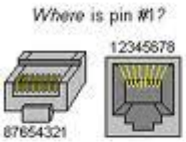
Tunze power heads come with a transformer. This transformer plugs into a regular wall outlet and converts 120vac into a variable DC voltage. You do not need to use an EB8 outlet. The 2 Channel Apex to Stream cable from the Apex plugs into that transformer and essentially acts as an 'on/off' switch and a variable speed control. If you want to turn off the power head completely, set the intensity to 0%.

Making your own cable

DISCLAIMER You are on your own. Verify all pinouts with a circuit tester before connecting anything. If you fry your controller or pump, don't blame me! If you know enough about crimping RJ-45 connectors then you know where to source the crimping tool and the blank connectors. The DIN-5 plugs are harder but you can find cables with a DIN-5 male on each end. Cut the cable in half and you have (2) connectors. Here's one source: http://www.cablesondemand.com/product/CS-DNPRDS5MX2/URvars/Items/Library/InfoManage/CS-DNMS6MD6F0.htm

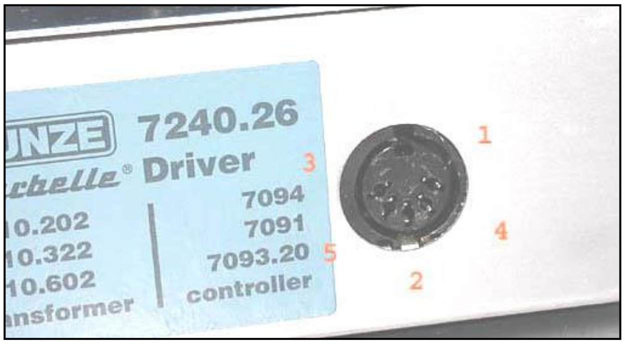
The cable (that you can get from Neptune for $20 – keep that in mind) has an RJ-45 connector at the Apex end and a DIN-5 connector at the Tunze end. Both are male. As you know, an RJ-45 can accommodate up to 8 wires although in our case only 4 are required. If you've ever crimped an RJ-45 connector you know that it's far easier to leave the unneeded wires in there than it is to cut them and try to line everything up. The only wires you need are 3,4,7 & 8. One RJ-45 per (2) Tunze pumps.

- The orientation here is important. Looking at the female Apex variable speed receptacle where the tab is DOWN, the pins are numbered from left to right, 1 – 8. Looking at the male RJ-45 connector with the tab DOWN, the pins are numbered from left to right 8 - 1. See drawing below.



Pin	Apex RJ-45 Usage
1	0-10vdc V1 and V3
2	Ground
3	
4	
5	0-10vdc V2 and V4
6	Ground
7	
8	

Pin	Tunze DIN-5 Usage
1	Not used
2	Not used
3	Not used
4	Ground
5	0-10vdc



On the Tunze end, only 2 of the 5 wires are required. The picture to the left shows the female DIN-5 connector on a Tunze controller.

Looking at the male end the pins are numbered clockwise 1, 4, 2, 5, 3.

Programming Tunze

You would program a power head like the Tunze by first configuring one or more profiles (you can have up to 16). Then, simply reference that profile name in your VarSpd outlet using any one of the statements that support the profile option (see *Programming Quick Reference Guide*), modifying it as you need:

- Set PF1
- If Time 08:00 to 18:00 Then PF1

Synchronizing Multiple Pumps

For more coordinated timing, use the profile "*synchronize enabled*" option with multiple profiles. For example, if you were to define three profile statements (assuming you have 3 Tunze pumps), the synchronize option automatically starts profile #2 at the same time profile #1 starts, #3 starts when #2 starts, etc. Technically, it synchronizes the 'initial off time' of the pumps. It uses the next lowest profile number (regardless of what you named it) to synchronize to. In these cases you would simply use the "SET" statement and would not use the "If Time" statement.

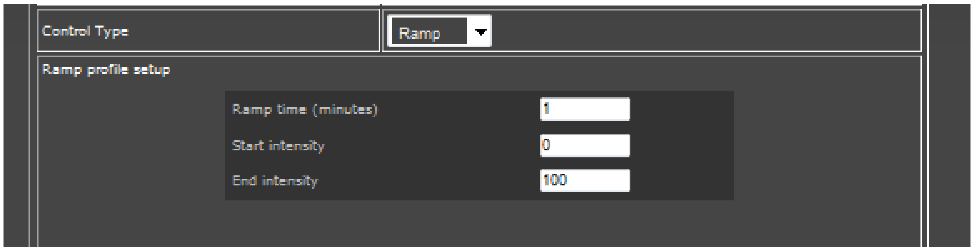
Divide by 10 Option

One final profile option needs to be discussed. If you need really fine control over your pump profile timing, you can enable the '*divide by 10*' option. This option essentially changes the oscillate portion of the profile from using seconds as the unit of measure to using tenths of a second as the unit of measure. So if your original values for off/on/off were 30/5/120 (seconds), the divide by 10 option processes those as 30/5/120 (seconds/10). The values themselves don't change – when you go back to your profile the numbers will be the same – but the controller interprets them differently such that 120 seconds becomes 120 *tenths* (or 12 seconds). 5 seconds becomes 5 *tenths* (or ½ second).

Note that it is not recommended to run Tunze power heads at less than 30% intensity. The voltage generated at that level is insufficient to properly energize the impeller coil causing the impeller to wobble and wear prematurely. This is the same reason older Koralia pumps are not supposed to be cycled on/off as with the oscillate command and the cause of the clanking you hear when they first start. In addition, at low voltages (1 – 29%) you may experience erratic pump behavior with the impeller starting and stopping at random. Run the pumps at 30% - 100% with 0% being OFF.

Dimmable Lights (RAMP)

The dimmable lights profile uses the RAMP control type. The setup for the RAMP control type is simpler than the PUMP setup and only contains three fields – *Ramp Time, Start Intensity and End Intensity.* The intensity fields perform the exact same function as in the PUMP type – they vary the voltage to the device from x% to y%. The *Ramp Time* defines how long this should take but unlike the PUMP profile type which is in seconds, the RAMP profile type is in minutes.



Control Type: Ramp

Ramp profile setup

Ramp time (minutes): 1

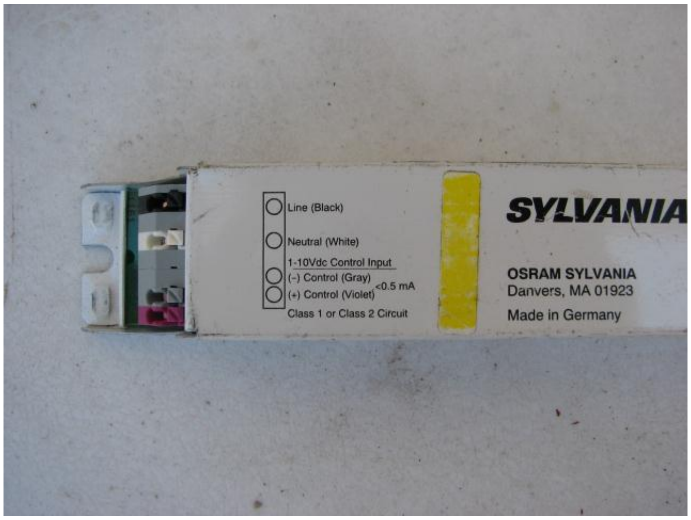
Start intensity: 0

End intensity: 100

Connecting dimmable lights

Installing dimmable lights is a bit of a DIY project as there are no off the shelf dimmable light fixtures compatible with the Apex. All the dimmable fixtures use their own proprietary control system. You must custom wire a dimmable ballast to your lamps, whether T5, LED or whatever. The RJ-45 (Apex) side of the pin out diagram for the Tunze pumps is the same as for the dimmable ballasts (pins 3,4 and 7,8).

To build your own you will need a power cable, the interface cable from the base unit to the ballast and the end caps to support the appropriate bulbs. The interface cable available from Neptune Systems is a "*2 Channel Apex to Light Dimmable Cable*" http://www.neptunesys.com/index.php?page=shop.product_details&flypage=shop.flypage&product_id=69&category_id=17&manufacturer_id=0&option=com_virtuemart&Itemid=26&vmcchk=1&Itemid=26:



Neptune certified and tested their system using a Sylvania QT2X54 dimmable T5 ballast which will support (2) 54 watt T5 HO fluorescent bulbs. As you can see from this photograph, there are separate connections for 120v power as well as the 0-10V DC control. The other end of the ballast has the lamp connections.

Advance/Phillips also makes a dimmable T5 ballast called the Mark 7 RZT-2S54 that is compatible with the Apex and supports the same bulbs as the Sylvania. The specifications can be found here: http://www.advance.philips.com/documents/uploads/literature/EL-2010-R05_Mark7_FINAL.pdf

LED Light Fixtures

There has also been success reported connecting LED's to the variable voltage ports. Mean Well's ELN series (ELN-30, ELN-60) of switching power supplies will accept a 0 – 10vdc signal that drives the DC output voltage. Neptune has not tested this with the Apex controller however and would not be in a position to support any issues you might have. You can get more information from Mean Well's website at www.meanwell.com .

Programming dimmable lights

Programming for dimmable lights is a little different than regular lights or even variable voltage pumps. For starters, you use two outlets for each pair of lights – an EB8 outlet for the 120vac power and a variable speed outlet to control the dimming. Next, you have to create the various profiles for starting the lamps and shutting them down. In the example below, two profiles have been created called "*RampUp*" and "*RampDown*" (you could have left them as PF1 and PF2 but what fun would that be?). Finally you program both the variable and the EB8 outlets, synchronizing their times.

The program statements would look like this:

	Profile *RampUp*	Profile *RampDown*
Ramp Time	60	60
Start Intensity	0	75
End Intensity	75	0

VarSpd outlet (0 – 10vdc):

```
Set OFF
If Time 08:00 to 09:00 Then RampUp          ← ramps it to 75%
If Time 09:00 to 19:00 Then ON              ← jumps it to 100%
If Time 19:00 to 20:00 Then RampDown        ← ramps it back down to 0%
```

EB8 outlet (120vac):

```
Fallback OFF
Set Off
If Time 8:00 to 20:00 Then ON
```

The ballast is energized from 8am – 8pm (08:00 – 20:00) per the EB8 outlet. At 8am the initial profile "RampUp" is also activated per the VarSpd outlet. That profile takes the intensity from 0% to 75% over 60 minutes. The light is at 100% on from 9am – 7pm. Then at 7pm (19:00) the "RampDown" profile takes the intensity from 75% to 0% over 60 minutes. At 8pm the EB8 outlet is turned off and the light goes out completely.

EcoTech Marine’s VorTech Pumps (VorTech)

Effective with release 4.03, Neptune has released the Wireless Expansion Module (WXM) to enable integration of VorTech propeller pumps with the Apex controller. With the WXM module you can use any of the VorTech pump modes in your Apex program statements. Essentially the WXM module is sending commands to the VorTech driver to change modes. The actual pump control where the pump speed increases/decreases based on the mode’s profile is done by the VorTech driver. This is different from the way Tunze pumps are handled with the Variable Speed ports. There you control the actual pump intensity profile but with the VorTech pumps, the VorTech driver takes care of that. You just define some variables that the VorTech driver uses like *Maximum Intensity* and *On Time*.

VorTech vs. Tunze

No, you’re not going to see a debate between VorTech and Tunze to find out which one is better. Both are excellent pumps that have the ability to be controlled and can generate a wide variety of flow patterns in a tank. However, the two manufacturers implement this technology differently.

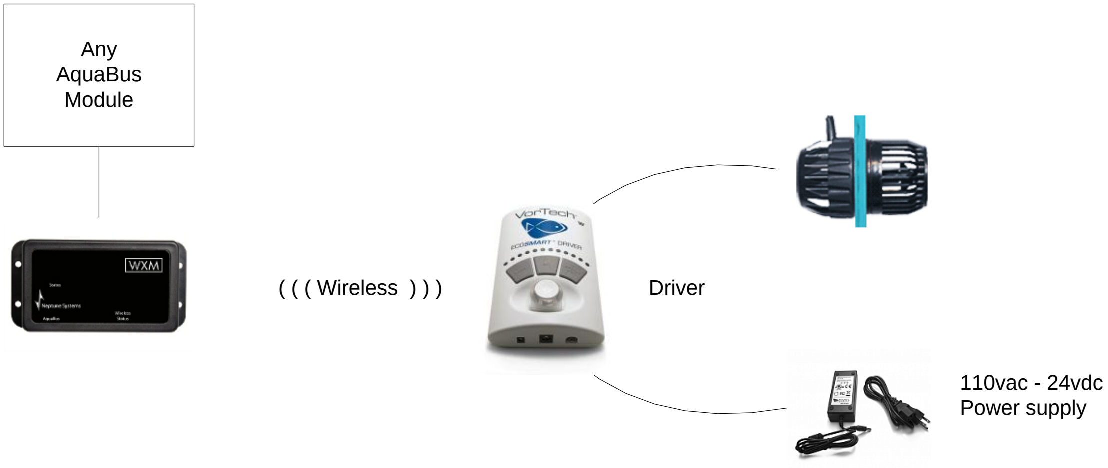
Tunze pumps sit entirely inside your tank and have the ability to swivel and be pointed where you want. Tunze pump profiles are completely configurable by you. When you setup a Tunze pump profile on the Apex, you define the intensity, the oscillation durations, the synchronization and even whether you want the times in tenths of a second. The VorTech pumps on the other hand, have these profiles pre-defined. You simply define the maximum intensity as a % and for some modes, the oscillation (called ‘pulse’ by VorTech) durations. The rest of the profile timing is built into the VorTech controller/driver. VorTech pumps have a ‘wet side’ and a ‘dry side’ with the dry side transferring power via magnets to the wet side which is in the tank. VorTech pumps have a wide flow pattern and according to EcoTech, don’t need to be pointed to have full tank coverage.

VorTech/WXM Hardware

There are 4 components to the VorTech/WXM system:

- The Neptune WXM module (will control up to 11 pumps)
- The VorTech wireless driver, either the Gen2 or EcoSmart
- The VorTech pump, either a MP10 or 40 (must be matched to the driver)
- A 110vac to 24vdc power supply

The WXM module is an AquaBus module. As such, it is ‘plug and play’ with the Apex system. Just plug in the supplied AquaBus USB cable to ANY available AquaBus port, whether on the base unit, an EB8 or any other AquaBus module and the base unit will recognize it and add it to the AquaBus chain. It will get the next available AquaBus address and will appear in your module list as ‘WXM_3’ (or whatever the next available address is on your bus).



The diagram above depicts how a typical WXM/VorTech installation would connect together. If you had multiple pumps you would repeat the right side for each pump (each VorTech pump requires a power supply and a matching driver). It is not necessary to use EB8 or DC8 outlets for the VorTech power supply connection. You can simply plug them into a wall outlet or power strip if you want. You will be able to turn the power head completely off from the Apex.

VorTech Software Installation

The Neptune documentation for the WXM module provides detailed instructions for getting the proper firmware in the WXM and the EcoTech driver. **The steps presented here are done on the Base module not the web pages.**

1. Download and install the most recent firmware to your Apex base unit (minimum of 4.03_4E10).
2. Update the module firmware in the WXM module.
3. Download the appropriate driver firmware (Gen2 or EcoSmart) from the Base module to the WXM module.
4. Wirelessly update the EcoTech driver with the firmware from step #3
5. Repeat steps 3 and 4 for each VorTech pump you have.

At this point you are ready to program and use the VorTech pump(s). As soon as step 4 is complete, that pump will be recognized by the Apex controller and a virtual outlet called 'VorTech_x_y' created where 'x' is the WXM module number and 'y' is a number from 1 – 11. For example 'VorTech_3_1'. If you added a second pump, it would be named 'VorTech_3_2' and so on. As with any other virtual outlet, you can then rename it to something you like or keep the default name. The limit is 12 characters, no spaces, case sensitive – just like every other outlet name.

So to recap, at this point we have created (3) new objects in our Apex:

- The WXM module with a default name of 'WXM_3'
- Pump called 'VorTech_3_1'
- Pump called 'VorTech_3_2'

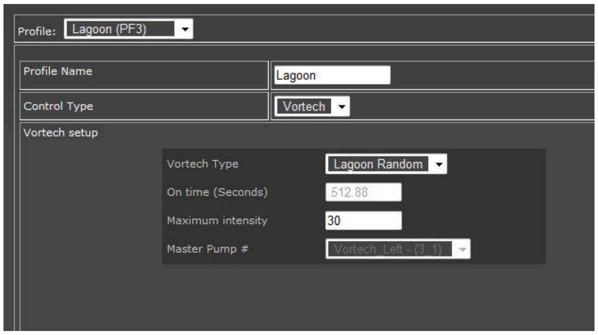
Errors

- *Download Error* – you might see this in step #3 above when downloading the driver firmware. Cycle the power to the WXM module by unplugging the AquaBus cable. Wait a couple of minutes then plug it back in and try again.
- *Update Error 0* – Make sure you have updated the web pages in step #1.

- *Missing* – VorTech pump shows as ‘missing’ on the status screen. The WXM has lost communications with it. Try moving it closer to re-establish communications, then place it back in position.

VorTech Programming

Remember that the VorTech pumps are created as virtual outlets. However, instead of using ON/OFF to control the pump, we need to use profiles. As with any outlet, it can then be referenced in any program statement that supports the PROFILE option. You can see a complete list of all program statements and their syntax in the Programming Quick Reference Guide in the Appendix.



Profile: Lagoon (PF3)

Profile Name: Lagoon

Control Type: Vortech

Vortech setup

Vortech Type: Lagoon Random

On time (Seconds): 512.88

Maximum intensity: 30

Master Pump #: Vortech Left - (3 1)

Before you can start referencing these new outlets, you have to establish profiles for the various VorTech pump modes like *Constant, Lagoon, ReefCrest*, etc. This is done through some new fields in the *Configuration – Profiles Setup* web page (or if you're a glutton for punishment you can use the Display module). There's a new profile *Control Type* called ‘VorTech’ and some new options – *VorTech Type, On Time, Maximum Intensity and Master Pump#.* These options are the equivalent to using the various keys on the VorTech driver to configure your pumps but in this format they are infinitely easier to use.

Let's take a simple example where you just want to have your pump operate in the gentle Lagoon mode. Further, you only want the maximum intensity to be 50% of the pump's capability. Here's the steps:

1. Select an unused profile from the *Profile* drop-down box.
2. Give it a meaningful *Profile Name* – I used ‘*Lagoon’*.
3. Select ‘*VorTech’* for your *Control Type*
4. Select ‘*Lagoon Random’* from the *VorTech Type* drop-down box
5. Ignore ‘*On Time’* and enter ‘50’ for the ‘*Maximum Intensity’*
6. If you had more than one pump, you would select which one is the ‘*Master Pump #*’ otherwise ignore that field.
7. Update/Save this profile.

Now, all that's left is to assign this profile we just created called ‘Lagoon’ to the virtual pump outlet. Again, you can use any program statements you want. We'll keep it simple and just use the following:

Fallback ON
Set Lagoon

What else could we do? Well, you could create a profile called ‘Emergency’ that uses the ‘Constant’ mode with a maximum intensity of 25%. Then change your Fallback statement to

Fallback Emergency

You could create a profile called ‘Slow’ that again uses the ‘Constant’ mode at 10%. Then add that to a Feed statement like this:

If FeedA 000 Then Slow.

You can create up to 16 profiles and each one can reference a different VorTech mode with different maximum intensities. The actual VorTech pump program contained in the Driver will then operate the pump, varying the intensity per the VorTech program up to your maximum intensity. For the modes that use a frequency to build waves (*Short Pulse, Long Pulse*) you use the *On Time* option when defining the profile.

VorTech Pump Modes

Constant

This mode is just what it says, constant speed. When you define your profile, you pick a speed/intensity % and the pump stays at that speed the entire time.

Lagoon

This is one of two random modes (the other being Reef Crest) where the VorTech driver varies the pump intensity up to a maximum percentage you define. With the Lagoon mode, the pump will run from about 80% to 100% of the maximum you specify in a random pattern simulating a quiet lagoon. If you were to graph this it would look like rolling hills.

Reef Crest

This is the other random mode where again, you define the maximum percentage of pump power. In this mode however, the intensity will vary from 10% to 100% and do so in much shorter periods of time, simulating the more violent action of the leading edge of a reef. If you were to graph this it would look like a seismograph.

Pulse (Short and Long)

Pulse modes are used to create waves. There are two pulse modes on the VorTech driver that have been combined into one on the WXM. The difference between them is that the short mode goes in seconds and the long mode is configured in minutes. Since the Apex can deal in seconds and tenths of seconds there was no need for two modes. Again, you define your maximum intensity % but you also define the pulse duration, the ‘On Time’. Pulses can be timed from .25 seconds all the way up to 1300 seconds.

Sync

Sync is used when you have more than one pump. You establish the mode of the first pump using any of the available modes. On the second or subsequent pump, you ‘sync’ it to the first pump. This way, you only have to define one pump and the other pumps just perform the same speed/time.

Anti-Sync

Anti-sync is also used when you have multiple pumps but in this case, the pump that is ‘anti-sync’ is exactly opposite the primary or master pump. So if the master is currently running at a high intensity, the anti-sync pump(s) are running at a low intensity.

Tidal Swell (EcoSmart drivers only)

This mode requires multiple pumps to operate correctly. Using a converging pattern of varying intensities, the tidal swell starts with separate swells then gradually over an hour synchronizes them to have all pumps working together. Then it starts over. Again, you define intensity and swell duration.

Nutrient Transport (EcoSmart drivers only)

This mode combines short-pulse and long-pulse together to stir up detritus in the tank and suspend it long enough for your filtration system to remove it. The process occurs over a 30 minute period and can repeat for as long as you wish. Multiple pumps will synchronize together to provide greater coverage and effectiveness.

Eco-Back (EcoSmart drivers only)

This is a special mode for pumps attached to the back of a tank and is designed to be used in conjunction with one or more additional pumps on the tank sides. It takes the sync / anti-sync concept one step further and integrates pumps on the sides and back of the tank into a unified system. The detailed EcoTech manual has more on this along with some graphics to better explain the concept.

Modes not supported by the WXM

There are three modes not incorporated into the WXM module: Night, Feed and Auto-Tune. The reason the Night and Feed modes were omitted is omitted is obvious. The Apex controller has a more flexible and powerful time and feed mode options. Simply create a profile using the Constant mode, give it a handy name like “Slow” and reference that in your Feed or Time statement like this:

```
If FeedA 000 Then Slow
If Time 19:00 to 08:00 Then Slow
```

In fact, because of the ability to have these separate program statements, you could create a slower (lower intensity) version of Reef Crest for example, and use it at night or during feedings – something impossible to do with the standard VorTech driver.

The Auto-Tune mode is not really a mode, it’s simply an easy way to step through intensity settings in the short or long pulse modes to establish what works well in your tank for a wave – kind of like scanning channels with your radio. When in Auto-Tune, the VorTech driver steps through the pulse durations automatically. When you find the one that works best in your tank, you press the SET button to store it. With the Apex, you just have to use trial and error to find the proper wave pulse timing.

Chapter 7: Other Neptune Accessories

Neptune manufactures a number of additional modules that are designed to expand the capacity of the base module when it comes to probes and switches. These modules fall into three basic categories – those that are output-only and connect via the legacy 'Control Bus' method, those that require input-output communications and connect via the Mini DIN-8 connector and the newer, more advanced 'AquaBus' which uses USB style connectors and is input-output. The AquaBus modules are not backward compatible with the older AC line of controllers and are only used on the Apex and Apex Lite. The control bus and the DIN-8 modules are compatible with the Apex and Apex Lite

Control bus devices use an RJ11 telephone style connection. They require an external power source. The control bus requires you to manually set the module's bus address via 'dip switches', either up or down that together form a pattern A1, B1, B2, H3, etc. This address must be unique so care needs to be exercised when setting the switches. You also have to manually add the module to the Apex via the *Module Configuration* web page in order for the default outlets to be generated. The AquaSurf, LunarSim, DC8, DC4 and DC4HD are all control bus type devices.

The Mini DIN-8 is used by the PX1000 and the I/O Breakout Box. They also require an external power source. You don't have to set dip switches to establish its bus address. These connect to a female DIN-8 jack in the Apex base unit. The PX1000 also has a female DIN-8 jack for chaining up to 6 more PX1000 modules together. This was the predecessor to the AquaBus.

The AquaBus modules on the other hand, are true 'plug and play' meaning that you just plug the module in and the controller recognizes it, automatically establishing its bus numbering, adding the module and creating the outlets. They get their power from the bus itself (the obvious exception is the EB8 of course). The AquaBus can be up to 200' in length.

The following table summarizes the various modules and their capabilities:

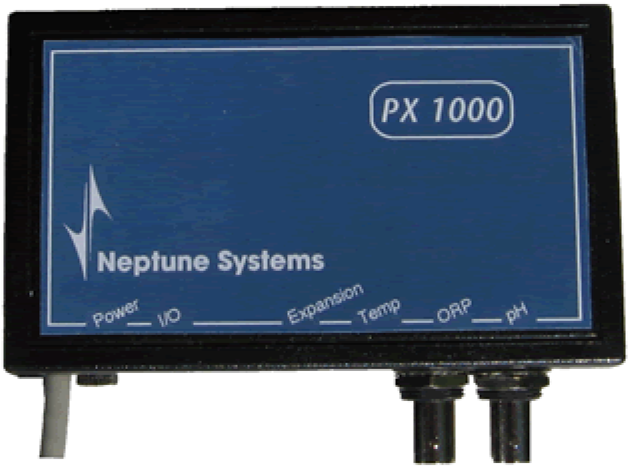
Module	Purpose	Interface	#Switches	Temp	pH only	pH/ORP (2)	Conductivity	Dissolved Oxygen
PX1000	Probe expansion	DIN-8	6 (1)	1	1	1	N/A	N/A
LunarSim	LED Moon Lights	Control	None	N/A	N/A	N/A	N/A	N/A
AquaSurf	Variable Speed Pumps	Control	None	N/A	N/A	N/A	N/A	N/A
PM1	Temp, pH Probes	AquaBus	6 (1)	1	N/A	1	N/A	N/A
PM2	Temp, Conductivity Probe	AquaBus	6 (1)	1	N/A	N/A	1	N/A
PM3	Temp, Dissolved Oxygen Probe	AquaBus	6 (1)	1	N/A	N/A	N/A	1
I/O Breakout Box	Switches	DIN-8	6	N/A	N/A	N/A	N/A	N/A

(1) – requires the I/O Breakout Box to get the 6 switch connections
(2) – software configurable for either (1) pH or (1) ORP probe

PX1000 Probe Expansion Module

The PX1000 was designed to support additional Temp, pH and ORP probes as well as contact switches on the AC line of controllers but can be used on the Apex and Apex Lite as well. The module connects via a DIN-8 cable to the base module. While there is only one DIN-8 jack on the base, there is an additional female DIN-8 jack on the PX1000. If you wanted to connect both a PX1000 and an I/O Breakout Box, you would connect the PX1000 to the base module and the I/O BB to the PX1000 'I/O' jack. The 'Expansion' DIN-8 jack would be used to connect a downstream PX1000. You would repeat the process with additional PX1000 modules.

You can have up to (7) PX1000 modules on the system. When the first PX1000 is plugged into the controller, you will have to tell the controller it exists. You do this via the display module by selecting *System – Prob Exp,* then reboot the controller. There are no outlets to create with a PX1000. The probe names will follow the same syntax as other expansion modules: **[PROBE] [EXPANSION MODULE] [PROBE NUMBER].** Probe = Temp, pH, ORP; expansion module = blank for the base unit then A, B, C, D, etc; probe number is 1 or 2 and is only used for the pH probe where you can have multiple probes on one expansion unit.



Note: if you don't see your probes after you connect them to your PX1000, check to make sure the probe is enabled (also make sure the probe expansion module is enabled, see above). From the display select *Setup – PH Setup – PH Enable* and select your new pH probe from the list of pH probe names. Do the same for the other probe types as each one has a menu item to enable or disable it. If you add a PX1000 and only want a pH probe, you would use the same menu commands to DISABLE the temperature probe so it doesn't show up in your status screen with a temperature of 20 degrees!

There is no difference to the program statements when using a PX1000 other than the probe names as I've already described. Any statement that you can reference the base unit probes can be used to reference the PX1000 probes.

Connection steps:

(1) Plug in the PX1000 to the base module
(2) Enable probe expansion via *System – Prob Exp*
(3) Reboot
(4) Enable probe(s) via *Setup – pH Setup – pH Enable*. Repeat for remaining probes.
(5) Calibrate your probe (pH only)
(6) Your probes are now ready to be used

Lunar Simulation Module

The Lunar Simulation module (LunarSim) makes it very easy to connect LED moon lights to your system and simulate the monthly phases of the moon as well as the varying seasonal moon rise/set times. This data can be found in the Seasonal Tables accessible from the web pages (See the 'Moon' command in Advanced Programming for more information). The 'INT (intensity) value is a rough equivalent to the waxing and waning of the moon – the highest number for the month equals a full moon, the lowest equals a new moon. The LunarSim module can be purchased with either 2, 3, 4 or 5 LED lights as a kit. It is an integrated system, getting its commands from the controller and powering the LED lights – do not attempt to wire up your own LED lights. Use the ones provided by Neptune and save yourself some headache.

The LunarSim module is a 'control' type module. It uses an RJ11 phone style cord (many phone cords only have (2) wires. The control bus requires (4) wires) to connect to the base unit's control port. Connect the other end of the cable to the LunarSim 'Control In' jack. Because it's a control type module you have to manually set its address on the control bus. If you don't do this correctly and assign an address that's already in use, this module won't work and in all likelihood the other module won't either – so take a couple of minutes here and understand what you're doing.

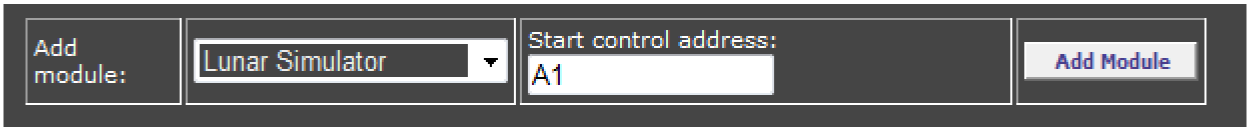
Each module on the control bus must have a unique address. Each address is comprised of a letter and a number. You set these addresses one time using tiny switches called a 'dip switch'. There are (4) switches on the LunarSim module. The first (2) switches establish the letter, the second (2) switches establish the number. A switch can either be UP/OFF or DOWN/ON. If this is your first LunarSim module, select the switch settings for 'A1'.

	Switch 1	Switch 2	Value
First LunarSim	UP	UP	A
Second	DOWN	UP	B
Third	UP	DOWN	C
Fourth	DOWN	DOWN	D

Switch 3	Switch 4	Value
UP	UP	1
DOWN	UP	2
UP	DOWN	3
DOWN	DOWN	4

The control bus address now must be entered in the Apex when you add the module. Remember, these control bus devices are not "plug and play" like the AquaBus devices are. You must add the module in the Apex and assign addresses manually.

From the web pages, select *Configuration – Module Setup*. Mid way down the page, select the Lunar Simulator from the drop down box and enter the control address you selected above. Then click on *Add Module*. You should now see your Lunar Simulator outlet named 'Lunar_A1' (or whatever address you selected above and input below) in the Status screen.



Add module: Lunar Simulator

Start control address: A1

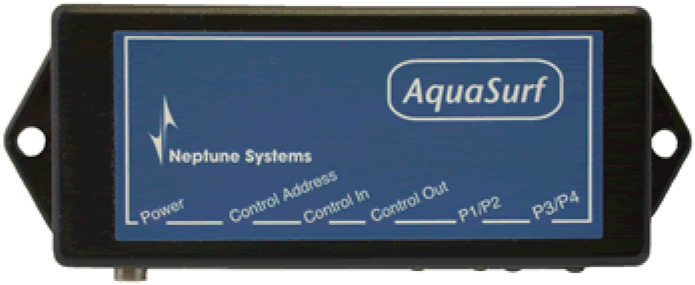
Connection Steps:

(1) Set the LunarSim control bus address using the dip switches
(2) Plug the RJ11 cable into the base unit and the LunarSim module
(3) Add the LunarSim module via the web pages *Configuration – Module Setup*
(4) Plug your LED lights into the LED jack on the LunarSim module

(5) Program the new LunarSim outlet (will be called “Lunar_A1”)

AquaSurf Module

The AquaSurf module is the predecessor to the (4) variable speed ports that come built into the Apex (not the Lite) base unit. However, unlike the Apex, the AquaSurf module only supports the PUMP function – it does not support the RAMP/ dimming function that the Apex does (for more on this, see the section on *Variable Speed Pumps and Dimmable Lights* in Chapter 6). Note below in the photo that the jacks are labeled P1/P2 and P3/P4 but on the Apex they’re labeled V1/V2 and V3/V4 to signify the difference. So keep that in mind if you’re thinking about purchasing the Lite and adding the AquaSurf at a later time. It does not give you the equivalent functionality of the Apex variable speed outlets.



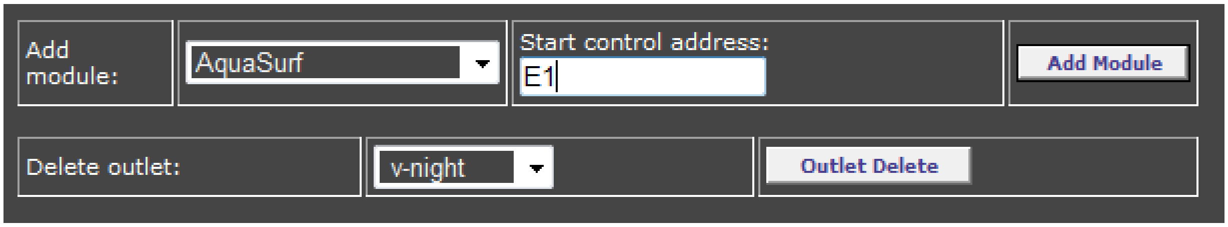
Like the LunarSim module, the AquaSurf module is a ‘control’ type module. It uses an RJ11 phone style cord (many phone cords only have (2) wires. The control bus requires (4) wires) to connect to the base unit’s control port. Connect the other end of the cable to the AquaSurf ‘Control In’ jack. Because it’s a control type module you have to manually set its address on the control bus. If you don’t do this correctly and assign an address that’s already in use, this module won’t work and in all likelihood the other module won’t either – so take a couple of minutes here and understand what you’re doing.

Each module on the control bus must have a unique address. Each address is comprised of a letter and a number. You set these addresses one time using tiny switches called a ‘dip switch’. There are (2) switches on the AquaSurf module. These establish the bus address. A switch can either be UP/OFF or DOWN/ON. If this is your first AquaSurf module, select the E1-E4 switch settings.

	Switch 1	Switch 2	Value
First AquaSurf	UP	UP	E1 – E4
Second	DOWN	UP	F1 – F4
Third	UP	DOWN	G1 – G4
Fourth	DOWN	DOWN	H1 – H4

The control bus address now must be entered in the Apex when you add the module. Remember, these control bus devices are not “plug and play” like the AquaBus devices are. You must add the module in the Apex and assign addresses manually.

From the web pages, select *Configuration – Module Setup*. Mid way down the page, select the AquaSurf from the drop down box and enter the control address you selected above. Then click on *Add Module*. You should now see your AquaSurf outlet named ‘Surf1_E1’ (or whatever address you selected above and input below) in the Status screen.



Add module: AquaSurf

Start control address: E1

Delete outlet: v-night

Connection Steps:

(1) Set the AquaSurf control bus address using the dip switches

(2) Plug the RJ11 cable into the base unit and the AquaSurf module
(3) Add the AquaSurf module via the web pages *Configuration – Module Setup*
(4) Plug your pump cables into the P1/P2 or P3/P4 jacks on the AquaSurf module
(5) Program the new AquaSurf outlets (will be called Surf1_E1 through Surf1_E4)

You can add multiple AquaSurf modules by daisy chaining them together via the Control In and Control Out ports. The Apex will support up to (4) AquaSurf modules.

DC8 Direct Connect 8

The DC8 Power Bar is compatible with the Apex and is the predecessor to the EB8 but lacks the ability for Fallback (see Chapter 4, Programming for more information on the Fallback command) and reporting amperage usage like the EB8. Regardless, it is still a viable module especially for anyone upgrading from one of the AC controllers. Each outlet is capable of up to 6 amps, outlets 1-3 & 5-7 are TRIAC outlets just like the EB8. Outlets 4 & 8 are mechanical relay outlets, again just like the EB8.

The DC8 is a control bus module and as such, has dip switches to set its bus address. There are (4) switches on the DC8 but only the first 3 are used. You can have (2) DC8 modules, the first with outlets 1-8, the second with outlets 9-16. As with other control bus modules, the switch settings are UP/OFF, DOWN/ON.

Switch 1	Switch 2	Value		Switch 3	Outlets
UP	UP	A		UP	1 - 8
DOWN	UP	B		DOWN	9 - 16
UP	DOWN	C			
DOWN	DOWN	D			

The control bus address now must be entered in the Apex when you add the module. Remember, these control bus devices are not “plug and play” like the AquaBus devices are. You must add the module in the Apex and assign addresses manually.

From the web pages, select *Configuration – Module Setup*. Mid way down the page, select the DC8 from the drop down box and enter the control address you selected above. Then click on *Add Module*. You should now see your new outlets along with their default names and programs. You can change both the outlet name and program statements just like the EB8 (see Chapter 4 – Programming for more information on outlet naming and programming).

PM1 Probe Expansion Module

The PM series of Probe Expansion Modules (PM1, PM2 & PM3) are the first accessory modules designed specifically for the Apex and Apex Lite controllers. They use the AquaBus architecture which enables true 'plug and play' setup and up to 200' for the total bus length. Gone are the days of switch setting and adding modules. Plug it in and you're ready to start programming.

The PM1 gives you (1) Temperature port, (1) BNC pH/ORP port and a DIN-8 connector for an I/O Breakout Box which will support up to (6) dry-contact switches. The pH/ORP port is configurable for one or the other (pH or ORP), not both. The probe names will follow the same syntax as other expansion modules: **[PROBE] [EXPANSION MODULE] [PROBE NUMBER].** Probe = Temp, pH, ORP; expansion module = blank for the base unit then A, B, C, D, etc; probe number is 1 or 2 and is only used for the pH probe where you can have multiple probes on one expansion unit.

Connection Steps:

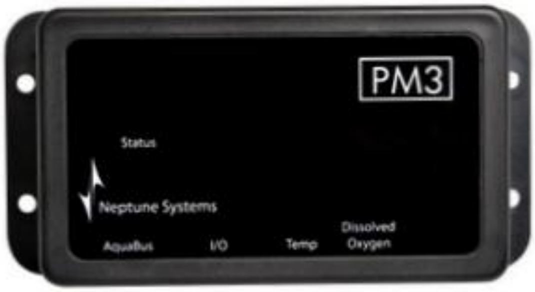
(1) Plug it in. Seriously. That's it.
(2) OK, make sure it's enabled in *System – Prob Exp*
(3) Enable probe(s) via *Setup – pH Setup – pH Enable*. Repeat for remaining probes.
(4) Calibrate it if it's a pH probe

PM2 Probe Expansion Module

Also an AquaBus module, the PM2 gives you (1) Temperature port, (1) DIN-4 Conductivity (salinity) port and a DIN-8 connector for an I/O Breakout Box which will support up to (6) dry-contact switches. The probe names will follow the same syntax as other expansion modules: **[PROBE] [EXPANSION MODULE] [PROBE NUMBER].** Probe = Temp, pH, ORP; expansion module = blank for the base unit then A, B, C, D, etc; probe number is 1 or 2 and is only used for the pH probe where you can have multiple probes on one expansion unit

PM3 Probe Expansion Module

Also an AquaBus module, the PM3 gives you (1) Temperature port, (1) DIN-6 Dissolved Oxygen port and a DIN-8 connector for an I/O Breakout Box which will support up to (6) dry-contact switches. The probe names will follow the same syntax as other expansion modules: **[PROBE] [EXPANSION MODULE] [PROBE NUMBER].** Probe = Temp, pH, ORP; expansion module = blank for the base unit then A, B, C, D, etc; probe number is 1 or 2 and is only used for the pH probe where you can have multiple probes on one expansion unit.



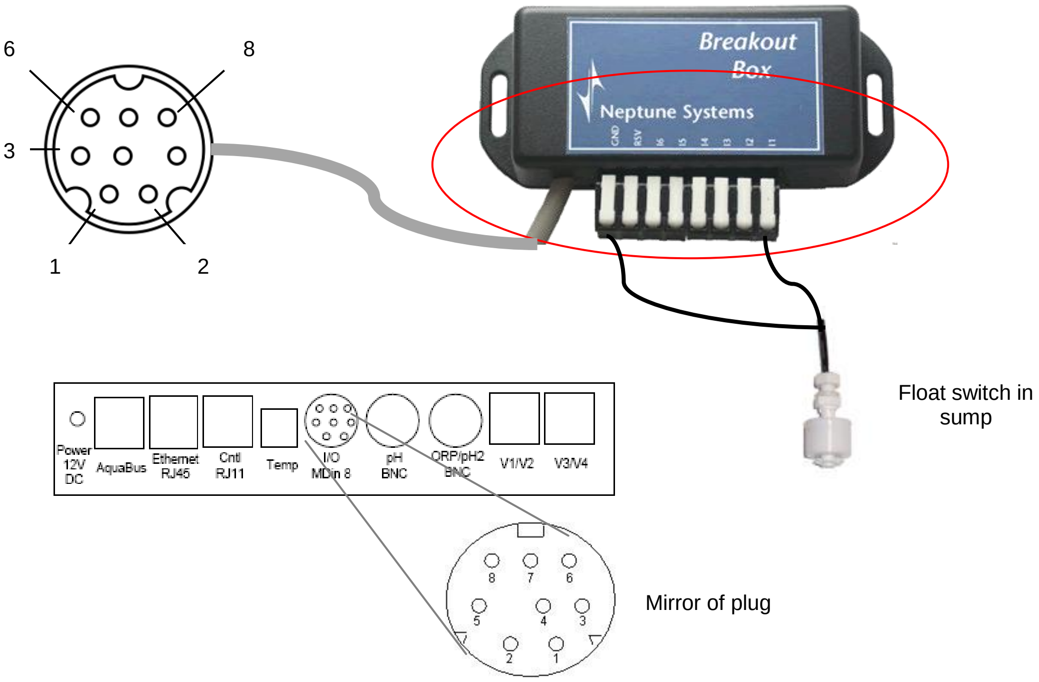
WXM Wireless Expansion Module

Also an AquaBus module, the WXM is designed specifically for EcoTech VorTech line of wireless power heads like the MP-10w and MP-40w. There are (2) AquaBus connectors but no port or DIN-8 connectors on a WXM.

Essentially replacing the EcoTech wireless driver, the WXM will control up to 11 VorTech wireless pumps and supports all pump modes (Constant, Reef Crest, Lagoon Random, Nutrient Transport, Tidal Swell, Pulse, Sync and Anti-Sync). If you need to control more than 11 pumps you can add additional WXM modules.

Since it's also fully Apex compatible you can activate pump modes with any profile capable command including Time, Day of Week, Seasonal, etc. (Chapter 6 has a discussion of profiles in the section on Variable Speed Pumps). The module is fully EcoTech firmware upgradable and will function as a master pump to upgrade your VorTech pumps.

I/O Breakout Box and Switches



The above pictures illustrates an application of the I/O breakout box (BB) from Neptune. The BB has 8 connectors, 6 for switches, 1 reserved and a Ground. Shown connected to it is a float switch with 2 wires. One wire goes to ground along with other ground wires from other switches connected to this BB. The other wire goes to positions 1 – 6. The BB has a “mini-DIN 8” connector on it which plugs into the Apex Base Module. That’s all that is required to connect up to 6 switches to the Apex.

Any ‘dry contact’ switch (i.e. no voltage) that has ‘normally open’ or ‘normally closed’ functionality will work. Examples are floats, window alarms (use when opening doors or canopy), water sensors, door bells – the list is quite large.

The BB is not a module and as such does not have to be added or setup in the base module. Plug in the cable from the BB to the base module and you can immediately begin using ‘switch1’, ‘switch2, etc. in your program

The most common usage of switches is for floats. Floats are referred to as being ‘normally open’ (NO) or ‘normally closed’ (NC). This is a direct result of the orientation of the float to the internal switch contacts. A normally open float means the contacts are open and when the float moves up or down, the contacts close. The reverse is true with normally closed floats. It is this OPEN or CLOSED that you use in programming for a switch. Some floats are reversible. They come in a particular orientation, like normally open but by removing a clip and reversing the float position it becomes normally closed. Other floats you have to purchase in the proper orientation and cannot change.

The reason the orientation is important is for failures. If/when a float switch fails, more likely than not it will fail in the open position (float sticks, wire breaks, contacts don’t close, etc). In addition, when you disconnect the BB from the Apex base module, all switches go to their OPEN state. So keep this in mind when you layout your floats. You want the OPEN condition to trigger stopping something, like a pump. You don’t want OPEN to trigger a pump starting or you could have a nasty surprise some day.

AquaHub has some good information on their website about how float switches work:

http://www.aquahub.com/store/howfloatsw.html

Switch syntax:

If [SWITCH NAME] [OPEN/CLOSED] Then [ON/OFF/PROFILE]

Including a switch in an outlet's program would look like this:

```
Fallback OFF
Set OFF
If Switch1 CLOSED Then ON
If Switch2 OPEN Then OFF
Min Time 10:00 Then OFF
```

Switch 1 (NO) is in the sump and triggers the ATO pump to start when it CLOSES, signifying a drop in the sump water level. Switch 2 is in the ATO reservoir and if OPEN indicates insufficient top off water is available. The Min Time prevents the process from cycling again too quickly which would lead to premature pump failure (you could also use the Defer command to achieve the same result a different way). If either switch1 or switch2 fail in their OPEN position, the ATO process will not occur and your tank is protected (you could also add the switch2 OPEN condition to your email alarm program to notify you that your reservoir is empty).

Another device known to work with the Apex and a BB is the Waterbug water sensor (WB-200). This device requires its own mini-controller which will handle up to 4 Waterbug devices. The controller attaches directly to the BB which in turn, connects to the Apex. It is recognized as just another switch to the Apex and you can test for it in outlet programs (i.e. turn off pumps, send email, etc.). In fact, you can connect dry contact switch that is normally open or closed to the BB. The contacts can only have 0 – 5V of power – power beyond that will damage the base module.

There's plenty of information on the ReefCentral Neptune forum about building your own DIY breakout box including sources for the MiniDin 8 cable, connection ideas, etc. A quick search for "breakout" will return hours of reading. Be careful of wire colors if you chose to DIY. Colors vary between cables and are not standardized. It's best to test each pin with a continuity meter.

Switch Naming Convention

With various expansion modules, you can connect dozens of switches to the Apex system. Naming them can be a bit confusing however because the naming convention is different between switches connected to the base unit, switches connected to 'control' type expansion modules (PX1000, LunarSim, AquaSurf) and switches connected to AquaBus type expansion modules (PM1, PM2, PM3). The following chart details these differences.

Connect I/O Breakout Box to:	Switch Naming Convention
Base Module	Switch1 – Switch6
PX1000 AquaSurf LunarSim	SwitchA1 – SwitchA6 (1)
PM1 PM2 PM3	Switchx5_1 – Switchx5_6 (2)

(1) 'A' is for the first expansion module, 'B' the second, 'C' the third, etc.

(2) ‘5’ varies based on that expansion module’s AquaBus number. If your PM1 is at AB #4, then the switches are named ‘switchx4_1’ through ‘switchx4_6’. If your PM1 is at AB #7, then the switches are named ‘switchx7_1’ through ‘switchx7_6’.

Display Switch Status

Currently you cannot display the status of switches. To get around this limitation, create a virtual outlet (see Chapter 8) specifically for the switch you want to monitor the status of. Name it something appropriate and program it such that ‘If Switch1 OPEN Then ON’. Then display the status of that outlet on the status screen.

Outlet that turns off and stays off

Here’s an example that Todd (swearint) and Jon (shwaggs) came up with. Suppose you want a switch that when tripped, will turn off a pump but then not turn it back on until you do something about it. The first part is easy but how do you keep it off AND make it somewhat easy to turn back on?

```
Fallback OFF
If FeedD Then ON
If Switch1 OPEN Then OFF
```

The above program turns the outlet OFF when the switch OPENS. That’s the easy part. To get it turned back on but only when you want, the ‘Feed’ statement is added. When you select ‘FeedD’ from the display module or web page, the outlet will turn on. FeedD will eventually expire (based on the ‘*feed interval’* you defined) but the outlet will not turn back OFF again until the switch opens the next time.

Push Button Switch Activation

Here’s a great solution that ‘Slief” conceptualized and Todd (‘swearint’) help solve. The requirement was for a push button that would activate a switch that in turn, would turn on a mixing pump instead of activating it via the status menu. Further, the desire was for the mixing pump to run for 60 minutes then automatically turn off. When not activated by the push button, we still want the pump to run for 10 minutes every 3 hours.

The push-button is a simple doorbell that you can get at Agent Orange or any other home improvement or hardware store. Wire it up just like you would a float switch. Then, create a virtual outlet (see Chapter 8 for more information) called ‘HOLD’ with the following program:

```
Set OFF
If Switch5 CLOSED Then ON
Defer 060:00 Then OFF
```

The doorbell is switch5. When you press it, the switch momentarily closes activating the virtual outlet ‘HOLD’. The Defer command keeps the virtual outlet in this state for 60 minutes.

Here’s the program for the physical outlet that controls the pump.

```
Fallback OFF
OSC 000:00/010:00/170 Then ON
If Outlet Hold = ON Then ON
```

The Oscillate statement is the recurring, all-day pump driver that runs it for 10 minutes out of every 3 hours. The ‘If Outlet’ statement is the push-button statement. If the push-button (doorbell) is ON, then the physical outlet is turned on regardless of where it is in its 10/180 cycle.

Examples of Dry-Contact Switches

Water Sensor switch:

http://www.specialtyalarms.com/site/1313932/product/80-519

Window Switch (use on doors, canopy, etc.)

http://www.smarthome.com/7352/Surface-Mount-Miniature-Magnetic-Contact-Switch-Normally-Open/p.aspx

---

Floats

http://www.aquahub.com/store/index.html

http://autotopoff.com/

Humidity Sensor (for your tank room)

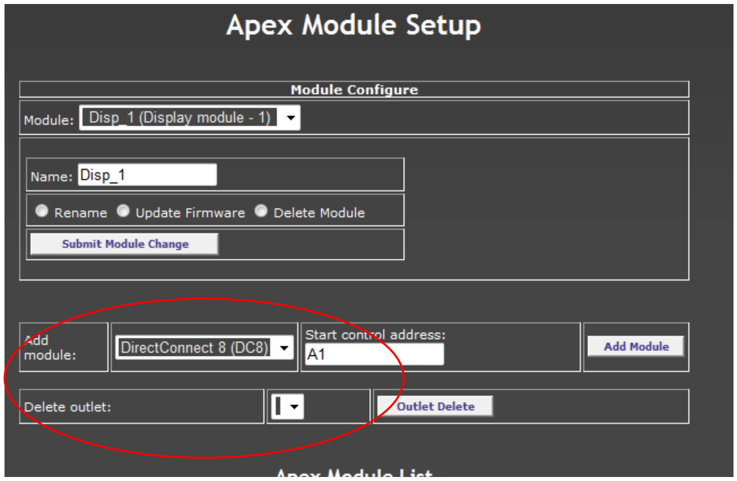
http://www.smarthome.com/7156/Humidity-Sensor-HA-1-1190/p.aspx

Chapter 8: Virtual Outlets

Virtual outlets (frequently referred to by that ignominious term “dummy outlets”) are a powerful tool in constructing complex programming conditions. Without virtual outlets you are limited to single condition ON/OFF statements. However with the ability to define virtual outlets you can combine conditions, similar to the “AND” condition in programming.

Creating a Virtual Outlet

All outlets are created when a module is added to the AquaBus. Virtual outlets are no different – you add a virtual module and get virtual outlets as a result. To add a module, you use the *Configuration → Module Setup* page.



Apex Module Setup

Module Configure

Module: Disp_1 (Display module - 1)

Name: Disp_1

○ Rename ○ Update Firmware ○ Delete Module

Add module: DirectConnect 8 (DC8)

Start control address: A1

Delete outlet:

1. Select the module type you want to add. The type of module you create depends on how many virtual outlets you want to create.
    a. Lunar Simulator – 1
    b. AquaSurf – 4
    c. DC4 – 4
    d. DC8 – 8
2. Once you create the module you will see the new outlets with their default names. Rename the outlets as you wish using the *Configuration - Outlet Setup* page.
3. If you only need 3 outlets for example, you could create a DC4 then either delete the fourth un-used outlet (via the *Configuration - Module Setup - Delete outlet* function) or simply rename it something for later use. I like to keep a ‘test’ outlet available for trying out program statements.

Using Virtual Outlets

Program a virtual outlet as you would program any other outlet (the Fallback statement is not required and has no effect if used and will get you chastised if found using it!). Typically you’re going to use the ‘Advanced’ method of programming but you’re not restricted to that. Any and all of the program statements available with normal outlets can be used with virtual outlets because as far as the controller is concerned, it is an outlet.

You can reference other virtual outlets in your virtual outlet program. Practically speaking, there’s no limit as to how many outlets you can chain together in program statements. You’re limited by your imagination. Let's look at a couple of examples of how you might use virtual outlets to accomplish what you want.

Modifying Alarms

You've got a sump float that you want an alarm on if it's OPEN (indicating a low water level). But you want to give it a chance to fill and don't want the alarm during the night. Adding a "Defer" and "If Time" statement to your alarm outlet will apply to ALL conditions in that outlet so that won't work.

```
Set OFF
If Temp > 81.5 Then ON
If Temp < 76.0 Then ON
If Switch1 OPEN Then ON
If Time 22:00 to 07:00 Then OFF    ← This "If Time" applies even to the temperature conditions
Defer 010:00 Then ON          ← applies to the outlet as a whole
```

The "Defer" and "If Time" statements in the above example will modify the entire alarm outlet to not activate if the time is 22:00 to 07:00 and only if the entire outlet stays in that state for 10 minutes. But if we create a virtual outlet that's an alarm just for the low sump conditions we want, we can reference that in the main alarm program.

```
Virtual outlet "Low_Sump_Alm"          ← I've created a virtual outlet and renamed it

Set OFF
If Switch1 OPEN Then ON                ← Switch1 is my sump float switch, normally CLOSED
Defer 010:00 Then ON                   ← Has to stay OPEN/ON for 10 minutes
If Time 22:00 to 07:00 Then OFF        ← Virtual alarm OFF at night
```

Now in my main alarm program, adjust it as follows:

```
Set OFF
If Temp > 81.5 Then ON
If Temp < 76.0 Then ON
If Low_Sump_Alm ON Then ON
```

Vary lights by day and time

Here's a good solution (posted by Ken Argo) on how to modify your light schedule for different days and times of the week. In this example, we want the lights ON Monday – Wednesday 10am – 7:15pm, Thursday & Friday 10am – 6:15pm, OFF all other days/time.

Ken creates two virtual outlets called "Lights_TF" and "Lights_S-W". He sets their ON/OFF using both the Time and DoW commands. Then in the real light outlet he tests for the virtual outlets to be ON (you can read about the DoW command in the next chapter).

```
Lights_TF
If Time 10:00 to 20:15 Then ON
If DoW SMTW__S Then OFF

Lights_S-W
If Time 10:00 to 18:15 Then ON
If DoW ____TF_ Then OFF
```

Then for the real light outlet use:

```
Set Off
If Outlet Lights_TF = ON Then ON
If Outlet Lights_S-W = ON Then ON
```

Multiple automatic feed cycles

Here's a solution posted by Russ Michel as a way to automate multiple feed cycles. Recall that the feed cycles can be manually initiated from either the browser or the display. Here the request was to automate and time the feed cycles (shut down pumps, etc.) to coincide with a programmable Eheim feeder. The feeder runs every day at 8am, 1pm and 5pm. We want to turn off the return pump and our skimmer while feeding.

Create a virtual outlet called Auto_Feed with the following program statements:

```
Set ON
If Time 08:00 to 09:00 Then OFF
If Time 13:00 to 14:00 Then OFF
If Time 17:00 to 18:00 Then OFF
```

Now, in each of the outlet programs for your return pump and skimmer, add the following:

```
If Outlet Auto_Feed = OFF Then OFF
```

You could tune the times a little so everything turns off 5 min. before the feeding begins and you can make the duration anything you want but you get the idea.

Notification if selected outlet(s) off for more than 30 minutes

Suppose you had some outlets that are supposed to turn ON/OFF at some regular interval. Further, suppose you wanted to know if those outlets were off more than 30 minutes. Again, a virtual outlet is the answer.

Create a virtual outlet called "Check_me_out"

```
Set OFF
If Outlet Ret_Pump = OFF Then ON
If Outlet Skimmer = OFF Then ON
If Outlet Chiller = OFF Then ON
Defer 030:00 Then ON
```

Now, in your email alarm outlet program add the following:

```
If Outlet Check_me_out = ON Then ON
```

Note that this will only detect if the Outlet is OFF. If the Outlet is ON but the device itself has failed for some reason and is off, this will not detect that condition.

Custom Maintenance Cycle

By now you know that you can use the Feed cycles for maintenance as well as feeding. They're limited to 999 minutes which is pretty long but if you cancel the cycle early, everything starts immediately. If you have your skimmer turned off as part of the maintenance cycle, having it turn on immediately will probably cause an overflow until the sump level stabilizes. Here's how you can create a custom maintenance cycle using virtual outlets that when cancelled will still delay your skimmer from starting.

Create a virtual outlet called "Maint".

```
Fallback OFF
Set OFF
```

Now, for each device you want turned on/off when in maintenance mode (i.e. your pumps), add the following line to their respective outlet programs:

```
If Outlet Maint = ON Then OFF  (or ON if for example lights in your sump)
```

For the skimmer however, you need three lines of code

```
If Outlet Maint = ON Then OFF
If Outlet Maint = OFF Then ON
Defer 005:00 Then ON
```

The last 2 lines tell the Apex that when the Maint outlet goes to OFF, turn the skimmer ON but only after waiting (Defer) for 5 minutes.

To operate this maintenance outlet, use the Status page or the display module to manually turn the outlet OFF. Everything shuts down that you want shut down. When you're done, set the outlet status to ON. All devices will start back up EXCEPT the skimmer which will wait an additional 5 minutes.

The above examples should serve to illustrate the kinds of things you can do with virtual outlets.   You might already be thinking of different uses or even different ways to solve the above challenges. The Apex programming language is powerful once you learn how to manipulate it.

Chapter 9: Logs and Logging

The Apex has built in data capture logs that record system activity. Some of the logs capture data on a specified interval. Other logs are transactional and capture data when an event occurs, like the state of an outlet changing. Log data is stored in XML format.

Beginning with version 4.02 the method of interpreting and capturing time was changed. Prior to this, all time, whether that was display time or log time, was in your local time that you input in the clock setup screen. Version 4.02 changed all of that. Now, time is stored in UTC (or GMT) time. It is still displayed in local time but in order to convert from UTC to local the controller needs to know what time zone you're in and whether daylight savings time applies or not (see the *Clock Setup* section of Chapter 2 for more information).

Every time you install a new firmware release the controller will start new data logs. Typically this doesn't cause any issues but when moving to release 4.02 what happens is you wind up with logs that store time in different formats. And not only is the time different. Since UTC is 'ahead' of the United States by anywhere from 5 (New York) to as much as 10 hours (Hawaii), the date that's stored in the log can actually be tomorrow for data you're recording right now! How weird is that? And to compound this, 4.02 introduced 24hr graphing meaning that when you look at a graph for today you actually see data from this time yesterday through now. So, take yesterday's data in one time reference and today's data in another and for a period as much as two days you're going to get really strange graphs. Eventually the logs with the time data in the old format will be removed and replaced by newer logs.

Log Names

- **Outlog** – every time the Apex changes the state of an outlet a record is written to this log. If no outlet ever changed state you would have zero records in this log. A new log is started daily at midnight. Log is named "yymmdd.odat".

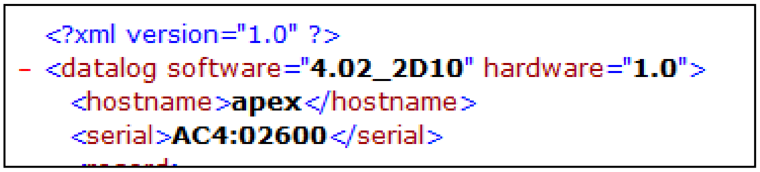
- **Datalog** – records probe value snapshots (Temp, pH, ORP, etc.) based on the logging interval you define (default = 20 minutes). You can change the interval via the *Display* module under *Data Log – Log Interval.* A new log is started daily at midnight. Log is named "yymmdd.pdat".

- **Program** – XML report of your current program statements by outlet. Runs real-time when you request it. Before version 4.02 and the ability to backup your programs, this was a good report to save somewhere in case you needed to reprogram your outlets.

- **Status** – Really more of a real-time report. It has two sections. The first is a record of the most recent power fail event (could be days old) and the second records the current outlet states and probe values. This is the data that is used for the Status screen.

The beginning of every XML log contains the same data:

- Log name
- Software version (Firmware)
- Hardware version
- Base unit serial number
- Date (Status and Program logs only)



```
  <?xml version="1.0" ?>
- <datalog software="4.02_2D10" hardware="1.0">
    <hostname>apex</hostname>
    <serial>AC4:02600</serial>
```

The Apex has the ability to store up to a year's worth of log data under normal polling frequency of every 20 minutes and 3 probes. If you have more probes or you poll more frequently, your log files will be commensurately larger. The controller will automatically remove the oldest log to make space if needed. Also, any time you load new firmware, add a module or a probe, new logs will be generated.

Beginning with firmware 4.01 you can view previous day's data and even multiple days of data in a single query. Using the following syntax from your browser, you can view a previous day:

http://<your IP address or DynDNS name>:<port#>/cgi-bin/outlog.xml?sdate=yymmddhhmm

For multiple days, use the above syntax and append as follows:

http://<your IP address or DynDNS name>:<port#>/cgi-bin/outlog.xml?sdate=yymmddhhmm&days=n

Here are two examples:

> http://192.168.0.100:5560/cgi-bin/outlog.xml?sdate=100320&days=7 (if local)
> http://mytank.homedns.com:5560/cgi-bin/outlog.xml?sdate=100320 (if remote)

The above works for the datalog and outlog – just substitute the log name in the above example. Note that you would use your IP address for internal (i.e. your home network) access and the DynDNS name for external (i.e. web) access. Also, the hours/minutes are optional in the date parameter.

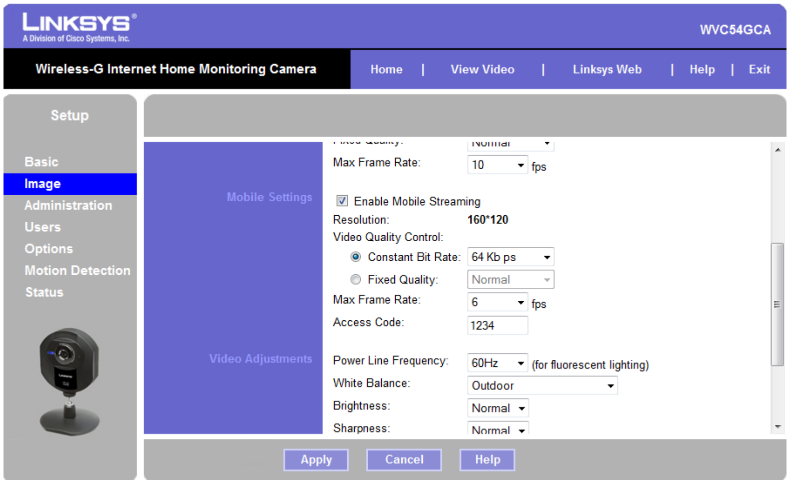
Chapter 10: Third Party Accessories

Web Camera

It's quite easy to setup a web camera so you can keep an eye on your tank while you are away, either via the web or through a smart phone like a Blackberry. One very easy to setup and inexpensive camera is the Linksys WVC54GCA. It is wireless and includes its own built-in web server. It can be found on the web for about $100. You don't need a computer to operate it, only a network connection through your wireless router and you can place it anywhere in your home as long as you have power nearby. To access it remotely over the web you will have to setup a static IP address and enable port forwarding (see Configure Internet Access) just like with the Apex.

One advantage of the Linksys is it supports the RTSP protocol (Real Time Streaming Protocol). This enables you to view streaming images with your smart phone. Not all smart phones support RTSP but the Blackberrys with O/S version 4 and up, do. Here are the steps I took to configure my camera and router – note that for remote access to the camera software and real time streaming, we have to define three ports; HTTP, RTP and RTSP. Also, this tutorial assumes that you've installed the camera on your network and you've established both local as well as Internet access to the camera. The instructions included with the camera and available at the Linksys website will walk you through the process.

1. Log into the camera, under the *IMAGE* menu scroll down to *Mobile Settings* and select "*Enable Mobile Streaming*". Select 32Kbps for Bit Rate, and 6 frames or less for Frame Rate. Enter in an Access Code like 1234



Max Frame Rate: 10 fps

☑ Enable Mobile Streaming

Resolution: 160*120

Video Quality Control:

● Constant Bit Rate: 64 Kb ps

○ Fixed Quality: Normal

Max Frame Rate: 6 fps

Access Code: 1234

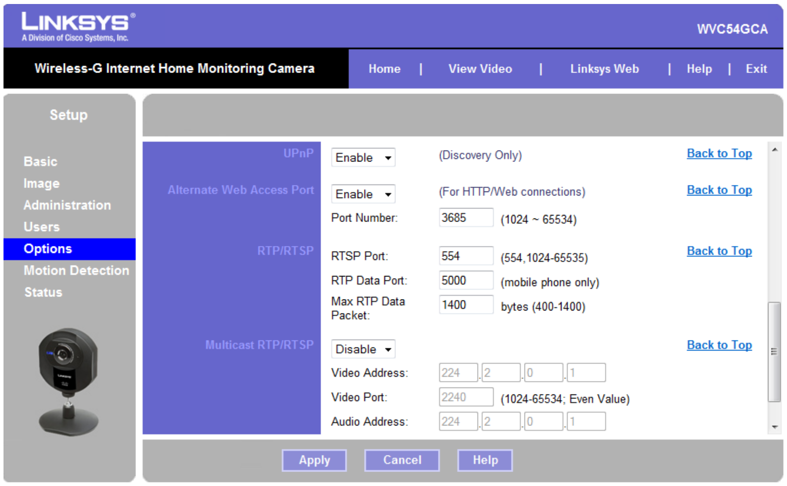
Power Line Frequency: 60Hz (for fluorescent lighting)

White Balance: Outdoor

Brightness: Normal

Sharpness: Normal

2. Go to the *OPTIONS* tab and check that the RTSP port and RTP Data Port are set properly. I used a port of 3685 for the camera's HTTP (don't ask me why, it just sounded good) and the default of 554 for RTSP and 5000 for RTP. I set the packets to their highest of 1400 but you can tweak this once you get it working. Note that you would have set the HTTP port when you established port forwarding to give you Internet access to the camera.


LINKSYS®
A Division of Cisco Systems, Inc.
WVC54GCA
Wireless-G Internet Home Monitoring Camera
Home | View Video | Linksys Web | Help | Exit
Setup
Basic
Image
Administration
Users
Options
Motion Detection
Status
Back to Top
Apply
Cancel
Help

UPnP: Enable (Discovery Only)

Alternate Web Access Port: Enable (For HTTP/Web connections)
Port Number: 3685 (1024 ~ 65534)

RTP/RTSP
RTSP Port: 554 (554,1024-65535)
RTP Data Port: 5000 (mobile phone only)
Max RTP Data Packet: 1400 bytes (400-1400)

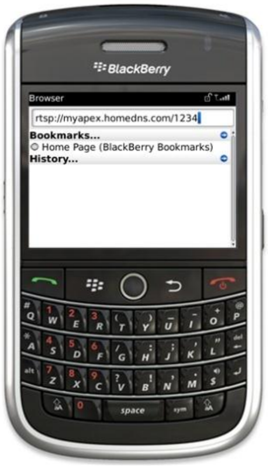
Multicast RTP/RTSP: Disable
Video Address: 224 . 2 . 0 . 1
Video Port: 2240 (1024-65534; Even Value)
Audio Address: 224 . 2 . 0 . 1

3. Refer to the Networking chapter for Domain Name Services (DNS). If you haven't setup a static address for your router with a domain name service like DynDNS, you'll need to do that now. If you have, you can use the one you created when configuring the Apex for Internet access. Again, this step has probably already been completed but is listed here for clarity.

4. From your PC, using Apple QuickTime or Real's RealPlayer, try to access your camera. When viewing internally, use the Internal IP address of the camera, NOT the DynDNS domain name. If your camera's IP was 192.168.0.105 you would enter in Quick Time's URL field (note that you must remove the http and replace it with RTSP): rtsp://192.168.0.105/1234 (or whatever access code you defined in #1). Note that this is different than using the camera's software viewing utility (which runs via HTTP). In this step, we're testing that video streaming using the RTSP protocol works which is why we're using QuickTime or RealPlayer instead of the camera's software.

Screenshot of QuickTime streaming video

5. In order to view your camera remotely via RTSP, you'll need to forward the RTP and RTSP ports to your camera. Log into your router and forward ports 554 RTSP and 5000 RTP to the IP address of your camera. You also need to forward the HTTP port of the camera itself – in this case we selected 3685 in step #2. All three need to forward to the IP address of the camera. Refer to this screenshot of the D-Link router port forwarding screen. Again, as mentioned in step #3 the forwarding of the camera's HTTP port has probably already been done if you have Internet access to the camera's internal software viewing utility but is mentioned here for clarity.


BlackBerry
Browser
rtsp://myapex.homedns.com/1234
Bookmarks...
Home Page (BlackBerry Bookmarks)
History...

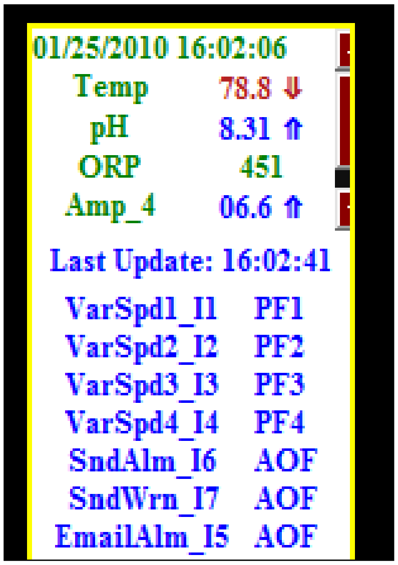
6. Open up your mobile device and go to its web browser. Enter in the camera URL, keeping in mind that you need to delete the http:// and enter rtsp://. Using the domain name from DynDNS, you would enter: rtsp://myDynDNShostname/accesscode. The example for the Apex from the Networking chapter was "myapex.homedns.com" so you would enter "rtsp://myapex.homedns.com/1234". This takes a few seconds and comes back with a "Buffering" message. A moment or two later, your video should start streaming. Take a moment and compare this URL to the URL you used in step #4 for internal access. In that example, you used the internal IP address of the camera. Here, you use the external or DynDNS name for the camera along with the RTSP at the front and the access code bringing up the rear.

7. If it doesn't work, try going to rtsp://M.Youtube.com. This is YouTube's RTSP server and if it works here then you have a configuration issue with your home network or camera, Check your settings and try again.

If you are using a corporate Blackberry that runs on your company's Blackberry Enterprise Server (BES), your IT department might be blocking the ports necessary for RTSP and RTP (554 and 5000) or the RTSP protocol itself. If you can view your camera's RTSP feed from a home PC (using QuickTime or RealPlayer) but can't access it on your Blackberry, this could be the cause. You'll have to figure out how to convince them to open up those ports in your corporate firewall (show them pictures of your clown fish – who could resist?).

The above instructions have only been tested on the default Blackberry browser. I've not tried these using Opera, Bolt or other third party smart phone browsers. If you're using one of these and have trouble, go back to the Blackberry browser and try from there.

Vista and Windows 7 desktop gadget (By Ken Argo)



01/25/2010 16:02:06

Temp	78.8 ⇓
pH	8.31 ⇑
ORP	451
Amp_4	06.6 ⇑

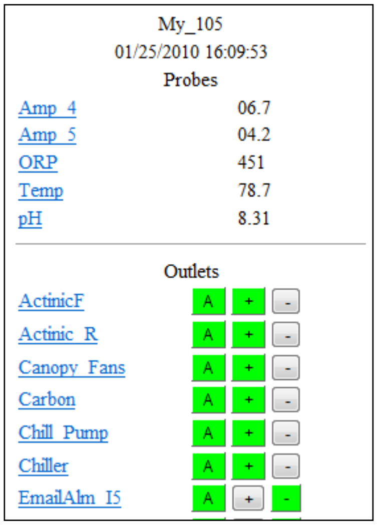
Last Update: 16:02:41

VarSpd1_I1	PF1
VarSpd2_I2	PF2
VarSpd3_I3	PF3
VarSpd4_I4	PF4
SndAlm_I6	AOF
SndWrn_I7	AOF
EmailAlm_I5	AOF

This free desktop gadget for Vista and Windows 7 (sorry XP users, no gadgets for you) presents your controller status page in a desktop widget (www.kenargo.com/software). It also maintains a database of outlet and sensor data, polling your tank on an interval that you define. This is a read-only application. You can't control your tank from this gadget but while you're at work you can keep an eye on your tank conditions. Contact Ken for more information.

Tank Manager (By Ken Argo)

Tank Manager (www.kenargo.com/TankManager) is a web based controller application for Blackberry, iPhones and other intelligent PDA's. It has functions to emulate the Apex status screen enabling you to turn outlets on/off as you would with the desktop application. In addtion you can graph probe and outlet data. With Tank Manager your controller is always with you.



My_105
01/25/2010 16:09:53
Probes

Amp_4	06.7
Amp_5	04.2
ORP	451
Temp	78.7
pH	8.31

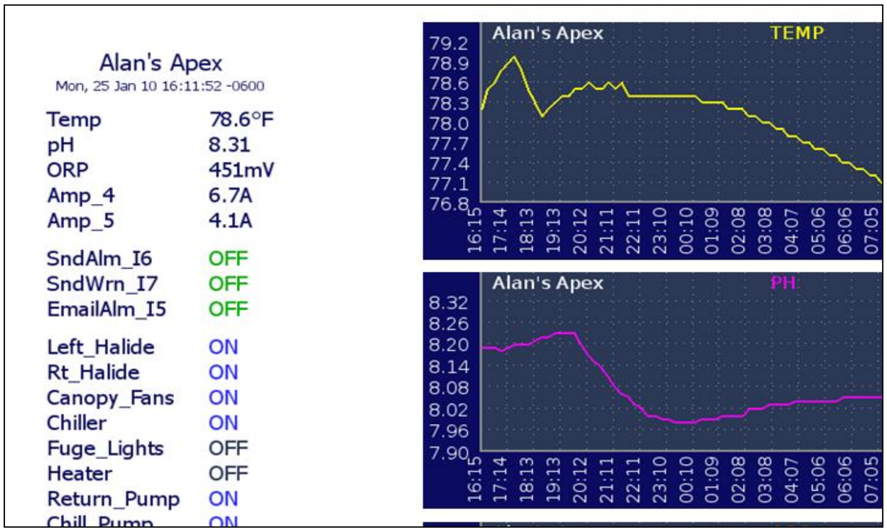
Outlets

ActinicF	A	+	-
Actinic_R	A	+	-
Canopy_Fans	A	+	-
Carbon	A	+	-
Chill_Pump	A	+	-
Chiller	A	+	-
EmailAlm_I5	A	+	-

Tank Manager also includes the ability to program your outlets. What makes Tank Manager unique in this regard is you can see all your outlets at one time and you can make program changes from your mobile device, something you can't do with the Neptune web application.

Ken also has applications for Windows Mobile Standard and Professional. Contact Ken for more information.

Reeftronics (By Russ Michel)



Alan's Apex

Mon, 25 Jan 10 16:11:52 -0600

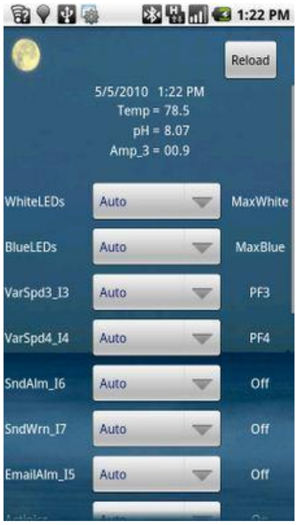
Temp	78.6°F
pH	8.31
ORP	451mV
Amp_4	6.7A
Amp_5	4.1A
SndAlm_I6	OFF
SndWrn_I7	OFF
EmailAlm_I5	OFF
Left_Halide	ON
Rt_Halide	ON
Canopy_Fans	ON
Chiller	ON
Fuge_Lights	OFF
Heater	OFF
Return_Pump	ON
Chill_Pump	ON

Reeftronics (www.reeftronics.net) is a free service that enables you to publish your tank status to the web as a series of tables and graphs. Reeftronics takes care of all the web setup and hosting. It is even optimized for touch screen devices like the iPhone; Android based phones and the Palm Pre. Every 15 minutes the Reeftronics server will poll your Apex controller and upload statistics to its own database. Then you can view status graphs in 24hr, 7day and 30 day increments. Reeftronics requires very little on your part to setup. Your controller must be accessible from the web and you need to provide your controller userid and password.

Russ has also developed some nice utilities to retrieve and view your log and outlet information. Contact Reeftronics for more information.

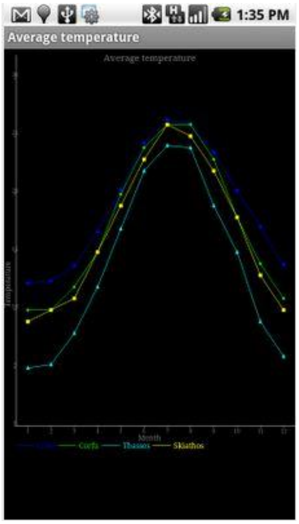
Android Status Viewer (By Joe Flynn)

If you have a mobile device that's running the Android operating system (Motorola BackFlip, Nexus One, Droid Incredible, etc.), there's an application for you. Joe Flynn (HeneryH) has developed a mobile Apex control, status and graphing application specifically for the Android operating system. You can download the latest version via your phone from the Android Market.



5/5/2010 1:22 PM
Temp = 78.5
pH = 8.07
Amp_3 = 00.9

WhiteLEDs	Auto	MaxWhite
BlueLEDs	Auto	MaxBlue
VarSpd3_I3	Auto	PF3
VarSpd4_I4	Auto	PF4
SndAlm_I6	Auto	Off
SndWrn_I7	Auto	Off
EmailAlm_I5	Auto	Off



Chapter 11: Firmware Upgrades

Overview

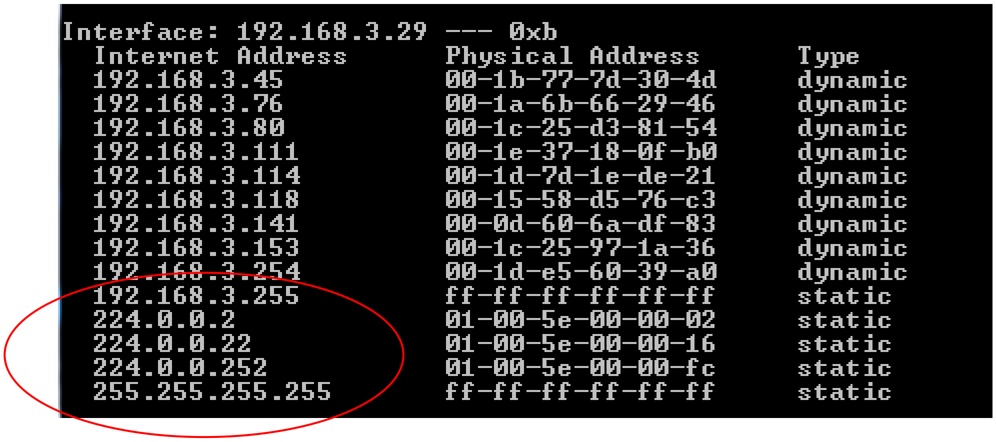
Firmware updates for the Apex happen on a regular basis as new features are introduced and issues addressed. There's a couple of things to remember about applying firmware updates that may make life easier for you.

Beginning with release 4.02 you can easily backup and restore your Apex's configuration and program so you might want to do this before applying any new firmware updates. The update process is not supposed to delete your configuration but it doesn't hurt to have a backup first in case something goes horribly wrong.

The Apex controller is very sensitive to networks that have multiple adapters with different address ranges. The reason is that the utility loader has to create some temporary addresses on the network. Having something like VPN software for work or a game controller will confuse the loader, preventing it from properly creating the required addresses. So you're going to want to disable all of that. Wireless is also a problem for the same reason. Disable your PC wireless adapter. If you have your base module running off its own wireless adapter, you must plug it directly into your router to load the firmware or take a laptop to the base unit and connect it (see ***Connecting directly to your PC/Laptop***) .

One way to check if your network is ready for an update is to run the "arp -a" command from the DOS command prompt. You should see a single interface listed (that's your PC's network card) with one or more IP addresses listed under it. These are devices on your network. If you have more than one interface listed, you're going to have problems loading the firmware. Neptune is working to make the loader more robust and less susceptible to these situations but for now it's safest to follow these rules.

The interface and every device listed under it should have the same first 3 octets of the IP address. For example, if your interface is listed as 192.168.3.29 then every other device listed in the arp -a command should start 192.168.3. If you have a device listed with different octets, you will not be able to update your firmware (in the following example, note the addresses starting with "224"). You have to figure out what the other adapter is and disable it. Again, some frequent culprits are VPN software and online games.



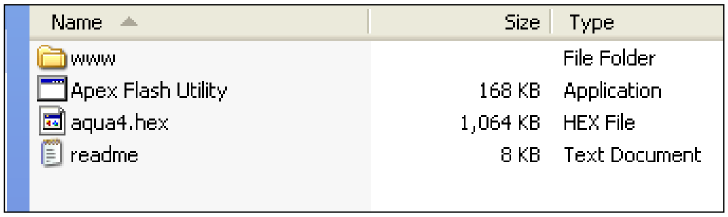
```
Interface: 192.168.3.29 --- 0xb
  Internet Address      Physical Address      Type
  192.168.3.45          00-1b-77-7d-30-4d     dynamic
  192.168.3.76          00-1a-6b-66-29-46     dynamic
  192.168.3.80          00-1c-25-d3-81-54     dynamic
  192.168.3.111         00-1e-37-18-0f-b0     dynamic
  192.168.3.114         00-1d-7d-1e-de-21     dynamic
  192.168.3.118         00-15-58-d5-76-c3     dynamic
  192.168.3.141         00-0d-60-6a-df-83     dynamic
  192.168.3.153         00-1c-25-97-1a-36     dynamic
  192.168.3.254         00-1d-e5-60-39-a0     dynamic
  192.168.3.255         ff-ff-ff-ff-ff-ff     static
  224.0.0.2             01-00-5e-00-00-02     static
  224.0.0.22            01-00-5e-00-00-16     static
  224.0.0.252           01-00-5e-00-00-fc     static
  255.255.255.255       ff-ff-ff-ff-ff-ff     static
```

If your network is clean the loader and the base unit will find each other and establish that initial handshake. The loader will issue a login and password and if successful, an 'update' command to put the base unit into bootloader mode. That's when you'll see the status light on the base unit turn blinking yellow. At that point a temporary IP address will get created for the loader with a new fixed MAC address of '0c:1d:12:e0:1f:10'. If you did another 'arp –a' command at this point you would see this new IP address in the arp listing. This is all just temporary and is required by the firmware update process (Mac users have to add the temporary IP address and MAC address manually, described later). The temporary IP address will disappear when the loader finishes.

One final note about the utility loader. It requires Microsoft .Net Framework version 3.5. You can determine your version of .Net by going to C:/Windows/Microsoft.Net/Framework and looking at the folder names. If you receive an error about "Application failed to initialize properly" or other program errors, that's most likely what is causing it. You can download the most current version of .Net from Microsoft here:

http://www.microsoft.com/downloads/details.aspx?FamilyId=333325FD-AE52-4E35-B531-508D977D32A6&displaylang=en

Execution

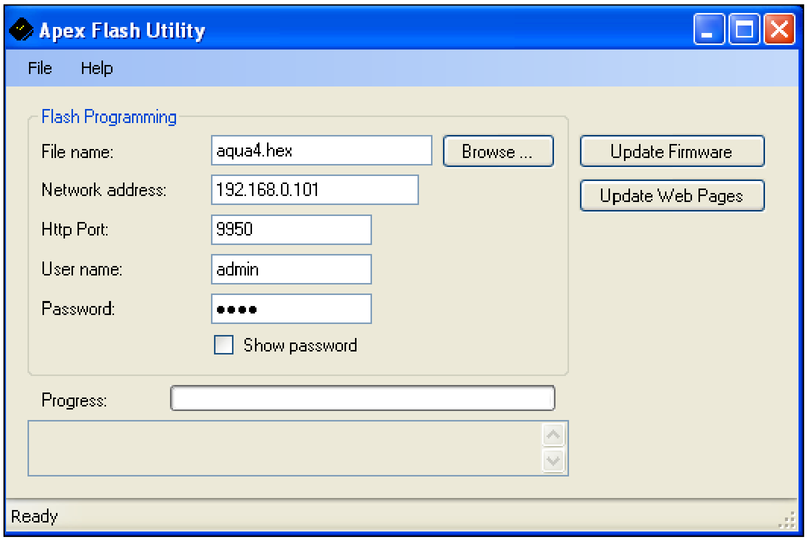


Name	Size	Type
www		File Folder
Apex Flash Utility	168 KB	Application
aqua4.hex	1,064 KB	HEX File
readme	8 KB	Text Document

When Neptune distributes a new software release, there are four components - the loader tool (has a .exe suffix), the new firmware (has a .hex suffix) ,a directory of new HTML and XML files (the directory is called "www") and a readme text file. All of this comes in a zip file that you un-zip to a temporary directory someplace on your PC. Neptune's web site has the latest production release of the software on the *Support* page under *Downloads*. If you want beta releases, you have to monitor the Neptune forum at Reef Central (Tip: search for posts from member "clp"). Neptune will announce beta releases and post links to download them.

There's no installation process - just un-zip the contents to a folder on your desktop.

When you start the loader tool (Apex Flash Utility.exe) you will be presented a screen where you input the name of the firmware file (should be prefilled), your base module's IP address (no slash at the end), HTTP port as well as your userid and password.



File name: aqua4.hex

Network address: 192.168.0.101

Http Port: 9950

User name: admin

Password: ●●●●

□ Show password

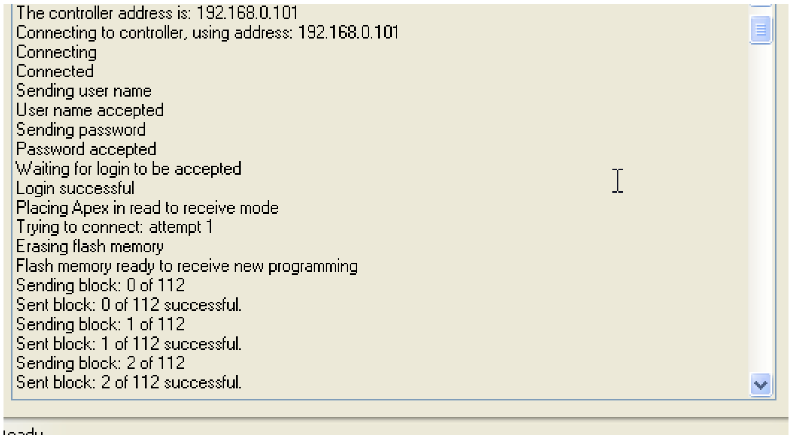
Progress:

Ready

The HTTP Port is only used by the '*Update Web Pages*' function. If you have trouble loading the web pages but successfully loaded the firmware, this is one thing to double check.

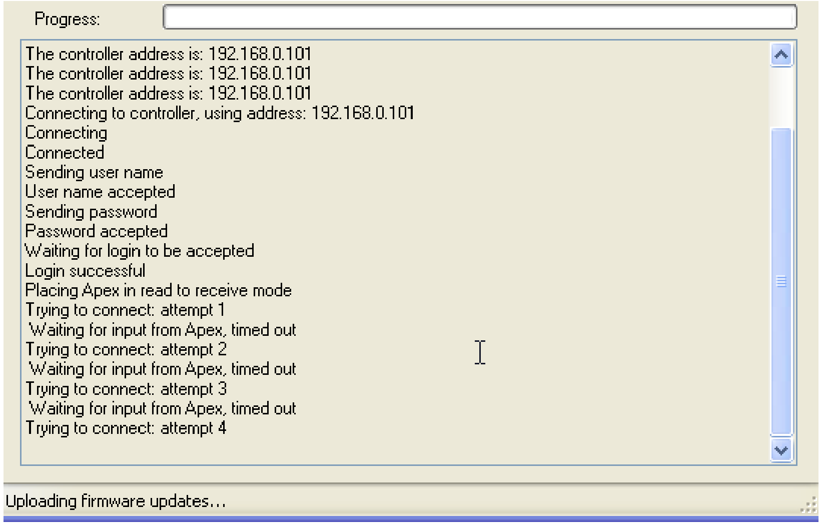
If your base module successfully establishes the login handshake and update command it will respond by placing itself in bootloader mode. You'll know this because the base module status light will change from its normal solid green to a blinking yellow. Within a few seconds you'll see a string of text scroll on the loader tool's screen. After about 10 seconds it's done loading the firmware.

This is what a successful load will look like. Blocks will repeat (scroll) up to 112 in this case. Do not interrupt the process at this point – note the message "Erasing flash memory". At that point and until the load completes, you have no firmware in your unit!



```
The controller address is: 192.168.0.101
Connecting to controller, using address: 192.168.0.101
Connecting
Connected
Sending user name
User name accepted
Sending password
Password accepted
Waiting for login to be accepted
Login successful
Placing Apex in read to receive mode
Trying to connect: attempt 1
Erasing flash memory
Flash memory ready to receive new programming
Sending block: 0 of 112
Sent block: 0 of 112 successful.
Sending block: 1 of 112
Sent block: 1 of 112 successful.
Sending block: 2 of 112
Sent block: 2 of 112 successful.
```

Sometimes putting the base module in update mode doesn't happen and the loader tool will timeout and try again:



```
The controller address is: 192.168.0.101
The controller address is: 192.168.0.101
The controller address is: 192.168.0.101
Connecting to controller, using address: 192.168.0.101
Connecting
Connected
Sending user name
User name accepted
Sending password
Password accepted
Waiting for login to be accepted
Login successful
Placing Apex in read to receive mode
Trying to connect: attempt 1
 Waiting for input from Apex, timed out
Trying to connect: attempt 2
 Waiting for input from Apex, timed out
Trying to connect: attempt 3
 Waiting for input from Apex, timed out
Trying to connect: attempt 4
```

Uploading firmware updates...

More likely than not, you've got a conflict on your network (see above). Disable wireless, disable VPN, disable game adapters, re-run "arp –a".

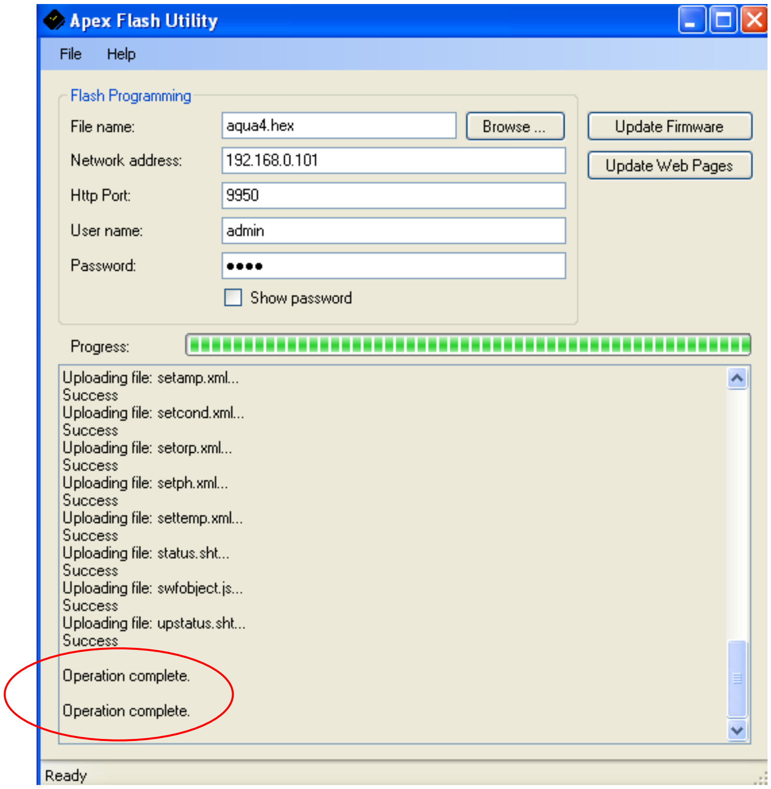
If you still can't successfully get the base unit into bootloader mode and you've disabled everything in your network properties other than your Ethernet adapter, you can sometimes force the base module into bootloader mode:

1. Start the loader app,
2. Select 'update firmware',

3. Let the loader try and find the base module. It won't and will time out, perhaps multiple times.
4. When it does (leave it running), unplug the base module from power and UPS (if you have a UPS),
5. Wait a couple of seconds and plug it back in. It will automatically go into bootloader mode as part of its startup sequence.

Hopefully it will see the loader tool and establish a connection. If the light on the base module turns green without having done the update it means it could not successfully go into loader update mode.

After your firmware is loaded the base module will reboot. This will take 10 – 15s. Be patient and let it complete. Once you have a solid green status light on your base module and EB8, you can load the new web pages. From the same utility loader program, click on "Update Web Pages". The new software for the web pages will load and the display will look like this:



File name: aqua4.hex
Network address: 192.168.0.101
Http Port: 9950
User name: admin
Password: ●●●●
□ Show password

```
Uploading file: setamp.xml...
Success
Uploading file: setcond.xml...
Success
Uploading file: setorp.xml...
Success
Uploading file: setph.xml...
Success
Uploading file: settemp.xml...
Success
Uploading file: status.sht...
Success
Uploading file: swfobject.js...
Success
Uploading file: upstatus.sht...
Success

Operation complete.

Operation complete.
```

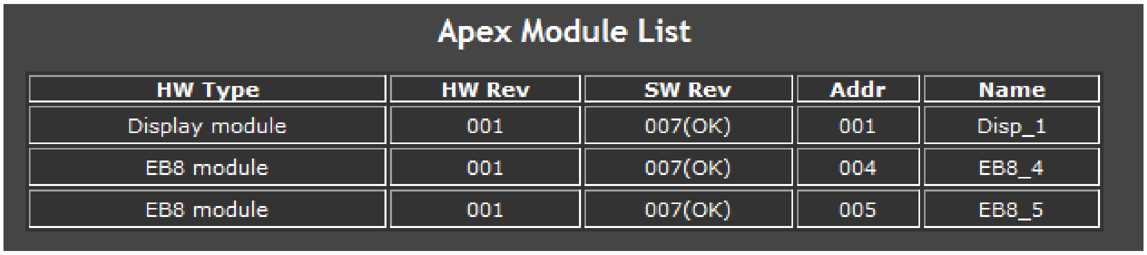
Once you get the "Operation Complete" message you have completed your firmware upgrade. To verify, log into your base module via the browser and run any of the XML reports (data log, outlet, program or status). The first line will indicate your firmware version. Compare to the readme notes in the update.

Effective with version 4.03, you can also see your current firmware version from the Display module by selecting the *Self Test* option.

Module Updates

Periodically Neptune Systems will release upgrades to the sub-set of firmware that runs various AquaBus modules. Every new controller firmware release does not necessarily have module updates. However, the controller will let you know if a module update is required and you should check after every firmware update just to be safe.

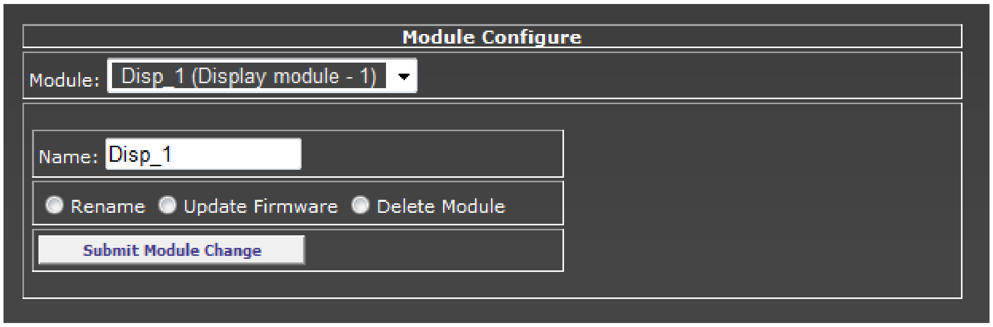
At the bottom of the *Configuration – Module Setup* will be a listing of your AquaBus modules, their software revision and whether an update is required. In the example below, the (OK) indicates that all modules are current and no update is required. If an update was required, (OLD) would be displayed. Note that the software revisions don't all have to be the same number, however all modules of a particular type need to be the same. So, it's possible that your EB8 is one version and your Display module is different. However, if you had multiple EB8's as shown below, all need to be the same revision.



Apex Module List

HW Type	HW Rev	SW Rev	Addr	Name
Display module	001	007(OK)	001	Disp_1
EB8 module	001	007(OK)	004	EB8_4
EB8 module	001	007(OK)	005	EB8_5

To update a module, first apply all applicable firmware and web page updates. Then, using the *Configuration – Module Setup* web page, select the module you want to update. Check the radio button option for '*Update Firmware'*, then click on '*Submit Module Change'*.



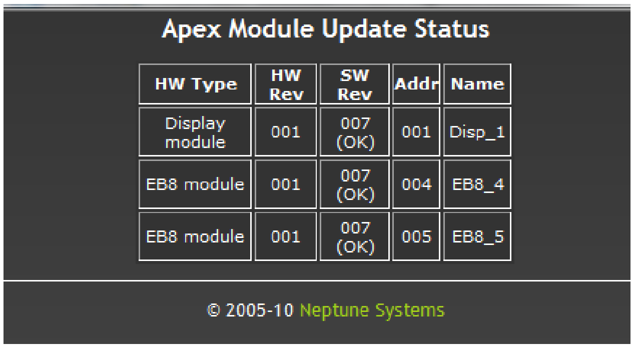
Module Configure

Module: Disp_1 (Display module - 1)

Name: Disp_1

○ Rename ○ Update Firmware ○ Delete Module

A small window will pop up detailing the update process.



Apex Module Update Status

HW Type	HW Rev	SW Rev	Addr	Name
Display module	001	007 (OK)	001	Disp_1
EB8 module	001	007 (OK)	004	EB8_4
EB8 module	001	007 (OK)	005	EB8_5

© 2005-10 Neptune Systems

That's it. All modules are now current.

Sometimes the module update fails to start. There are a couple of things you can try if this happens:

1. Make sure you're running Internet Explorer. Problems have been reported with Safari when doing the module update. IE works.
2. If the update won't start, try unplugging the module from the base unit then start the update. When the update starts, plug the module back in. You should see the progress indicator.
3. Make sure you've downloaded the firmware AND the web pages before you attempt the module update. Frequently the web page update is skipped – this is where the module update happens.
4. Reboot your PC and the Apex base module. Don't ask me why but sometimes this is all it takes to get things going.

Non-Microsoft Platforms

Beginning with release 4.02 you no longer are restricted to using only a PC to apply firmware updates – you can now use a Mac as well. However, you have to do a little work first. You have to download and install an open-source version of the .Net development environment called “Mono” which the Apex firmware loader app requires.

The instructions below are taken from Neptune’s readme file for the 4.02 release:

Beginning with version 3.3 of the Apex firmware uploader you will be able to update the Apex family of products from non-Microsoft platforms.

Requirements:

** Installed version of Mono, available for download at http://mono-project.com. Mono is a cross-platform (open source) version of .NET and is available for many Microsoft and non-Microsoft platforms. The Apex firmware update utility has been tested with Mono version 2.6.1 on Macintosh OS X 10.6.*

** Macintosh OS X requires super user privelage in order to create an entry in the Address Resolution Protocol (ARP) table. This entry is needed in order to open communications between the Apex firmware uploader and the Apex base unit. The Apex firmware upload application will detect that this condition and supply information for manual entry of the ARP table entry (details below). It is likely that other mono environments may have simular requirements and if so the information provided here and by the Apex firmware upload applicatino should be used to add the entry for the particular OS environment you are running.*

Manual creation of ARP table entry

Open a terminal session and enter the command specific to your OS environment to list ARP entries. For the Macintosh this command is:

```
arp -a
```

You will be shown a list of addresses and associated Media Access Control address (MAC) addresses. From this list you should see an entry for the machine you are running on and you may see an entry for the Apex and other network accessible resources. In order to manually create the correct entry for the Apex boot loader mode you will need to identify the Apex or machine address and use the information to build the Apex boot loader entry.

Take the address close to the Apex's address and change the last octet/value in order to make the new address for the Apex boot loader; be sure to choose an address that is not in use (changing one of the 1st 3 octets would likely leave the uploader unable to communicate with the Apex and the upload to fail). Once you select an address to be used you will need to assign the address to the Apex boot loader's MAC address of '0c:1d:12:e0:1f:10' (all Apex units answer to the same MAC address when running under the boot loader).

For the Macintosh the command will look simular to the following:

```
SUDO arp –s <some unused address> c:1d:12:e0:1f:10
```

Connecting directly to your PC/Laptop

If your Apex is operating in a wireless mode and you can't run a cable from it to your home network router, you can load the firmware by connecting your PC or laptop directly to the base unit. The basic process is to define a mini-network consisting of just your PC and Apex. Because you're not using your home network's router, you will not have Internet or email access this way. So download the appropriate firmware file to your laptop first. Also, you do NOT need a special Ethernet 'crossover' cable for this – any good quality 'straight through' Ethernet cable will do. So take your PC/Laptop to your Apex, connect them with your Ethernet cable, and follow these instructions being careful to write down what your options were before making these changes so you can change things back:

Windows XP

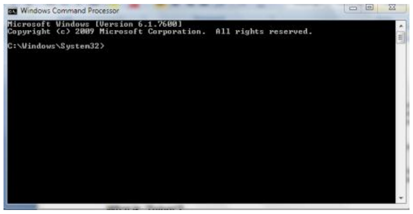
1. Disable the wireless interface on the PC/Laptop by using *Control Panel – Network Connections*. Find your wireless interface, right click and select *Disable*.
2. From the Apex Display module, disable DHCP by using *System – Net Setup - DHCP*.
3. Again, from the Display module, use *System – Net Setup – IP Address. W*rite down the current IP address of the Apex. You will need it for the next step.
4. Now, you have to select an address for the PC (normally you would have DHCP enabled and your router would do this for you). You have to match the first 3 pieces (octets) of the Apex's address and have something different for the fourth. For example, if the Apex is 192.168.1.50, you could use 192.168.1.10. Next, we have to put this new address into the PC and turn off some other controls.
5. On your PC, go into *Control Panel - Network Connections*. You may have multiple network interfaces displayed. Identify the one that is your *Local Area Connection* or *Ethernet Adapter*.
6. Right-click on the icon for the Local Area Connection interface, and select *Properties*.
7. Select *Internet Protocol (TCP/IP),* then click *Properties*
8. On the General tab, *Obtain an IP address automatically* will be selected. We need to change that so we can define one our self. Click the button *Use the following IP address*.
9. Complete the three fields using the IP address you selected above. The *Subnet Mask* should be 255.255.255.0. Make sure *Default Gateway* is blank because we're not using a router anymore.
10. Leave all the other options alone.
11. Click OK as needed to get out of the network configuration windows.
12. Now you're ready to load the firmware. Follow the instructions under 'Execution' earlier in this chapter.

Windows 7 and Vista

1. From the *Control Panel – Network and Sharing Center*, select your wireless interface if you have one, right click and disable it.
2. Follow steps 2, 3 & 4 above.
3. Again, from the *Control Panel – Network and Sharing Center* select your *Local Area Connection*.
4. Select *Properties*, then find and select *Internet Protocol Version 4*, then click *Properties*.
5. Follow steps 8 – 12 above.

Chapter 12: Common issues and solutions - FAQ

What is the 'DOS Command Prompt'?



```
Microsoft Windows [Version 6.1.7600]
Copyright (c) 2009 Microsoft Corporation.  All rights reserved.

C:\Windows\System32>
```

Before Windows there was DOS (Disk Operating System). DOS formed the basis for PC operating systems up through Windows 98. Now, just bits and pieces of the DOS kernel remain. It's there, you just don't always see it. DOS is a command line, text based method of issuing command to your PC. It is still useful for performing tasks where there is no good Windows equivalent.

XP: To activate the DOS window, Click on the 'Start' button then type 'cmd' in the 'Run' field. Your DOS command box should pop up. To end the DOS session, just click the red 'X'.

Windows 7: To activate the DOS window, click on the Windows logo (what used to be the 'Start' button) and in the 'Search programs and files' box enter 'cmd'. The program 'cmd.exe' should be found and you can click on it to start.

If you want to see what all you can do with the DOS command processor, just enter 'help' in the DOS window.

What is 'Telnet'?

Telnet is an old communications protocol that, like DOS still has its use today. Again, it is a text based method of issuing commands and runs in the DOS window. So to use telnet, first start the DOS command window (see above) then enter the command 'telnet' followed by one or more arguments.

For example, to access your Apex controller via telnet, enter the following:

telnet [Apex IP address]. For example: telnet 192.168.1.50

press Enter or Return.

For a list of all the telnet commands, enter telnet /?.

If you are running Windows 7 you might find the telnet command is missing. This is by design. To enable telnet on Windows 7, do the following:

1. Start
2. Control Panel
3. Select 'Programs and Features'
4. Click 'Turn Windows features on or off'
5. Locate and check the 'Telnet Client'
6. OK

I have more modules declared than I physically have

Every time a module is plugged into the base module, the base module will automatically add it and if it's an EB8, it will assign default outlets and programs as well. It's not uncommon for some modules to be declared at the factory. Then, when you plug in your module they're added and you wind up with 2 EB8's when you only have 1. There are two ways you can fix this. You can clear the module list and force the base module to re-scan all modules. You do this via the display module. Navigate to *Setup → Init Memory → Initialize AquaBus* and press select. Also, you can delete extra modules via the module setup screen. Just unplug them, delete them, then plug them back in. The Apex base module will add them and you will no longer have the default factory modules. When you delete an EB8 module you will delete all the program statements associated with the outlets on that module so it's best to do this before you input any program statements.

My Apex does not recognize my module

Occasionally you will get a module that the controller does not recognize. Every AquaBus module has internal memory that records where this module is on the AquaBus – its address. When a module comes from the Neptune, it does not have an address. When you plug it in and connect it to the AquaBus for the first time, the controller registers it and assigns it a unique address. Then later, if you plug this module in again, the address is recognized and not re-assigned. For example, your EB8 may show up as EB8_3. That means it has address #3 on the bus. The controller is #0, the display usually #1, #2 is reserved and all other AquaBus modules go from #3 and up.

Sometimes you wind up with modules with duplicate address numbers. When this happens, the controller is confused and doesn't know where to send that module's data or commands. You can't clear it by re-scanning the AquaBus because the problem is in the module, with its address number.

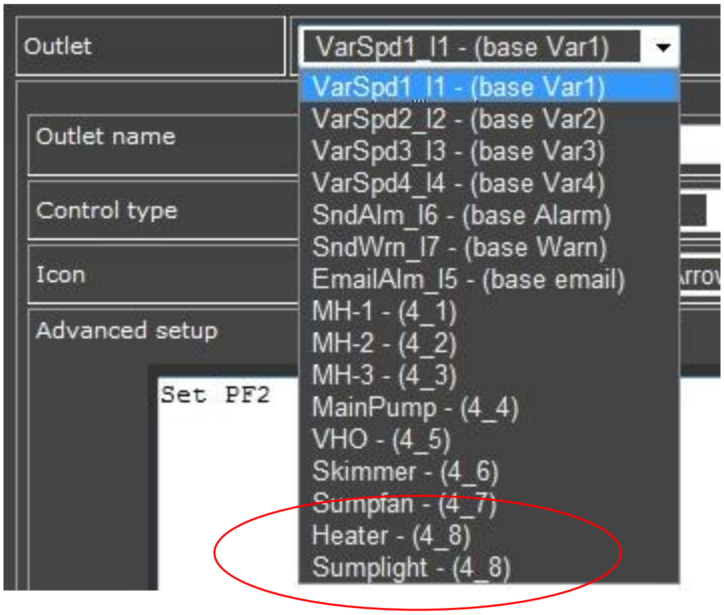
Fortunately, Neptune can fix this but you do have to contact them to do it.

I don't like my outlet or module names – what can I do?

You can change your outlet or module names to whatever you want. Prior to version 4.01 you were limited to 11 characters. You cannot have spaces in your name – if you try, the Apex will remove them. Use the underscore "_" character instead. To rename outlets, use the *Outlet Setup* page. For modules, use the *Module Setup* page. Remember that if you rename outlets or modules and you already have some outlet program statements, you might have to change them if they reference an outlet or module whose name you are changing.

I have duplicate outlet numbers – how do I fix?

If you ever wind up with duplicate outlet numbers like in the example to the left, it's easy to correct. You want to tell the Apex to re-scan the AquaBus. Before you do however, make sure you first backup your program (*see How do I save my Programming and Configuration?*).



From the Display module select *System – Init Memory – Init AquaBus Dev*. The Apex will re-scan your AB modules and recreate the default outlet names. Once that is complete, you will have to re-name your outlets back to their customized labels. Then you can re-load your saved program statements.

How do I send a text message to my phone?

When you configure your email you have the option of sending an email, a text message or both. To send a text message to your phone, enter the phone's text message address in the "alt" field of the email configuration page. See the "*Configure e-Mail*" chapter for more information and a table of valid formats by carrier.

Controller module is unresponsive, Display module is blank

If this happens it almost always means your firmware has been corrupted somehow. First, try cycling the power to the base module by unplugging it from the EB8 and any 12v power. If you have PC access to the base module, try to reload the firmware using the *Configuration → Module Setup* page. If you've lost access to the base unit as well, try loading the firmware this way. Unplug the base unit from power. Start the firmware loader utility, fill in the parameters and start the update. It probably won't find the controller and will begin timing out. Plug the base unit back in at this point. Part of the Apex's startup sequence is going into bootloader mode. The loader should find the base and load the firmware. If this doesn't solve the problem you can try to reload the firmware using the DOS based firmware loader utility. You need to contact Neptune for instructions on how to do this.

Time and Date are wrong after a power failure

The Apex base module has an internal battery that maintains the clock through power failures. This battery can come in a discharged state with a new device and it may take a few days of trickle charging to fully charge the battery so it maintains proper time through a power failure. If it still won't hold a charge you'll see it after a power failure - the base module will come up with a date dozens of years in the future or past. If this happens, you have to contact Neptune support and they can advise you. Beginning with version 4.02 and the NTP clock synching capability, this is less of an issue. The base unit as part of its initial startup will go and check time against a NTP server.

EB8 shows power failed but no power failure occured

The EB8 has intelligence and can communicate back and forth with the base module over the AquaBus. Sometimes its internal calibration can become inaccurate. When this happens the EB8 will report to the base module that it lost power when in fact your outlets are working fine. Normally this won't cause any real issues and the EB8 does not go to fallback mode - your program will still work but it can be annoying if your base module sends you alerts. Neptune support can walk you through the process to recalibrate the unit.

If your status page shows a power failure date/time but no power failure occurred, it's most likely because the controller rebooted for some reason. This is not the same as a 'restart' in the network setup screen which only restarts your network drivers. If the controller reboots, it will create an entry here that will also appear in the XML status log and on the display in the power fail log. It will initialize all outlets, then process all commands to reset them to their proper state. You will see this in the XML outlet log.

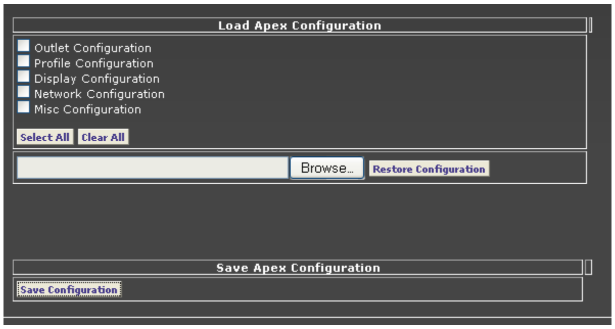
EB8 has intermittent power failures

If the EB8 ever loses communications with the base module, it will default to Fallback settings for each outlet. If this is happening to you first, make sure you have proper fallback settings especially on your critical outlets like pumps. Second, check your firmware and make sure you have the latest release including any beta version. If this doesn't solve the problem, you have to contact Neptune support.

EB8 shows incorrect amperage

Similar to the previous issue, the EB8 may need recalibration. Neptune support can help walk you through the process to do this. You will not lose your programming or configuration.

How do I save my programming and configuration?



Prior to version 4.02 you can’t save your program. Actually you can but it's through a roundabout method. If you go to the XML menu from the web interface and select "XML Program" you will get an XML dump of all your program statements. Another method is via telnet. Log in and issue the ‘l’ command for ‘list’. Select and copy this to either Notepad or Word and save it on your local hard drive. You will need it at some point. Regarding your configuration information (network setup, feed intervals, etc.), there's no such dump. The best thing to do is to use a combination of screen prints from your PC and writing down in a log someplace the configurations noted only in the display module.

Beginning with release 4.02 the ability to save your program and configuration information has been added. There is a new menu item on the *Configuration* page called *Load/Save*. The configuration is saved as a file called “apexcfg.bin”. If you want a different name, rename it after saving.

To reload your configuration, check the appropriate boxes indicating what you want to reload, enter or browse to the filename and press “*Restore Configuration*”. There is no message indicating completion but the base module will restart. Depending on what you reloaded, you may need to exit the browser and go back in to see your restored program and configuration.

My probe (pH or ORP) doesn't read the correct values any more

pH and ORP probes are similar in that they utilize a thin glass bulb like membrane at their tips. These can become coated with algae and other waterborne contaminants that require cleaning. There are a couple schools of thought on this. One says it's safe to gently scrub the membrane with a soft bristle toothbrush or Q-Tip. The other school says soaking in a diluted vinegar solution will dissolve any coating without damaging the delicate membrane. Do your research and make your own choice. However you clean the probe, if that does not correct the problem then a re-calibration may be in order. pH probes are easier and the calibration solutions (pH 7.0 and 10.0) are commonly available at your local fish store for a couple of dollars each. The ORP probe requires quinhydrone in addition to pH 4 and 7 solutions. All can be sourced with a little bit of searching on the web. The calibration process for each is documented in the manual as well as on numerous web sites.

My temperature probe reads 20°

When the Apex has no signal on its temperature probe port (i.e. during a reboot) it reports that temperature at 20 degrees. This is typically temporary unless your probe is plugged into the wrong port or just not working at all. Re-check your probe connections. If you’re confident your connection is correct then the problem is likely a faulty probe or broken wire.

If your heater does not have an internal thermostat that you set a couple of degrees higher than what your Apex would normally turn off the heater, consider modifying your heater program to deal with this possibility. Most heater programs are set to activate the heater when the temperature falls below a certain degree. Also, if it drops another couple of degrees you might activate the alarm. If your temperature probe fails completely, you might not want to activate the heater (leave your alarm alone – you want to know this happened).

Typical heater program:

```
Fallback OFF
Set OFF
If Temp < 77 Then ON
```

Consider this instead:

```
Fallback OFF
Set OFF
If Temp < 77 Then ON
If Temp < 50 Then OFF
```

The second temperature test is low enough that your heater, if working properly will keep the tank temperature above. If the probe reports something below 50, then you know you have a problem with the probe and you don’t want your heater running wild.

My temperature probe reports strange temp but no entry in the log

Sometimes the temperature probes can report strange temperatures that never show up on your data logs. However, the base module recognizes it and responds accordingly, perhaps sending you an email alarm. When you look at your actual tank temperature and the Apex records everything looks fine. What can happen is the probe picks up some type of electrical interference. This causes a brief “blip” in the reported temperature. The base module is analyzing data every second so it sees this and responds. The data logging, only being polled every 10 – 20 minutes, never records the event.

First, try re-routing your probe keeping the cable away from any high current devices like MH lights. See if that helps. If you have to come close to high current cables, try crossing them at 90 degrees instead of parallel. Finally, if that doesn’t work, try adding the “Defer” statement. Defer will make sure the outlet value stays in that state for the prescribed length of time before “really” setting the outlet state. So, in your email or alarm outlet program, <u>as the very last statement</u>, add:

```
Defer 00:30 Then ON
```

This insures the email or alarm outlet stays in the ON state for at least 30 seconds before triggering an alarm. If in that 30 seconds the outlet state goes to OFF, then the Defer statement is not satisfied and no email would be sent. That should help eliminate these spurious readings. If this doesn’t work then you probably have a faulty probe and should contact Neptune.

How do I setup my Apex to notify me of a power failure?

The Apex base module and the EB8 module can each detect power interruptions and you can test for this in your email alarm outlet program. The key to being successful here is to keep power to the base module and your network equipment long enough to send the email. The base module accepts power from two sources - the EB8 via the USB looking AquaBus connection or via a DC adapter (Neptune P/N PS12, 500ma output, 12v, 2.1mm tip, center electrode positive). To insure you keep power long enough, get a small UPS. It only needs to be big enough to power the base module, your router and modem and your PC (assuming you don't use a laptop). A UPS in the 300 - 500Va capacity should be more than enough.

In the *configuration → outlet setup* page, the following statements for the emailAlm (email) or the SndAlm (audible) outlets should do the trick (example assumes you have a single EB8, number 3):

```
Set OFF
If Power EB8_3 Off 001 Then ON  ← this detects if the EB8 loses power
If Power Apex Off 001 Then ON   ← this detects if the base module loses power
```

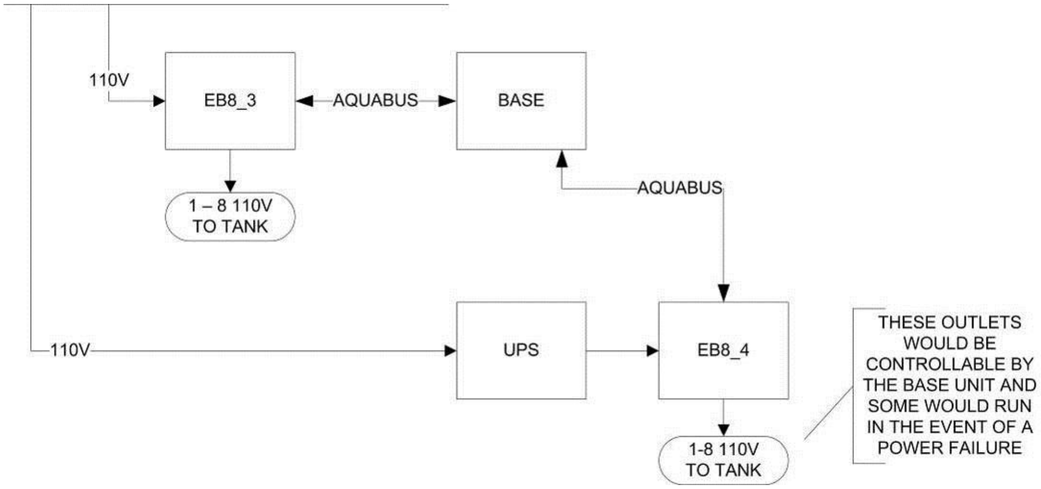
You might wonder, if the base module is on a UPS, how would the "If Power Apex" statement ever be true. Normally it wouldn't, the UPS will prevent that unless of course its battery fails. But anytime you reboot the base module, it is momentarily without power and will recognize that as a power failure, triggering the alarm condition. If your base unit reboots on you during the middle of the day, you'll know about it. Of course, the EB8 gets its power from the wall and would note any normal power interruption. If you wanted to limit these emails to only if the power has been out for a couple of minutes, you can use the new *Defer* command.

Beginning with release 4.03 a new option has been added to the *System – Misc Setup* menu. Called *Power Monitor*, this option will enable the base module to sense a loss of power on the supplemental 12v power port. This gives you the ability to put your EB8 on a UPS and plug the 12v adapter into a regular wall outlet. If power fails, the 'If Power Apex' statement will detect it while the UPS provides uninterrupted power to the EB8 and through that, the base module. The default for this option is 'OFF' so make sure you set it through the display module if you want to use this feature.

After sending you the email alarm, the Apex will wait a short while before sending another email message to avoid streaming dozens of emails to you regarding the same condition. This delay, known as the "re-email delay" is defined in your email configuration in the *configuration → network setup* page. You can set it to however many minutes you want. The default is 5 minutes.

How can I keep essential devices running during power failures?

The following diagram depicts how you can have controllable outlets that are available even after a power failure using a second EB8 running off a UPS. How you size the UPS dictates how many outlets and how long you can run them. A small 300W (600VA) UPS can run a 15W power head for a couple hours.

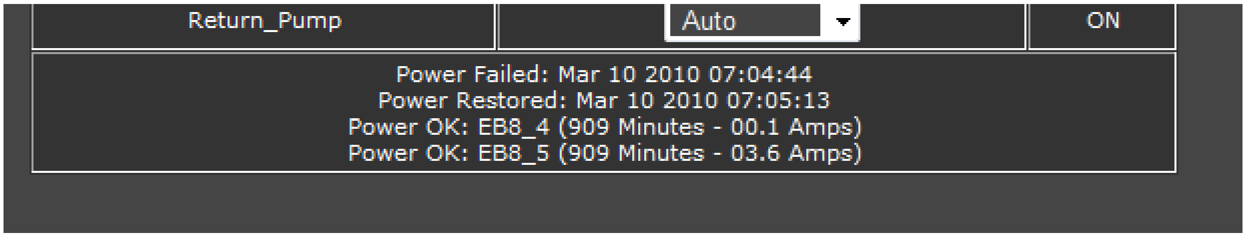

110V
EB8_3
AQUABUS
BASE
1 – 8 110V TO TANK
AQUABUS
110V
UPS
EB8_4
1-8 110V TO TANK
THESE OUTLETS WOULD BE CONTROLLABLE BY THE BASE UNIT AND SOME WOULD RUN IN THE EVENT OF A POWER FAILURE

In the above example, you would put critical items like power heads on EB8_4 along with other non-critical devices. Then in the event of a power failure, all devices on EB8_3 stop but the base unit and devices on EB8_4 continue to function as long as the UPS lasts. Using the "If Power EB8_3 OFF" command, you would then program the Apex to selectively shut down outlets on EB8_4 that are not critical to your tank, leaving just the critical devices running.

Effective with release 4.03 you no longer need a second EB8 in order to get full power redundancy. All you need is a UPS large enough to carry the essential load of your single EB8, a 12v power adapter for the base unit and some logic changes (as documented above) to shut down non-essential load.

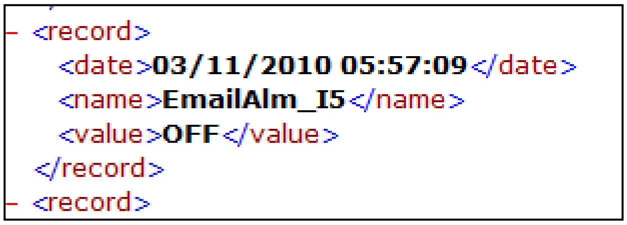
My Power Failed time is not correct

The last time the Base unit lost power or rebooted is displayed at the bottom of the status screen along with the current amp probe values of your EB8 modules. If you are using a supplemental 12v power supply for your Base unit, it's entirely possible that during a power failure your Base never loses power.



Power Failed: Mar 10 2010 07:04:44
Power Restored: Mar 10 2010 07:05:13
Power OK: EB8_4 (909 Minutes - 00.1 Amps)
Power OK: EB8_5 (909 Minutes - 03.6 Amps)

Consider the above example. The screen shot was taken at 21:00 on Mar 11 2010. If you subtract 909 minutes you arrive at approximately 06:00 Mar 11 but the date displayed for the power fail/restoration is Mar 10 at 7am. In fact, power did fail on Mar 11, from about 5am – 6am but the Base never lost power because it is on a UPS system. The Mar 10 entry was because of a reboot.



```
- <record>
    <date>03/11/2010 05:57:09</date>
    <name>EmailAlm_I5</name>
    <value>OFF</value>
  </record>
- <record>
```

Here is a snippet of the XML Outlet log for Mar 11. You can see where at approximately 6am the email alarm outlet was turned OFF corresponding to when power was returned. However, looking at the Datalog for the 5am – 6am period shows the amp probes for EB8_4 and EB8_5 were 0.0 amps. This shows that the EB8 can lose power, but it may not show up in the power fail log.

What happens if a problem occurs with my tank at night?

The base unit has the ability to sound an audible alarm via the *SndAlm* and *SndWrn* outlets (you configure the actual tune selected via the display). But, it's not very loud. If you're worried about a power failure in the middle of the night and possibly not hearing the Apex, you can purchase what's known as a "Failed Circuit Alarm" (Google for it, Amazon carries them for about $30). Normally used for refrigerators and freezers in your basement or garage, this device plugs into a wall outlet and is normally off. When it detects a power failure, an internal battery sounds an 85db (the threshold where the government recommends ear protection) alarm that continues until power is restored, the battery dies, you turn it off with a switch on the face of it or beat it to pieces with a hammer.

You could also plug this device into an EB8 outlet and program it to sound for any valid Apex condition you desire – just remember it alarms when the outlet is "OFF". Your outlet program would look something like this:

```
Fallback ON
Set ON
If Temp > 85 Then OFF
If Temp < 76 Then OFF
```

My data logs have corrupt data – how can I clean it up?

It's not uncommon during setup and configuration of your Apex that the logs get cluttered with junk data – probes not calibrated correctly, power failures, moving outlets and devices around. You cannot delete selected log entries but you can delete an entire day of log data. The Apex stores probe and outlet data in a directory on the base module called 'logs'. If you telnet into the base module, you can list these with the command *'dir logs'*. Each file is named by day. The Apex starts a new file each day at midnight. The file names follow the convention 'yymmdd.odat' and 'yymmdd.pdat' for the outlet and probe logs respectively. Note that the year assumes the 21st century – this is important if your logs got named while you had a clock problem. A year of '57' is '2057' not '1957'. You can view the contents of your probe log via the *'Data Logs'* screen. Just enter the proper date. To view the contents of the outlet log, use the *'XML Outlet'* screen.

To delete a corrupt log, from a telnet session enter *'del logs/<log name>.* Beginning with version 4.02 you can issue the new *'logdel'* command and the oldest log will be deleted without having to enter the full name.

How do I view other days in the XML screens?

Beginning with firmware 4.01 you can view previous day's data and even multiple days of data in a single query. Using the following syntax from your browser, you can view a previous day:

http://<your IP address or DynDNS name>:<port#>/cgi-bin/outlog.xml?sdate=yymmddhhmm

For multiple days, use the above syntax and append as follows:

http://<your IP address or DynDNS name>:<port#>/cgi-bin/outlog.xml?sdate=yymmddhhmm&days=n

Here are two examples:

http://192.168.0.100:5560/cgi-bin/outlog.xml?sdate=100320&days=7
http://mytank.homedns.com:5560/cgi-bin/outlog.xml?sdate=100320

The above works for the XML datalog, XML outlet and graph queries. Note that you would use your IP address for internal (i.e. your home network) access and the DynDNS name for external (i.e. web) access. Also, the hours/minutes is optional in the date parameter.

My graphs hang while loading

The problem of your graphs page not fully loading, basically hanging in the middle seems to be related to JavaScript, Flash and particular browser configurations (steps courtesy of RussM).

1. If you're running IE, don't. Friends don't let friends run IE. You heard it here first.
2. Make sure you have the latest version of Flash installed. Go to Adobe's website to check/download.
3. Don't run IE. Oh, sorry I already said that.

If you're running **Firefox**, try the following:

In the address bar where you would normally enter a URL, type "about:config" without the quotes. Accept the warning.

1. Scroll down the list of options until you find "network.http.max-connections-per-server".
2. Right click and select "modify".
3. Enter a new value of 2.
4. Exit then restart Firefox.

The fix for **Internet Explorer** is much more complicated. Do not attempt this if you're not comfortable with editing the registry:

1. Start Registry Editor (*regedit*).
2. Locate the following key in the registry:HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Internet Explorer\MAIN\FeatureControl\FEATURE_MAXCONNECTIONSPERSERVER
3. On the Edit menu, point to New, click DWORD Value, and then add the following registry values:
   - Value name: iexplore.exe
   - Value data: 2
   - Base: Decimal

4. Locate and then click the following registry subkey: HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Internet Explorer\MAIN\FeatureControl\FEATURE_MAXCONNECTIONSPER1_0SERVER

5. On the **Edit** menu, point to **New**, click **DWORD Value**, and then add the following registry values:
   Value name: iexplore.exe
   Value data: 2
   Base: Decimal
6. Exit Registry Editor and restart IE.

None of the above is a sure-fire fix for the problem. IE seems to be more sensitive than Firefox but even Safari shows the symptoms. Make sure you have the latest Flash installed and that you're not blocking any Flash or JavaScript in your browser configuration.

I changed my password but it doesn't work.

There are two passwords on the Apex base module. The first and most obvious is the password to access the unit via the web pages. This comes preset from the factory as userid: admin and password: 1234. You can change it via the display unit through *System – Net Setup*. <u>You have to reboot the base module in order for the changes to take effect.</u>

There is also a password you can set on the display itself to prevent tampering by your 3-year old. By default, this is "xyz" and can be updated via the *Setup → Display Ssetup → Display Password* screen. To lock the display, select *Control/Status – Display Lock*. You will be prompted for the password the next time you want to use the display.

If you set the display lock and later forget what the password is you can reset it to the default by cycling the power on the base unit. It will reset to the default 'xyz' any time you cycle the power.

What are the firmware versions?

This is an unofficial listing of recent firmware versions. The readme file of each release lists the various improvements and bug fixes. The 4.01 release introduced multi-day graphing. Release 4.02 introduced better time keeping, Salinity and DO probes and the ability to save your program and configuration. Release 4.03 introduced support for VorTech wireless power heads.

You can view your version and release in the first line of text in any of the XML reports,

File Name	Version	Release	Date	Note
Apex_4.03_5A10	4.03	5A10		- Fixed EB8 power failure detection - firmware update of Apex Display to fix firmware corruption issue
Apex_4.03_4E10.zip	4.03	4E10	4/21/10	-change graph page to single graph from multi graph to correct Flash loading problem
Apex_4.03_4A10.zip	4.03	4A10	4/1/10	- Support for PM3 Module - Support for WXM Module - Enhanced power failure support for Apex base module (see system->Misc setup menu) - Fixed DST clock web page bug. - Fixed outlet initialization bug when clock auto updates. - Fixed EB8 power failure detection
100216_2H10	4.02	2H10	3/10/10	-4.02 Becomes production release
100216_2H10	4.02	2H10	2/24/10	-Fixes issues with PM1 temp probe
100214_2F10	4.02	2F10	2/17/10	- Amp graph issue fixed - optimized web page loads for Safari/Firefox
100203_2D10	4.02	2D10	2/8/10	- Profile restore issue - Amp graph not being displayed correctly in some cases. - Probe calibration issue.
100203_2C10	4.02	2C10	2/5/10	-Fixes problem with non-zero value in leap second
100202_2B10	4.02	2B10	2/4/10	-Fixes ORP calibration problem -Fixes re-email delay issue
100201_2A10	4.02	2A10	2/3/10	- A few conductivity module fixes/enhancements - Graphing issue fixed.
100124_1F10	4.02	1E10	1/27/2010	- ORP calibration issue fixed
100124_1F10	4.02	1A10	1/26/2010	- Support for PM2 Module - Preliminary support for VorTech module - Initial graph page displays last 24 hours. - Customizable web status page. - Configuration Load/Save - Enhanced clock features. Day light savings time added, NTP internet clock set support -v3.3 of the firmware loader adds Mac support
91104	4.01	BA09	11/4/2009	- Intermittant web page update failure. - 12 character outlet name bug in xml log files. - support for iPhone application enhancement.

File Name	Version	Release	Date	Note
091023F	4.01	AF09	10/23/2009	Build 2
91016	4.01	AE09	10/16/2009	- xml fix for illegal probe name characters - Multi-day graphing and data logging on web pages. - support for multi-day xml data logs and output logs i.e.: http://apex/cgi-bin/outlog.xml?sdate=091011&days=30 http://apex/cgi-bin/datalog.xml?sdate=091011&days=30&interval=60 sdate = start date, MMYYDDHHMM days = number of days to download interval = specifies the number of minutes between samples to output (only valid for data logs) - Made all probe names consistent between all xml and status pages. - support Apex Lite - support PM1 - added defer command to filter/remove short lived changes in probe values. i.e. Defer 010 Then On
90829	4.0	8J09	8/29/2009	- Improved accuracy of EB8 current logging - Added popup window for the module firmware update status - Fixed AM/PM clock bug - enhanced EB8 power failure functionality. - fixed misc. web page errors.
	4.0	8H09	8/11/09	- fix output status display selection bug - fixed email bug
	4.0	8C09	8/3/09	- Added EB8 support for lunar cycle dim command - on/off only - Optimized HTML web pages - Optimized HTTP web server - Fixed seasonal table bug for March - Fixed 'If Power Apex Off' bug - Fixed Feed Delay bug - Enhanced module and outlet deletion

I want to just erase everything and start over

Don't worry; we've all been there at some point. If you really need to rid yourself of your current program or configuration, there's an easy way to do it with the Apex using the *Init Memory* command. Before you do, make sure you make a backup of your device program and configuration using the *Configuration → Save/Load* function.

The *initialize memory* command has four choices:

1. **All** – initializes your configuration, your outlet programs, your profiles, probe calibrations and clears/re-scans the AquaBus. Basically resets the unit to factory defaults.
2. **Outlets** – clears only your program statements and outlet names but leaves your configuration (i.e. network and email settings) alone.
3. **AquaBus Dev** – re-scans all your AquaBus devices (i.e. EB8's) and re-registers them with the base module (potentially re-numbering them as well). You could use this if you had some ghost modules and wanted to clear them.

Note – AquaBus Dev will delete any outlet programs and re-create default outlet names and programs.

4. **Profiles** – clears your pump/dimming profiles and sets them back to defaults.

I don't like how my outlets are ordered – how can I change?

Outlets are created one of two ways – by connecting a physical AquaBus module to the system or by manually adding one or more modules to create virtual outlets. The 'all' status page will display first the base unit's virtual outlets (variable speed, warning and alarm sounds and email alarm), then the EB8 outlets beginning with the lowest numbered module/outlet to the highest. Finally, your virtual outlets that you created are listed. However, if you then delete a module the next virtual outlets will be added in those empty 'slots'. This can cause confusion when viewing a status page because your virtual outlets appear to be in a random sequence.

Fortunately there's a way around this. Effective with version 4.02 you can configure up to 4 status pages on both the display module and the web pages. You do this by selecting *Configuration – Display Setup*. Pick the page you want to work on then you select the probes you want displayed. Finally, select up to 16 outlets (physical or virtual) that you want to display. This will apply to both the web page and the display module.

Unfortunately, you can't name these pages but if you wanted a page dedicated to just your pumps, you can do that – same if you wanted a page just for virtual outlets. Again, you can configure up to 4 pages with up to 16 outlets per page.

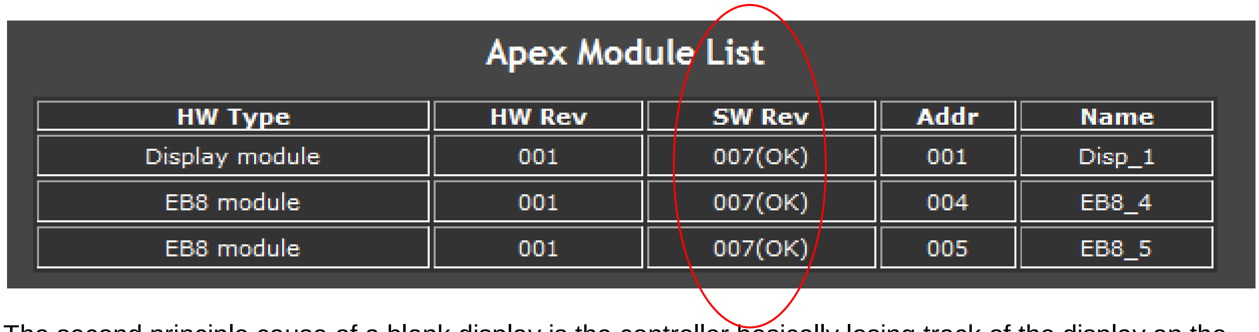
I'm getting a '404 Not Found' error

Assuming you have access to your controller, this is almost always caused by corrupt or missing web pages. Rather than try to figure out which web page is faulty or missing, the simplest thing to do is to reload the web pages. You do this through the firmware utility loader but you don't have to load the firmware. Just select the '*Update Web Pages'* option. For more details on how to do this, see *Chapter 11 Firmware Updates.*

My Display is blank!

This is a common occurrence that seems to happen more with new units than established units. The symptoms are that the controller is working but the display is blank. Typically the web pages are working and you can get into the controller and operate it. Sometimes even the web pages aren't working.

There are two principle causes of this. The first is that the firmware in either the Base module or the Display module is corrupted or partially loaded. The solution for this is simple – reload the firmware and the web pages. Before re-loading the web pages, look at the bottom of your *Configuration – Module Setup* screen. It will show your installed modules and whether their software revision is current. All the numbers should be the same. If they're not, try updating the Display firmware via the *Configuration – Module Setup* screen. Select the Display module, check the '*Update Firmware*' radio button, then *Submit* the change. A small window will pop up showing you the status. After rebooting, the revision numbers should be updated and hopefully, your Display module is working again.



Apex Module List

HW Type	HW Rev	SW Rev	Addr	Name
Display module	001	007(OK)	001	Disp_1
EB8 module	001	007(OK)	004	EB8_4
EB8 module	001	007(OK)	005	EB8_5

The second principle cause of a blank display is the controller basically losing track of the display on the AquaBus. Sometimes it's because somehow two displays wound up being defined on the AquaBus when you only have one. Regardless, the solution is to have the controller rescan the AquaBus. Since your display is not working, you can't do this via the display menu – you have to do it via telnet. I don't document unpublished telnet commands at Neptune's request, but if you post a new thread on the ReefCentral forum about the problem, someone will respond privately.

In release 5A10 a fix was introduced that corrects a problem with the Display module's firmware becoming corrupted during some power fail events. If you have this problem, make sure you download and apply this firmware update. Once you apply the update to the base module you must update the firmware in the Display module itself. The SW Rev will show '009(OK)' once you have done this.

Appendix

Display Module - Menu Hierarchy 4.01

The following is a listing of the various menu items available via the display module. This is relative to release 4.01.

Items that can ONLY be done via the Display are indicated in CAPS.

- Main Menu
  - Control/Status
    - Feed
    - Manual Control
    - Feed Cancel
    - Module Status
    - Display Lock
  - Data Log
    - Graph
    - Power Fail Log
    - Log Interval
  - Setup
    - Outlet Setup
      - Modify Outlet
      - Program Outlet
      - Add Outlet
      - Delete Outlet
      - Repeat Interval
      - FEED INTERVAL
    - Profile Setup
      - Edit Type
      - Edit Settings
    - Module Setup
      - Modify Name
      - Update Module
      - Add Module
      - Delete Module
    - Clock Setup
      - Tank Time Set
      - 12/24 HR
      - LEAP SECOND
    - pH SETUP
      - pH CALIBRATE
      - pH ENABLE
      - TEMP COMP
    - TEMP SETUP
      - TEMP CALIBRATE
      - TEMP ENABLE
      - SCALE
    - ORP SETUP
      - ORP CALIBRATE
      - ORP ENABLE
    - Net Setup
      - DHCP
      - IP Address

    - Net Mask
    - Gateway
    - ADMIN LOGIN
    - ADMIN PASSWORD
    - Host Name
    - DNS Server
    - Alt DNS Server
    - Email Setup
    - Http Port
    - Restart
  - Display Setup
    - Home Display
    - DISPLAY PASSWORD
    - SET DAYTIME
      - DAY START
      - DAY END
    - DAY BRIGHTNESS
    - NIGHT BRIGHTNESS
    - DISPLAY CONTRAST
  - Sound Setup
    - Alarm Sound
    - Warning Sound
    - KEYPRESS BEEP
  - Probe Exp
  - INIT MEMORY
    - ALL
    - INIT OUTLETS
    - INIT AQUABUS
    - INIT PROFILES
- System
- SELF TEST

Display Module - Menu Hierarchy 4.02

The following is relative to release 4.02. Note that a new menu item “System” has been added and some items from the old “Setup” menu have been moved there (Clock, Net, Email, Display, Sound, Probe Exp and Init Memory). In the “Setup” menu new options for Dissolved Oxygen (DO) and Conductivity (Cond) have been added.

Items that can ONLY be done via the Display are indicated in CAPS.

New 4.02 items indicated with ⇨

- Main Menu
    - Control/Status
        - Feed
        - Manual Control
        - Feed Cancel
        - Module Status
        - Display Lock
    - Data Log
        - Graph
        - Power Fail Log
        - Log Interval
    - Setup
        - Outlet Setup
            - Modify Outlet
            - Program Outlet
            - Add Outlet
            - Delete Outlet
            - Repeat Interval
            - FEED INTERVAL
        - Profile Setup
            - Edit Type
            - Edit Settings
        - Module Setup
            - Modify Name
            - Update Module
            - Add Module
            - Delete Module
        - PH SETUP
            - PH CALIBRATE
            - PH ENABLE
            - TEMP COMP
        - TEMP SETUP
            - TEMP CALIBRATE
            - TEMP ENABLE
            - SCALE
        - ORP SETUP
            - ORP CALIBRATE
            - ORP ENABLE
        - ⇨ COND SETUP
            - COND CALIBRATE
            - COND ENABLE
            - COND RANGE
            - TEMP COMP
        - ⇨ DO SETUP

        - DO CALIBRATE
        - DO ENABLE
        - ALTITUDE
        - SALINITY
- System
    - Clock Setup
        - Tank Time Set
        - 12/24 HR
        - Time Zone
        - Auto Set
        - DST
        - LEAP SECOND
    - Net Setup
        - DHCP
        - IP Address
        - Net Mask
        - Gateway
        - ADMIN LOGIN
        - ADMIN PASSWORD
        - Host Name
        - DNS Server
        - Alt DNS Server
        - Http Port
        - Restart
    - Email Setup
    - Display Setup
        - Home Display
        - DISPLAY PASSWORD
        - SET DAYTIME
            - DAY START
            - DAY END
        - DAY BRIGHTNESS
        - NIGHT BRIGHTNESS
        - DISPLAY CONTRAST
    - Sound Setup
        - Alarm Sound
        - Warning Sound
        - Keypress Beep
    - PROBE EXP
    - INIT MEMORY
        - ALL
        - INIT OUTLETS
        - INIT AQUABUS DEV
        - INIT PROFILES
- SELF TEST

Display Module - Menu Hierarchy 4.03

Items that can ONLY be done via the Display are indicated in CAPS.

New 4.03 items indicated with ⇦

- Main Menu
  - Control/Status
    - Feed
    - Manual Control
    - Feed Cancel
    - Module Status
    - WXM STATUS ⇦
    - Display Lock
  - Data Log
    - Graph
    - Power Fail Log
    - Log Interval
  - Setup
    - Outlet Setup
      - Modify Outlet
      - Program Outlet
      - Add Outlet
      - Delete Outlet
      - Repeat Interval
      - FEED INTERVAL
    - Profile Setup
      - Edit Type
      - Edit Settings
    - Module Setup
      - Modify Name
      - Update Module
      - Add Module
      - Delete Module
      - CONFIG MODULE ⇦
    - PH SETUP
      - PH CALIBRATE
      - PH ENABLE
      - TEMP COMP
    - TEMP SETUP
      - TEMP CALIBRATE
      - TEMP ENABLE
      - SCALE
    - ORP SETUP
      - ORP CALIBRATE
      - ORP ENABLE
    - COND SETUP
      - COND CALIBRATE
      - COND ENABLE
      - COND RANGE
      - TEMP COMP
      - DO SETUP
      - DO CALIBRATE
      - DO ENABLE

            - ALTITUDE
            - SALINITY
    - System
        - Clock Setup
            - Tank Time Set
            - 12/24 HR
            - Time Zone
            - Auto Set
            - DST
            - LEAP SECOND
        - Net Setup
            - DHCP
            - IP Address
            - Net Mask
            - Gateway
            - ADMIN LOGIN
            - ADMIN PASSWORD
            - Host Name
            - DNS Server
            - Alt DNS Server
            - AQUASERV SETUP
                - AQUASERVER STATUS
                - SERVER
                - USER ID
                - AQUASERVER PASSWD
                - CONTROL
            - Http Port
            - Restart
        - Email Setup
        - Display Setup
            - Home Display
            - DISPLAY PASSWORD
            - SET DAYTIME
                - DAY START
                - DAY END
            - DAY BRIGHTNESS
            - NIGHT BRIGHTNESS
            - DISPLAY CONTRAST
        - Sound Setup
            - Alarm Sound
            - Warning Sound
            - Keypress Beep
        - Misc Setup
            - PROBE EXP
            - POWER MONITOR
        - INIT MEMORY
            - ALL
            - INIT OUTLETS
            - INIT AQUABUS DEV
            - INIT PROFILES
        - SELF TEST

Apex Programming Quick Reference Guide

TIME = HH:MM in 24 hour format **DURATION** = MMM:SS **DELAY** = MMM

Statement	Syntax	Notes
Day of Week	**If DoW [SMTWTFS] Then [ON/OFF/PROFILE]**	**Substitute '-' for day to evaluate FALSE**
Defer	**Defer [DURATION] Then [ON/OFF]**	
Fallback	**Fallback [ON/OFF]**	**For physical outlets only**
Feed / Maintenance	**If [FEED CYCLE] [DELAY] Then [ON/OFF/PROFILE]**	
Minimum Time	**Min Time [DURATION] Then [ON/OFF]**	
Moon Rise/Set	**If Moon [+/-] [DELAY] / [+/-] [DELAY] Then [ON/OFF/PROFILE]**	**(+) is assumed and not required**
Oscillate	**Osc [DURATION / DURATION / DURATION] Then [ON/OFF/PROFILE]**	**'divide by 10' option changes DURATION to SSS:ss**
Outlet State Testing	**If Outlet [OUTLET NAME] = [ON/OFF] Then [ON/OFF/PROFILE]**	**Outlet Name can be physical or virtual**
Power	**If Power [POWER PROBE] [ON/OFF] [DELAY] Then [ON/OFF/PROFILE]**	**Probe names are 'Apex' or 'EB8_x' where 'x' is the EB8 AquaBus position.**
Probe Testing	**If [PROBE NAME] [</>] [VALUE] Then [ON/OFF/PROFILE]**	**Probe names are Temp, PH, ORP, Cond, DO. Expansion boxes add 'Xn'.**
Regional Temperature	**If Temp [</>] RT [+/-] [DIFFERENTIAL DEGREES] Then [ON/OFF/PROFILE]**	**Differential Degrees is DD.d**
Set	**Set [ON/OFF/PROFILE]**	
Sun Rise/Set	**If Sun [+/-] [DELAY] / [+/-] [DELAY] Then [ON/OFF/PROFILE]**	**(+) is assumed and not required**
Switch Testing	**If [SWITCH NAME] [OPEN/CLOSED] Then [ON/OFF/PROFILE]**	**Switch names are SWITCH1 through SWITCH6. Expansion boxes change to "A1" through "A6", "B1" – "B6", etc.**
Time	**If Time [TIME] to [TIME] Then [ON/OFF/PROFILE]**	

Program Examples

Here is a collection of program examples from forum posts for some of the more common tasks you would use your Apex for. Feel free to use/modify them for your particular needs. As with any program, please take the time to test before you implement it. Don't assume it works correctly in your environment without testing it first!

Dosing – Calcium and Alkalinity

This example shows two separate dosing outlets called Calcium and Alkalinity. The Calcium outlet runs for 3 minutes 20 seconds each hour except at night (lights = OFF), during a FeedA cycle or if the pH is > 8.35. The Alkalinity runs the same except is offset by 30 minutes from the Calcium

```
[Calcium]

Fallback OFF
OSC 000:00\003:20\057:40 Then ON
If Outlet [Lights] = ON Then OFF
If pH >8.35 Then OFF
If FeedA 000 Then OFF

[Alkalinity]

Fallback OFF
OSC 030:00\003:20\027:40 Then ON
If Outlet [Lights] = ON Then OFF
If pH >8.35 Then OFF
If FeedA 000 Then OFF
```

ATO – Double Switch on a schedule

Here, an outlet is triggered to run for 5 minutes every 4 hours unless Switch1 is OPEN. Switch1 is a high water level switch in the sump. Switch2 is in the ATO reservoir and if OPEN, indicates the reservoir is empty.

```
Fallback OFF
OSC 000:00/005:00/235:00 Then ON
If Switch1 OPEN Then OFF
If Switch2 OPEN Then OFF
```

ATO – Single Switch on an 'as-needed' basis

Switch1 is a water level switch in your sump. When OPEN, the sump is full. The Defer command keeps the program from rapidly cycling – the switch has to be CLOSED for 5 minutes before activation can occur.

```
Fallback OFF
If Switch1 OPEN Then OFF
If Switch1 CLOSED Then ON
Defer 005 Then ON
```

VorTech Variable Feed Mode

Posted by ‘YCore’ this program uses a series of virtual outlets to have VorTech pumps go through a series of modes – Off, Nutrient Export, ReefCrest – to perform a custom Feed cycle. Entire process spans 55 minutes.

Sequence Logic:
1) Feed B is started. Return and skimmer turn off. All virtual outlets turn on, but V_OFF is listed last, so Vortech turns off.
2) 15 minutes later, feedB ends, pump and skimmer stay off, V_OFF becomes disabled, and V_Slow becomes the active program statement since it now last in the list that is on.
3) 15 Minutes later (total 30), V-Slow turns off and V_Nutrient becomes active.
4) 10 Minutes later (total 40), Nutrient turns off and pump resumes reef crest.
5) 10 minutes later (total 50), return pump turns back on
6) 5 minutes later (total 55), skimmer turns back on

Virtual Outlets:

V_OFF
```
Set OFF
If FeedB 000 Then ON
```

V_SLOW
```
Set OFF
If FeedB 015 Then ON
```

V_NUTRIENT
```
Set OFF
If FeedB 025 Then ON
```

Physical Outlets:

Return Pump:
```
Fallback ON
Set ON
If FeedB 035 Then OFF
```

Skimmer
```
Fallback ON
Set ON
If Outlet PUMP_CRIT = OFF Then OFF
Defer 005:00 Then ON
If FeedB 040 Then OFF
```

Vortech
```
Fallback ON
Set OFF
If Time 11:00 to 23:00 Then Reef_C_D
If Time 23:01 to 10:59 Then Lagoon_N
If Outlet V_Nutrient = ON Then Nutrient
If Outlet V_Slow = ON Then Slow
If Outlet V_OFF = ON Then OFF
```

Index