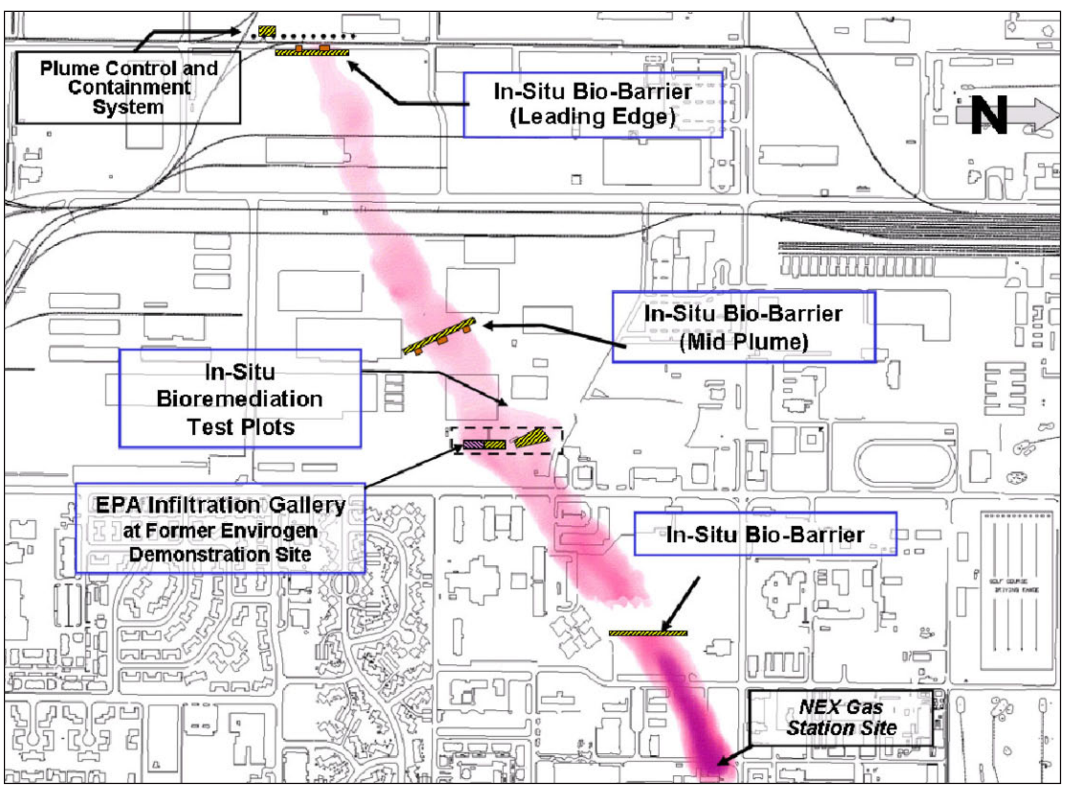
United States Environmental Protection Agency

EPA/600/R-08/068
June 2008

Demonstration and Evaluation of an Automated Infiltration Gallery System at Port Hueneme, CA



EPA/600/R-08/068
June 2008

Demonstration and Evaluation of an Automated Infiltration Gallery System at Port Hueneme, CA.

by

Carl G. Enfield[1] and Bob K. Lien[2]

**[1]Pegasus Technical Services, Inc.
Cincinnati, OH 45219
[2]Land Remediation and Pollution Control Division
National Risk Management Research Laboratory
Cincinnati, Ohio 45268**

**National Risk Management Research Laboratory
Office of Research and Development
U.S. Environmental Protection Agency
Cincinnati, Ohio 45268**

DISCLAIMER

The work reported in this document is funded by the United States Environmental Protection Agency (EPA) under Work Assignment 15 of Contract EP-C-05-056 to Pegasus Technical Services, Inc. It has been subjected to the Agency's peer and administrative reviews and has been approved for publication as an EPA document. Any opinions expressed in this paper are those of the author(s) and do not, necessarily, reflect the official positions and policies of the EPA. Any mention of products or trade names does not constitute recommendation for use by the EPA.

FOREWORD

The United States Environmental Protection Agency (EPA) is charged by Congress with protecting the nation's land, air, and water resources. Under a mandate of national environmental laws, the Agency strives to formulate and implement actions leading to a compatible balance between human activities and the ability of natural systems to support and nurture life. To meet this mandate, EPA's research program is providing data and technical support for solving environmental problems today and building a science knowledge base necessary to manage our ecological resources wisely, understand how pollutants affect our health, and prevent or reduce environmental risks in the future.

The National Risk Management Research Laboratory (NRMRL) is the Agency's center for investigation of technological and management approaches for preventing and reducing risks from pollution that threaten human health and the environment. The focus of the Laboratory's research program is on methods and their cost-effectiveness for prevention and control of pollution to air, land, water, and subsurface resources; protection of water quality in public water systems; remediation of contaminated sites, sediments and groundwater; prevention and control of indoor air pollution; and restoration of ecosystems. NRMRL collaborates with both public and private sector partners to foster technologies that reduce the cost of compliance and anticipate emerging problems. NRMRL's research provides solutions to environmental problems by developing and promoting technologies that protect and improve the environment; advancing scientific and engineering information to support regulatory and policy decisions; and providing the technical support and information transfer to ensure implementation of environmental regulations and strategies at the national, state, and community levels.

This publication has been produced as part of the Laboratory's strategic long-term research plan. It is published and made available by EPA's Office of Research and Development to assist the user community and to link researchers with their clients.

Sally Gutierrez, Director
National Risk Management Research Laboratory

CONTENTS

ABSTRACT

Laboratory and field studies have shown that is possible to degrade most fuel components under oxidizing conditions. The spread of soluble fuel components released to groundwater environments is often enhanced because ground water can not supply oxygen at a rate equal to the demand of the dissolved fuel. This study was conducted to evaluate the feasibility of using the vadose zone or soil above the water table which was maintained in an oxidizing state by a control system without artificially adding oxygen to treat fuel components. The test was performed at Port Hueneme California in an area formerly contaminated by a fuel spill. The scope of the study was limited to evaluating the field reliability of a remotely operated infiltration gallery. The results showed that it was possible to operate the system remotely with very little intervention from a field technician. None of the problems normally associated with infiltration galleries plugging from biological growth were observed. This was likely due to the draining of the distribution lines between each water application cycle. The study was conducted for more than a year but there was little contamination remaining in the groundwater when the study was performed and it was not possible to evaluate the chemical performance of the system.

FIGURES

APPENDICES

ACKNOWLEDGEMENTS

We wish to thank the NFESC personnel at the US Navy’s Port Hueneme for their support in obtaining a permit from the state of California to perform the experiment and the continued support during the experiment making field observations and recording data on the data log. Without this support this project would not have been possible.

1 INTRODUCTION

1.1 Objectives

The objective of this project was to demonstrate and evaluate the robustness of an automated infiltration gallery system at Port Hueneme, CA designed to remove fuel components from ground water by moving the ground water from a reducing environment in the water table aquifer to the vadose zone where oxidizing conditions exist providing favorable conditions for aerobic bioremediation of the potentially contaminated groundwater. The primary source of oxygen in the vadose zone is the atmosphere. Oxygen is supplied by a combination of processes including diffusion, and natural pumping actions caused by changes in water content and changes in atmospheric pressures. An infiltration gallery, as used in this report, refers to a structure in the soil consisting of slightly perforated (10% or less of the surface area) horizontal pipes surrounded initially by permeable sands or gravels used to distribute water within the soil profile and then allow the water to infiltrate naturally. Infiltration galleries are frequently used for onsite waste treatment called leach fields after a septic tank and to assist in dissipating waters in storm water catchments basins in urban areas. Installing an infiltration gallery at underground storage tank (UST) facilities provides immediate access to local groundwater providing rapid identification of contamination with a built in remedial system capable of handling small releases. Questions to be answered directly by this investigation were: (1) the ability to detect the depletion of oxygen in the vadose zone and consequently reduce the application of water until oxygen returns to the infiltration gallery; (2) automatically calibrate oxygen sensors used to assess the oxygen status of the vadose zone; (3) maintain the ability of delivering groundwater to the vadose zone without plugging the distribution system with biological growth which has been a major shortcoming of the infiltration gallery approach; (4) monitor the performance of the system remotely with minimum local intervention. In addition, there were concerns as to the durability of the different sensors, in particular the oxygen sensors many of which have a short useable life (the ones used in this study were reported to have a life expectancy of ten years), and control systems under this type of environmental conditions.

1.2 Rationale

Methyl tertiary Butyl Ether (MTBE), the principal oxygenate in fuel, peaked in US production at about 3.3 billion gal per year in 1998 (USGS, 2006). MTBE was the second most-produced chemical in the U.S. in 1997 (OEHHA, 1999). MTBE is the second most frequently detected VOC in samples from domestic and public wells despite its relatively short production and use history (USGS, 2006). In general, most fuel components are biologically degradable under aerobic conditions. If favorable subsurface environmental conditions can be maintained, there is considerable potential for natural attenuation processes to mitigate small releases from Underground Storage Tanks (UST). Selected evidence of the attenuation potential is discussed below.

1.2.1 MTBE biodegradation

There are numerous laboratory and field studies that document the degradation of MTBE and other fuel components under oxidizing conditions (Borden et al., 1997; Bradley et al., 1999; Church et al., 1999; Javanmardian and Glasser, 1997; Mackay et al., 1999; Mackay et al., 2000; Wilson et al., 2002; Wilson et al., 1999). Compared to other fuel components MTBE may be more recalcitrant than many but there are sufficient organisms naturally occurring to expect degradation can occur. When MTBE is present one of the daughter products, tertiary butyl alcohol (TBA), is also frequently observed. TBA is also degradable under oxidizing conditions (Bradley et al., 2002; Kaharoune et al., 2002; Schmidt et al., 2004; Wilson et al., 2001). At subsurface fuel releases, even when the resident ground water is initially oxidizing, the demand for oxygen frequently out strips the natural ability of a formation to attenuate the contamination before risks are evident at water supply wells. The problem is exacerbated when the groundwater is reducing.

There are technologies available in the waste water industry that effectively treat oxidizable waste waters. Most of the systems collect the waste water and treat it in centralized plants. Unlike the municipal and industrial waste water sources, our gasoline distribution system is diffuse and scattered in our urban and rural communities such that it is not practical to collect contaminated ground water and treat it at centralized facilities. An alternative would be to use one of the land-based on-site small flows waste water treatment systems. In this study, an infiltration gallery, utilizing the principals of a Mound On-Site Treatment System, was selected for evaluation because of its adaptability to a large range of soil conditions.

1.2.2 Mound On-Site Treatment System

The objective of the mound, as with other soil-based units, is to treat and disperse domestic and commercial waste water on-site via the subsurface in an environmentally acceptable manner and to protect the public health (Converse and Tyler, 2000). A mound treatment system consists of a septic tank that removes solids by settling and floatation with some of the solids transformed into soluble material which pass to the dosing chamber (Figure 1). The dosing chamber contains a pump or siphon, which transfers effluent, under pressure, to a distribution network of small diameter pipes with small perforations which distributes the effluent uniformly over the absorption area of the mound. The effluent infiltrates into and percolates through the mound sand and native soil, the pathogens are removed, the organic matter is transformed both aerobically and anaerobically, nitrogen is transformed sequentially from organic and ammoniacal forms to nitrate which requires oxidizing conditions and phosphorus is retained in the native soil and may slowly migrate depending on the soil properties.

The design and performance of the mound system, as well as other soil based treatment systems, is usually based on typical domestic waste water properties which has been pretreated by passing the waste water through a septic tank. Typical domestic effluent

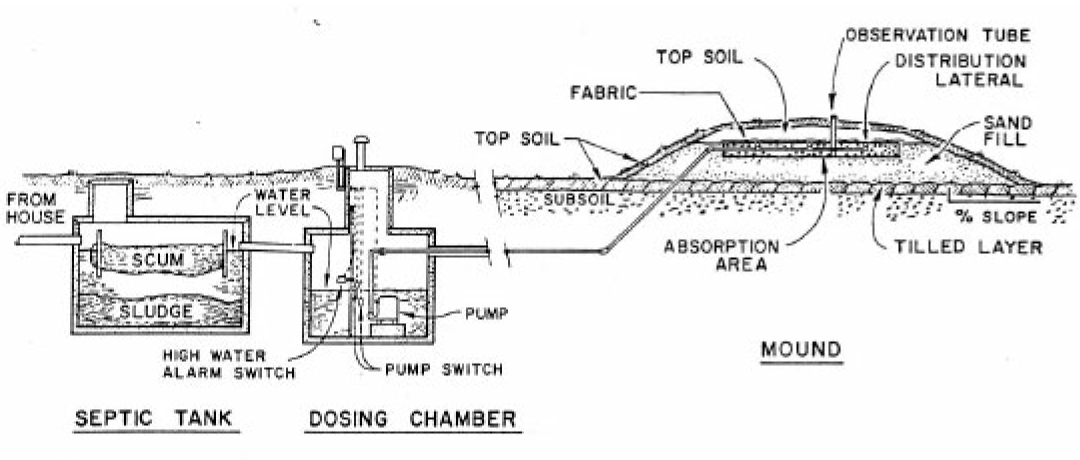


Figure 1. Schematic of the Wisconsin mound system showing septic tank, dosing and mound (Converse and Tyler, 2000).

will have a biochemical oxygen demand (BOD) in the range of 150 - 250 mg/L and total suspended solids (TSS) in range of 50 - 100 mg/L. Fats oils and greases (FOG) are typically below 15 mg/L. The mound treatment system is suitable for final treatment and dispersal of highly pretreated effluent from units such as aerobic treatment systems, sand filters, peat filters and biofilters which typically produce effluent with BOD and TSS less than 25 mg/L (Converse and Tyler, 2000).
An efficient on-site waste water system has a balance between the amount of oxygen entering the system with the amount of oxygen needed to decompose the organic matter and meet the demand for other reactions. If the amount of oxygen entering the system cannot meet the demand, a clogging mat will form (Erickson and Tyler, 2001). Clogging mats form at a contrast in the hydraulic conductivity (when the fluid passes from a high hydraulic conductivity area to a lower hydraulic conductivity). If the system design can be manipulated to supply oxygen into the soil exceeding the rate needed, the life and loading rate of the on-site waste water system can be greatly increased. Aerobic bacteria use oxygen as their terminal electron acceptor to convert organic molecules to carbon dioxide and ammonia to nitrate. Oxygen is the most effective oxidizing agent; therefore decomposition in an aerobic setting is far more efficient than in an anaerobic environment.

In addition to supplying the oxygen needed to avoid a clogging mat, sufficient separation distance is required between the discharge point in the mound and the regulatory control plane to allow time for aerobic transformations. The regulatory control plane is the location along the flow path of the treated waste water where the treated waste water must meet health and safety requirements (usually considered to be the water table or perched water table under the mound).

The natural soil formation must be capable of accepting the hydraulic load from the mound system. The acceptable loading rate can be estimated based on the soil texture, depth to the seasonal high water table, and slope of the soil surface (water table slope and soil surface slope are usually assumed to be the same) (Tyler, 2001).

The composition of the contaminants in groundwater down gradient from a fresh fuel release will be the mole fraction solubility of the components in the fuel (Cline et al., 1991). With time, the composition will change and the ability of the treatment system to accept a load will change. For this study, rather than designing the system based solely on an expected water composition, the oxygen status is monitored and water is applied only when there is sufficient oxygen to prevent the development of a clogging mat. An upper application rate is limited to the maximum the natural soil can accept hydraulically.

2 SITE DESCRIPTION

2.1 Prior Studies and Information Relevant to System Design

The test plots were located at a site previously used to study the degradation of MTBE in a plume created by a leaking underground storage tank at the Port Hueneme's Base Exchange as shown in Figure 2. Port Hueneme has performed numerous studies related to MTBE remediation and the evaluation of sight characterization tools. The

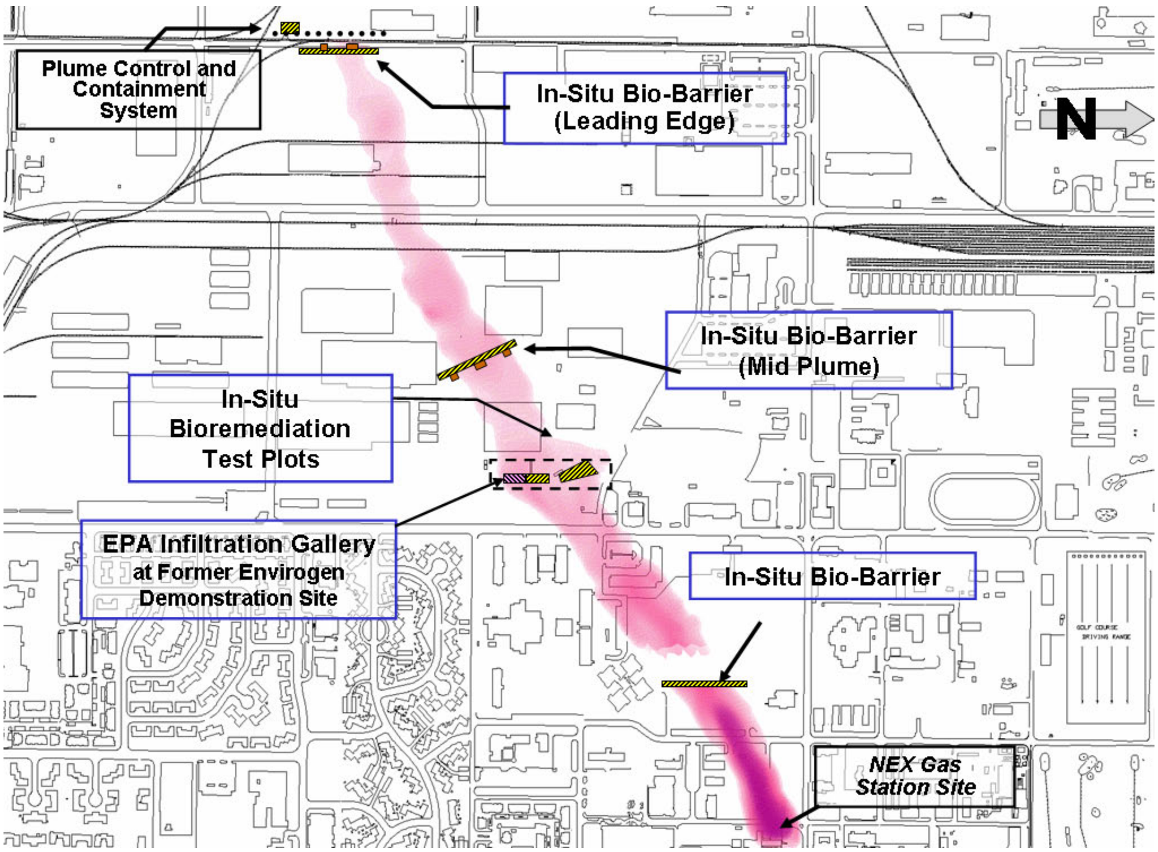


Figure 2. Port Hueneme MTBE and BTEX plume as of May 2001 showing project area used in this study.

infrastructure already available made it a good candidate as a test site for this study. The study site selected by personnel at Port Hueneme was a site previously used by Envirogen to study MTBE degradation *in situ* using propane to stimulate biodegradation. We modified the Envirogen site for the infiltration gallery study. The study area was near monitoring well CBC-45. These test plots were previously set up to monitor MTBE degradation in the water table aquifer and installed normal to the direction of groundwater flow as shown in Figure 3. Previously during the Envirogen study a bromide tracer study was performed at the site using all of the 2" wells shown in Figure 3. The study was performed similar to a borehole dilution study where bromide was injected into the formation at a constant rate and water within the borehole was circulated with non contact pneumatic pumps to generate a uniform concentration in the bore hole.

Time of travel to the various monitoring points was then evaluated. The flow rate through aquifer was variable as a function of depth with a maximum determined interstitial velocity of 0.05 m/day for a zone 2.3 m thick. In a regional study based on historical data of the movement of the MTBE, the average interstitial velocity was determined to be 0.006 m/day with a total average aquifer thickness of 3.5 m. This information, based on prior work not funded by this project, was used to estimate the desired maximum pumping rate for the plots. Each plot for the current study was constructed 5 m in width. If one assumes the porosity of the formation is 0.25 then the volume flow needing to extract all of the water passing through a plane through the infiltration gallery averages 0.05 l/m for the high flow rate zone measured using tracers or 0.015 l/min for the aquifer averaged flow rate calculated based on an interstitial velocity of 0.006 m/day.

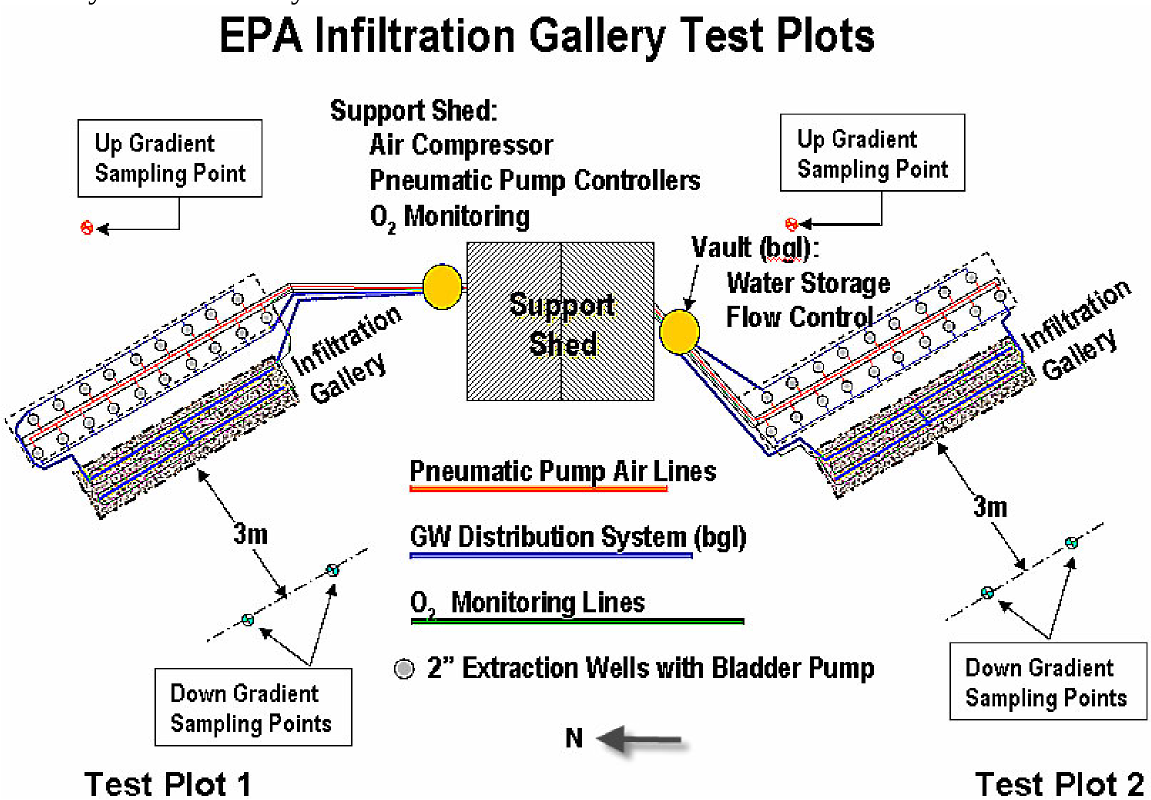


Figure 3. Schematic layout of test plots. Wells used in a previous study at this location were left in place. Many of the wells and monitoring points were reused in this study.

2.2 Infiltration Gallery and Plot Installation

The area previously occupied by the oxygen and propane injectors used in the Envirogen study was removed (Figure 4) and the infiltration galleries constructed in this area. The existing soil was removed down to 1.2m (Figure 5) and the formation soil replaced by a medium sand. The infiltration gallery (Figure 6) consisted of a water distribution system constructed from 1/2" polyethylene tubing and oxygen permeable membranes placed at two depths below the water distribution system. The lower loop oxygen permeable membrane (1/8" ID x 0.365 OD silicone tubing) was placed on a bed of sand. Then 45 cm of sand placed on the lower membrane. And then an upper oxygen measurement membrane was installed in the plot. The upper membrane was the same design as the lower membrane. The upper membrane was covered by 45 cm of sand and the distribution system was placed approximately 30 cm below the ground surface. Multi depth vadose zone samplers were installed in the plot but sampling was discontinued when it was discovered that there was little quantifiable MTBE in the formation water.

Figure 4. Location of north infiltration gallery showing Envirogen gas injection wells prior to the beginning of work.

Figure 5. Removal of 1.2 m of the existing formation material to allow for the constriction of the infiltration gallery.

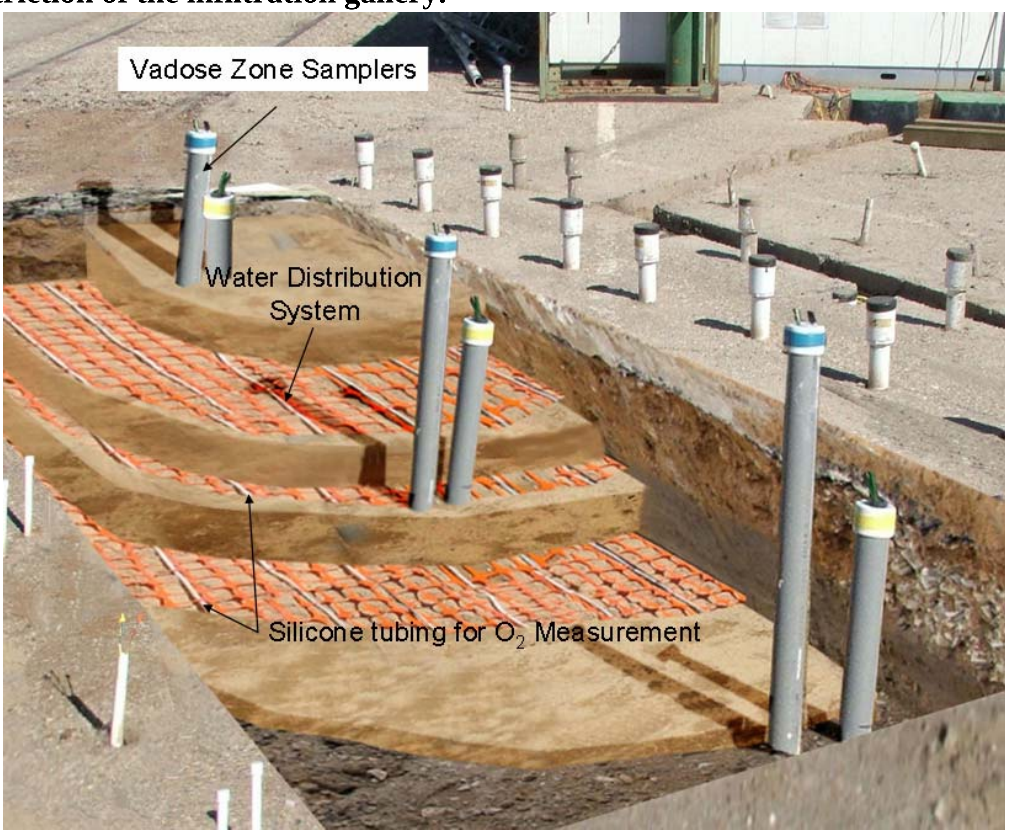


Figure 6. Infiltration gallery as installed.

2.3 Limitation of Scope Due to MTBE Depletion

When the project was initially proposed and the permitting process initiated, the site was contaminated with MTBE based on data provided by the Envirogen study. Vadose zone samplers were installed to monitor water quality as it passed through the vadose zone. However, by the time permits were in place and the system constructed, either the MTBE had been dissipated by up-gradient remedial activities or the flow direction changed sufficiently to deplete the MTBE from the test area making chemical performance evaluation a moot point. It was decided, after the first few months of sampling did not find MTBE above quantifiable limits, to continue the evaluation without performing chemical analysis and evaluate the mechanical performance of the design system and not artificially add chemicals to the water potentially increasing Port Hueneme's liability by adding contaminants to the site.

3 LABORATORY GAS PERMEABLE MEMBRANE DIFFUSION TEST

Remediation of MTBE contaminated water requires oxygenated water as referenced above. One of the objectives of this work was to maintain an oxidizing environment in the vadose zone. The hypothesis was that one should be able to change the water application rate such that oxidizing conditions were continually maintained in the formation. Thus if oxygen conditions in the vadose zone fell below a set level (15% in air in equilibrium with the soil solution) the water application rate would be turned off. To do this, it is necessary to monitor the oxygen in the vadose zone. This suggests two issues that must be addressed. First there needs to be a method for measuring the oxygen status of the formation and a need to be able to calibrate the measurement device to insure accurate measurements. Silicone tubing is known to be permeable to oxygen as well as numerous other gases. It was selected as a gas permeable membrane that could equilibrate with the soil water allowing the assessment of the oxygen in the water. The membrane material was tested under laboratory conditions to evaluate how fast the gas within the membrane could respond to a step change in oxygen concentration surrounding the membrane. Ideally, if the response were very fast then air could be pumped through tubing installed in the infiltration gallery and the amount of oxygen remaining would represent an equilibrium state between the gas in the tubing and the water in the formation. This equilibrium value could be determined by some type of sensor with the output signal used for control purposes. The oxygen sensor selected was either a Figaro KE-50-F3 which is reported to have a life of 10 years and a nominal output of 250 mV at 100% oxygen or a Figaro KE-25-F3 with a reported half life of 5

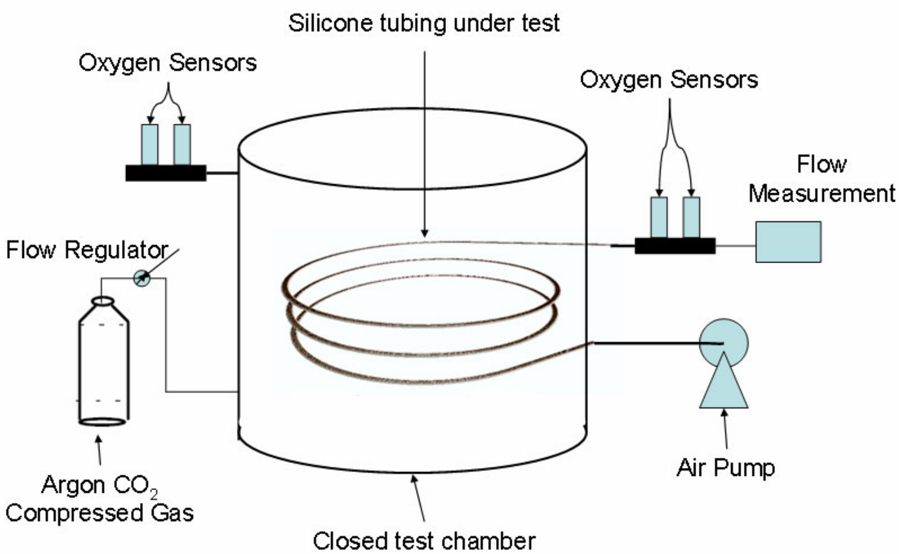


Figure 7. Schematic drawing of the laboratory test chamber used to evaluate the silicone tubing used as a diffusion membrane for oxygen measurement.

years and a nominal output of 60 mV at 100% oxygen. The membrane diffusion was tested by first calibrating four of the oxygen sensors using air as 20.5% oxygen and then passing a welding gas composed of Argon and CO_2 passed by the sensing membrane assumed to represent 0% oxygen. For the membrane diffusion test, two of the sensors were placed at the discharge of the chamber used (Figure. 7) to verify that the oxygen in the test chamber had been purged. Two oxygen sensors were also placed at the discharge from a known length of silicone tubing placed in the chamber. Air was pumped through the tubing at known flow rates and measurements made of the oxygen concentration emitted. At time zero for each flow rate both the oxygen in the test chamber and the oxygen in the tubing were at the same concentration assumed to be 20.5% oxygen. Then the test chamber was purged with a mixture of CO_2 and Argon (i.e. no oxygen). The chamber oxygen level rapidly droped from 20.5% to near zero. This was verified using the oxygen sensors at the discharge of the test chamber and is equivalent to a step change in relative concentration going from one to zero. The air entering the silicone tubing remained at 20.5% but diffused through the tubing to the test chamber where it is flushed away. The longer the gas resides in the tubing the lower the oxygen concentration in the effluent. Since this is primarily a diffusion process, the shape of the curve should be exponential where the loss is initially fast and then slows down as the concentration gradient is reduced. Evaluations can be made only at the discharge of the test chamber. Thus, changing the flow rate through the membrane tubing changed the residence time of a packet of air analyzed by the oxygen sensor. Residence time is inversely proportional to flow rate through the tubing. The test was performed using the same type of oxygen sensors used in the field so the response time reported includes the response time of the sensors (Figure 8). The figure shows the relative change in oxygen as a function of residence time in the oxygen permeable tubing. The last data point is likely in error and the residence time was likely much longer than reported. The reason for the error is that the flow rate was at the limit of both the pump and flow measuring equipment and it is believed that the flow intermittently stopped during this portion of the test. It was not possible to maintain flow at lower flow rates with the equipment available. Based on the data presented in Figure 8, the time that would be required to achieve equilibrium was sufficiently long that the design used in the field employed a closed loop for measuring the oxygen rather than single pass air.

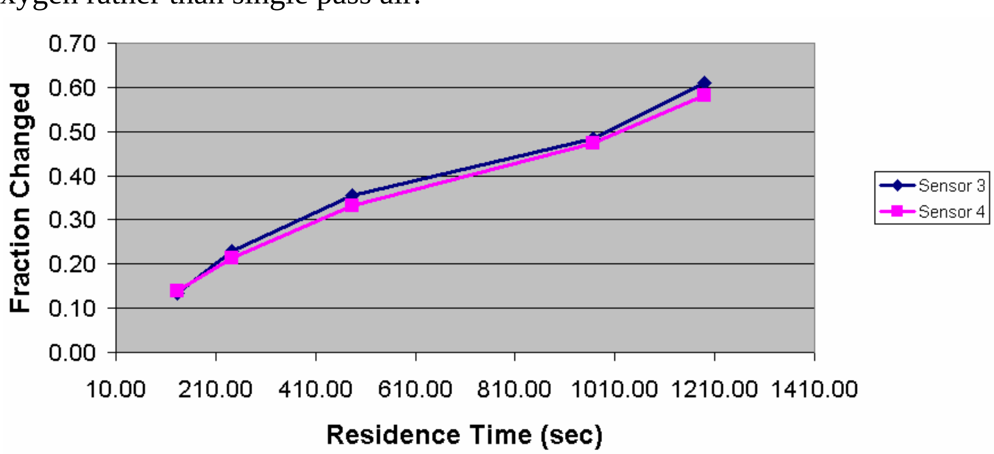


Figure 8. Performance of a length of silicone tubing in the laboratory test chamber. Residence time is the length of time a parcel of gas remains in the test tubing. This was calculated based on the measured flow rate and the manufactures reported inside diameter of the tubing.

4 AUTOMATED INFILTRATION GALLERY CONCEPT & DESIGN

The design of the system was to pump ground water up to the vadose zone and apply water to the vadose zone with a maximum rate that would capture all of the ground water passing beneath the infiltration gallery and redirecting the water through the vadose zone occupied by the infiltration gallery (Figure 9). In the design it was necessary to keep all of the water below grade to avoid the regulatory requirement for treating water that passed through the plane of the surface before it could be reinjected into the ground. As mentioned previously, the depth of aquifer treated has the saturated thickness above a confining clay (approximately 4 m). Laboratory studies have suggested that there needs to be at least two meters of soil above the water table to be able to maintain sufficient oxygen to achieve the treatment desired. The depth to the water table was greater than 2 meters so it was not considered necessary to build a mound to operate the system. If water could be applied at the maximum design rate, essentially all of the groundwater would be redirect through the vadose zone. The maximum design flow rate was based on previous work at the site mentioned above.

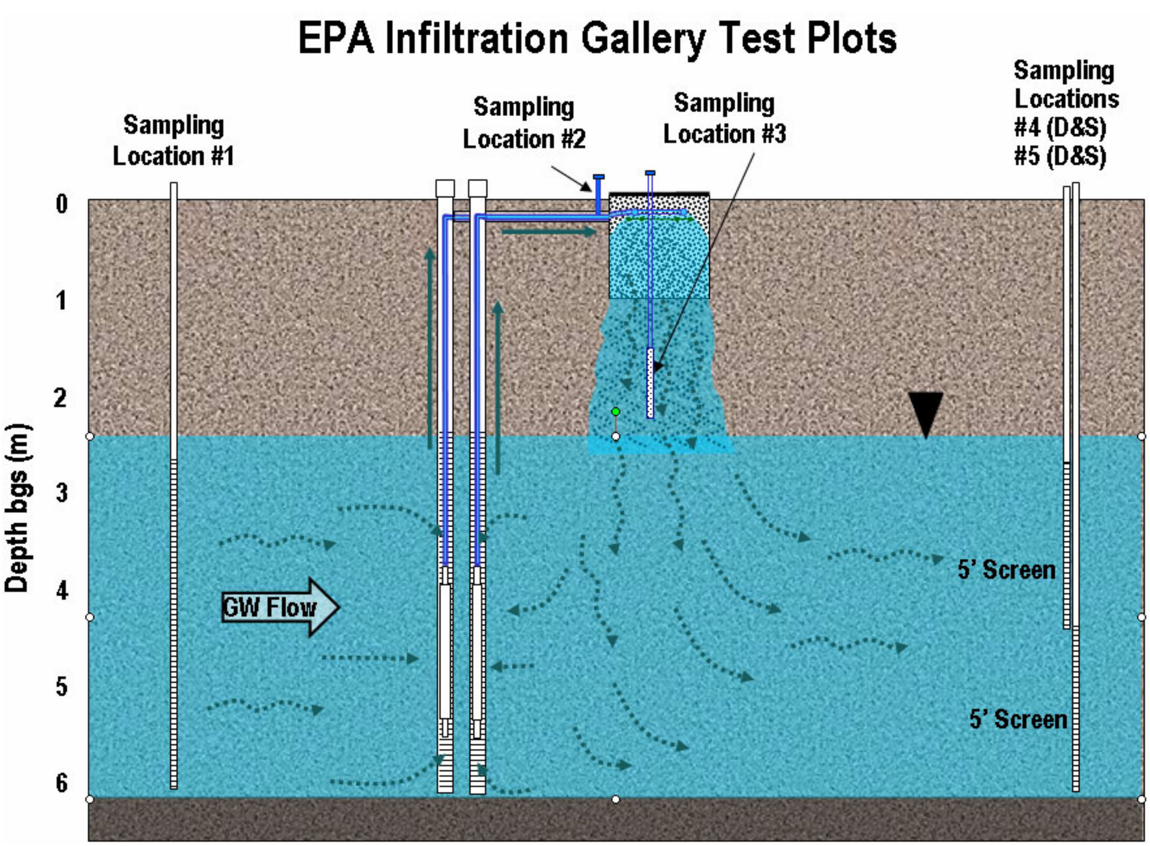


Figure 9. Cross section of infiltration gallery as originally proposed.

4.1 Infrastructure

4.1.1 Ground water extraction and distribution

Each plot had four non-contact bladder pneumatic pumps that were placed in 2" monitoring wells immediately upgradient from the infiltration gallery. The pneumatic extraction pumps delivered water to a pressure tank and maintained the pressure between 9 and 10 psi (Figure 10). Pneumatic pumps are operated by pulses of air such that, when pressurized, they force water into the pressure tank and, when they vent, the pump is allowed to refill the pump body. Two check valves are used in the pump, one to prevent back flow into the well from the pressure tank and the other to prevent water in the pump body from reentering the well bore. The pump cycle used was 40 seconds of pressure and 20 seconds of venting. The pumps operated on this cycle whenever the pressure in the pressure tank was less than 9 psi, but once the cycling started they continued to pump until they exceeded 10 psi.

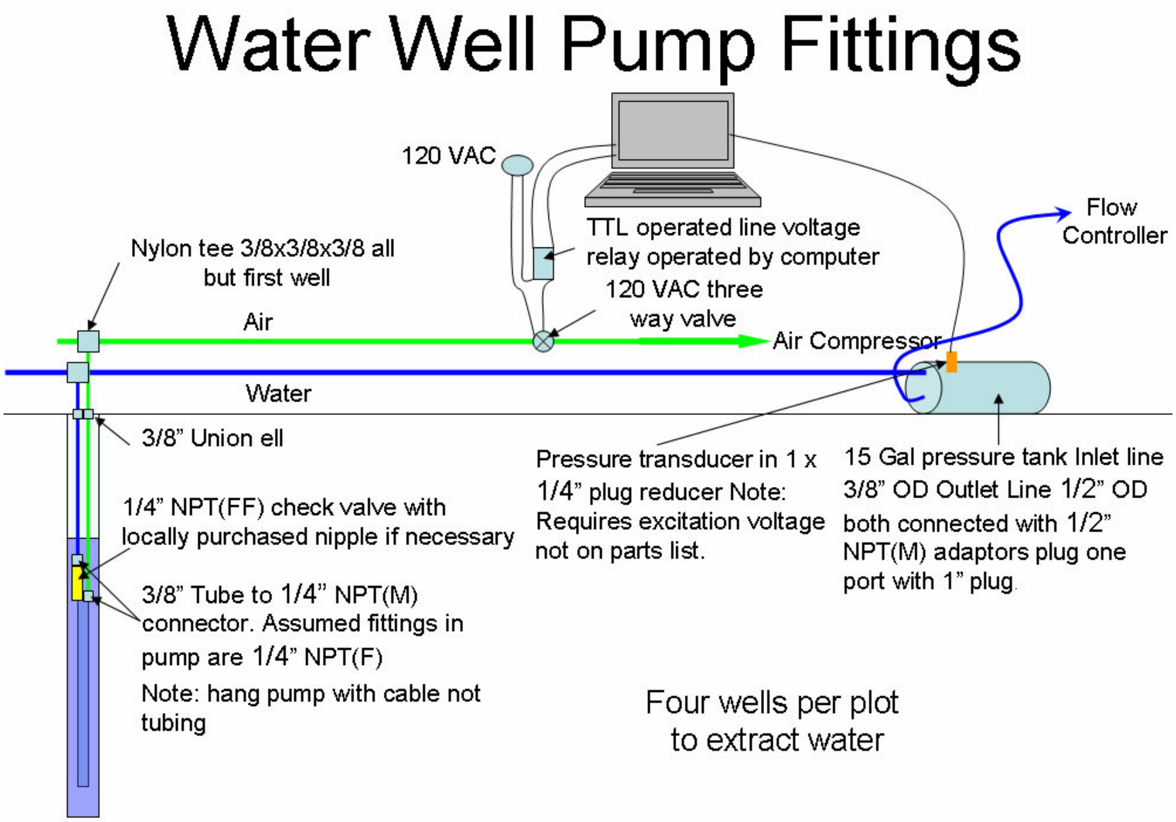


Figure 10. Water Extraction System.

Water was distribution (Figure 11) to the plots cyclically. This was done for two reasons: First, if flow is continuous and slow, bacteria begin to develop in the piping and soon plug the distribution system to a point where water could not be delivered to the infiltration gallery. This was observed at a previous study with fuel contaminated ground water performed at Dover AFB and is a common cause of failure of infiltration systems. The second reason for pulses of water is to improve the distribution of the water. If water is delivered at a slow rate, then it would be delivered only to the emitter locations and penetrate to ground water relatively rapidly along preferential flow paths. The amount of

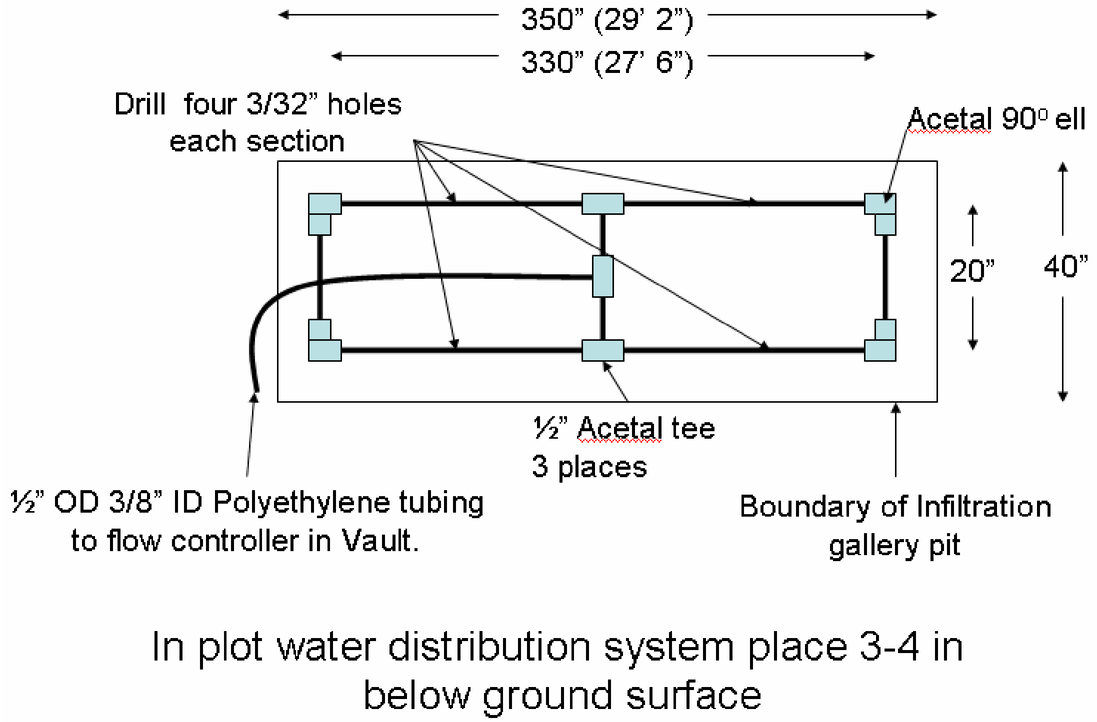


Figure 11. Water distribution system.

contact with vadose zone oxygen would be small and the system would likely fail due to either poor oxidation of the chemicals or successive plugging of the formation. Applying water at instantaneous rates, greater than the saturated hydraulic conductivity of the soil, will cause the water to spread horizontally filling the void spaces in the vadose zone improving the areal distribution of the water. The system was designed to deliver pulses of water followed by two pulses of air after each pulse of water. The air was used to drain the distribution system and minimize the potential of the distribution fouling from bacterial growth, at the same time adding some oxygen to the vadose zone. Oxygen from this source is not a requirement for system operation. The major components of the system are shown in Figure 12.

In the initial design, a programmable metering pump was used to deliver water to the plot, rather than a flow controller shown in Figure 12. This system failed due to flooding of the site. When the system was rebuilt, a vault was added such that the top was above the high water level of the previous years flood, but still keeping the fluids below ground elevation. A sump pump was added to remove any water that might penetrate the vault's walls. At this time the decision was made to replace the pump with a less expensive flow controller. An advantage of the flow controller was that it included a flow totalizer, a stop valve when the control signal was set to zero, and a flow rate output that could be logged by the computer system. The flow controller used has some restrictions related to the amount of grit that could be tolerated and the manufacture suggested that the water be filtered prior to passing through the flow controller. Filtering was the only treatment provided before the ground water was injected into the infiltration gallery. It was necessary to manually record the totalized flow. Manual recording of totalized flow occurred on an approximately weekly schedule. The totalizing flow meter would frequently roll over and begin counting from zero between reading cycles. The data was manually incremented to reflect these changes. Another disadvantage of the flow controller was that it relied on the pressure developed by the pneumatic extraction pumps for power to deliver water to the plots. The pneumatic pumps used were not designed to operate at high pressures. The pumps were operating near the maximum head that could reliably operate over long periods of time and were not able to operate at a rate that would supply the maximum design flow.

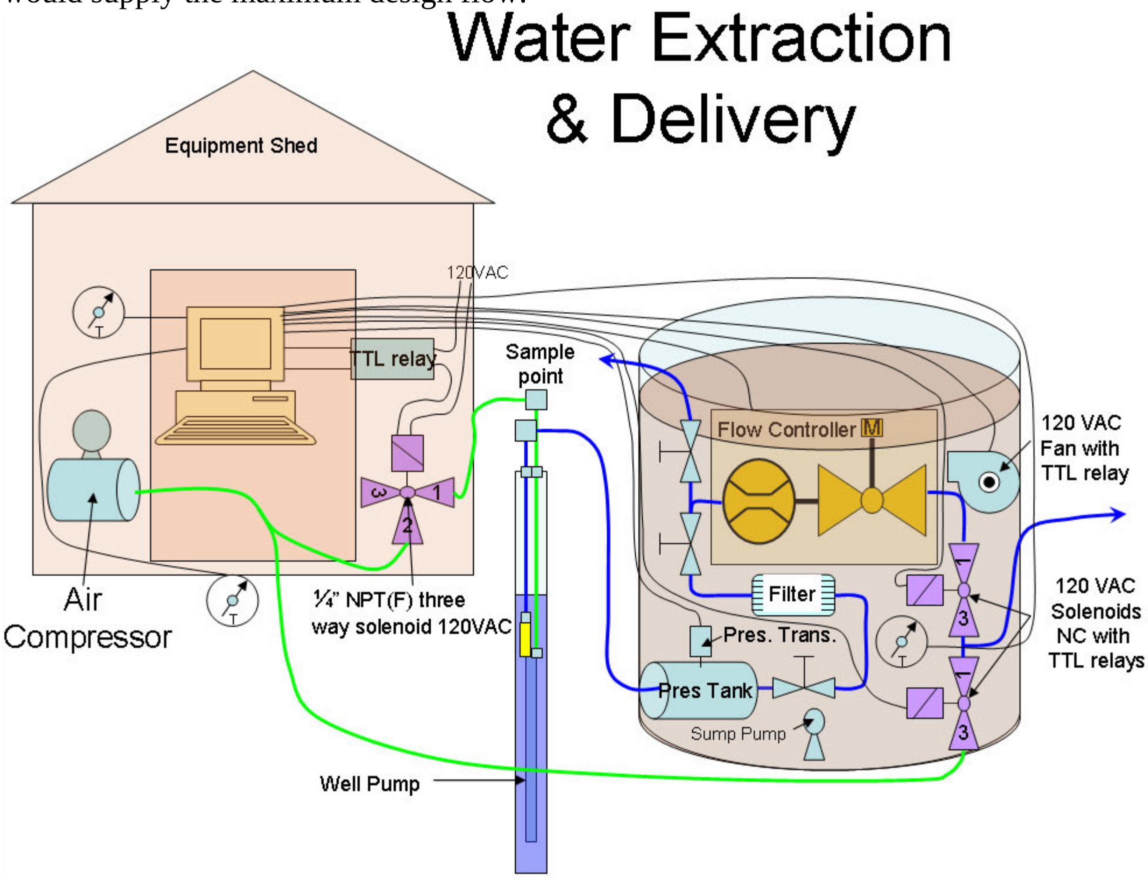


Figure 12. Water extraction and delivery system.

4.1.2 Infiltration Gallery Oxygen Monitoring

The control system was to apply all of the groundwater flowing past under the infiltration gallery to the infiltration gallery as long as there was sufficient oxygen in the vadose zone to avoid dropping the oxygen content in air in equilibrium with the soil moisture below 15%. If the oxygen fell below 15% the system stopped applying water until sufficient oxygen was observed. The system was set up such that under program control the oxygen sensors could be presented one of three different gases (Figure 13). Two of the gases were used for calibration. One calibration gas was atmospheric air representing a reference 20.5% oxygen, and the other was GC grade nitrogen which served as a

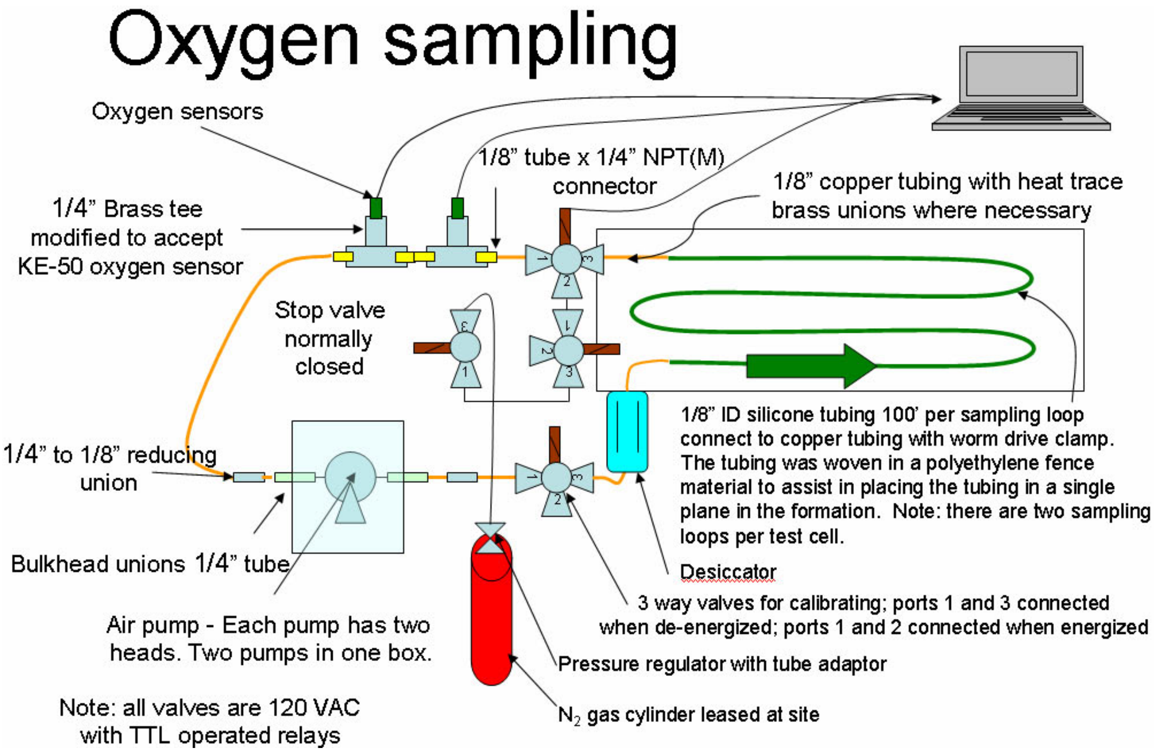


Figure 13. Gas sampling system.

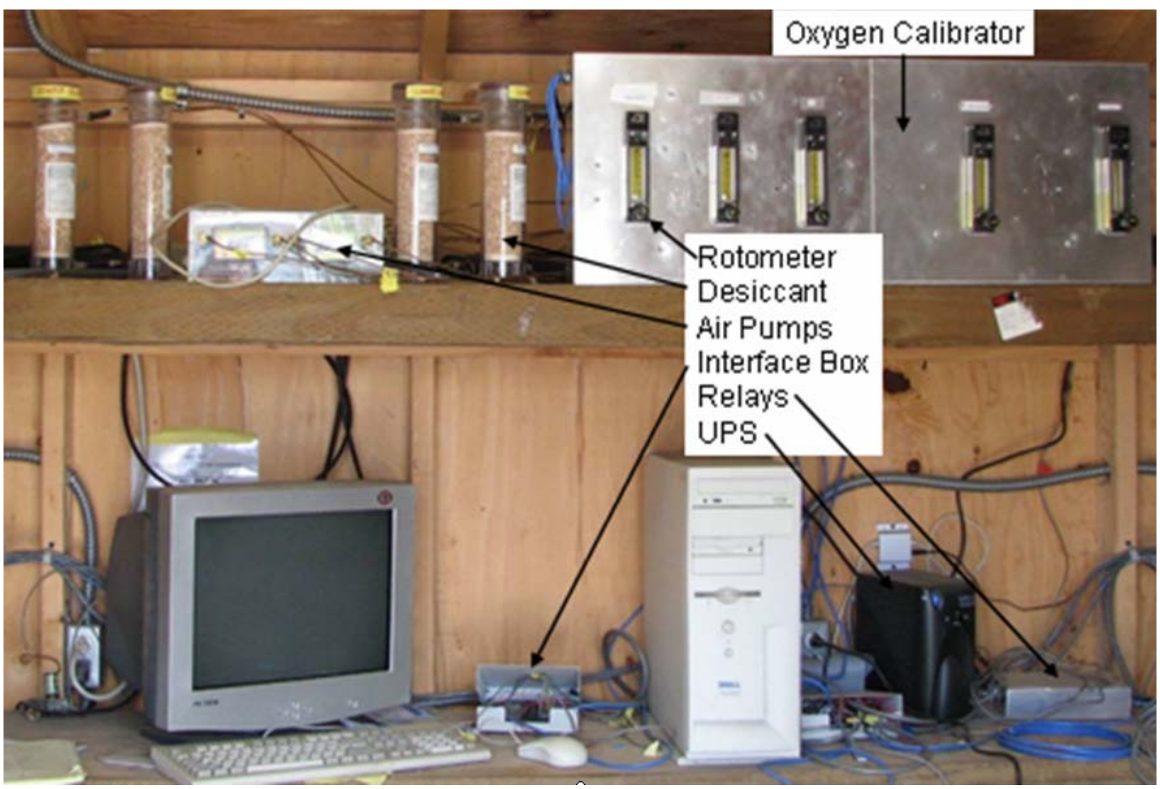


Figure 14. Instrumentation housed in the equipment shed. Note, the rotometers in the photo were not in use.

reference 0% oxygen. The third gas was the gas in the sampling loop. The sampling loop was isolated and closed when the system was being calibrated. Initially a rotometer, as seen in Figure 14, was used to measure gas flow, but continuous maintenance was required to obtain reasonable reading. The rotometers were removed from the system to minimize potential leaks. In the de-energized state, gas can flow through the solenoid valves used in the gas sampling loop was from port 3 to port 1. Normally all of solenoid valves used in the gas sampling loop remain de-energized. When the system calls for a calibration, the gas diffusion membrane was isolated from the system and atmospheric air was pumped by the oxygen sensors for a predetermined amount of time. The voltage was recorded for all of the sensors every 20 seconds for four minutes. Then nitrogen was pumped through each individual sensor pair one at a time. This was done in case there was a leak in one of the systems. Data was again recorded every 20 seconds for four minutes and all four loops were sequenced. The final measurements, air, representing 20.5% oxygen, and nitrogen, representing 0% oxygen, were used to calculate slopes and intercepts used to evaluate the oxygen levels.

4.2 Program Control Logic

The system was controlled using an INTEL 386™ based PC. System changes were slow and there was no need for a high speed computer. The computer was placed on a small UPS to handle short term power interruptions and ran continuously throughout the project using Windows 2000™ operating system without any interruptions. The program was written in Visual Basic 6 with a supplemental library of subroutines supplied by Measurement Computing®. The remote interface was based on SYMANTEC pcANYWHERE™. Communication between the PC and the Internet was provided wirelessly by an IP provider, Sky River. There were two primary functions for the onsite hardware: data collection and storage and process control. To accomplish these functions, two Measurement Computing® interface cards were added to the PC. The two PCI cards were a DAS6033 with 64-channels of 100 kS/s analog I/O with eight digital I/O, and two counter/timers, and a PCI-DDA02/16 with 2 Channels of 16-Bit Analog Output and 48 Bits of Digital I/O. In the upper right hand corner of Figure 14, is the oxygen calibrator. The rotometers were initially in the system but removed because they continuously needed to be cleaned to give reliable results. The pumps that moved the gas through the diffusion membrane and desiccators are in the upper left of the picture. In the lower right of the figure, in addition to the computer, are two breakout boxes where the control signals and sensors were attached to the computer, the small UPS system, and one relay box. The digital I/O ports functioned as output switches. The power up state for the ports was a digital high. The interfaces to the AC loads were zero voltage relays capable of handling loads up to 20 Amp. The relays were turned on by a digital high. The power up state of the Measurement Computing® interface cards was also a digital high. To avoid potential problems during power up sequencing, an additional transistor switch was used between the computer card and the output relay to present the relay with a digital low when the digital output port was set high and a digital high when the digital output port was set low. This is counter intuitive, requiring special attention during the preparation and evaluation of the code.

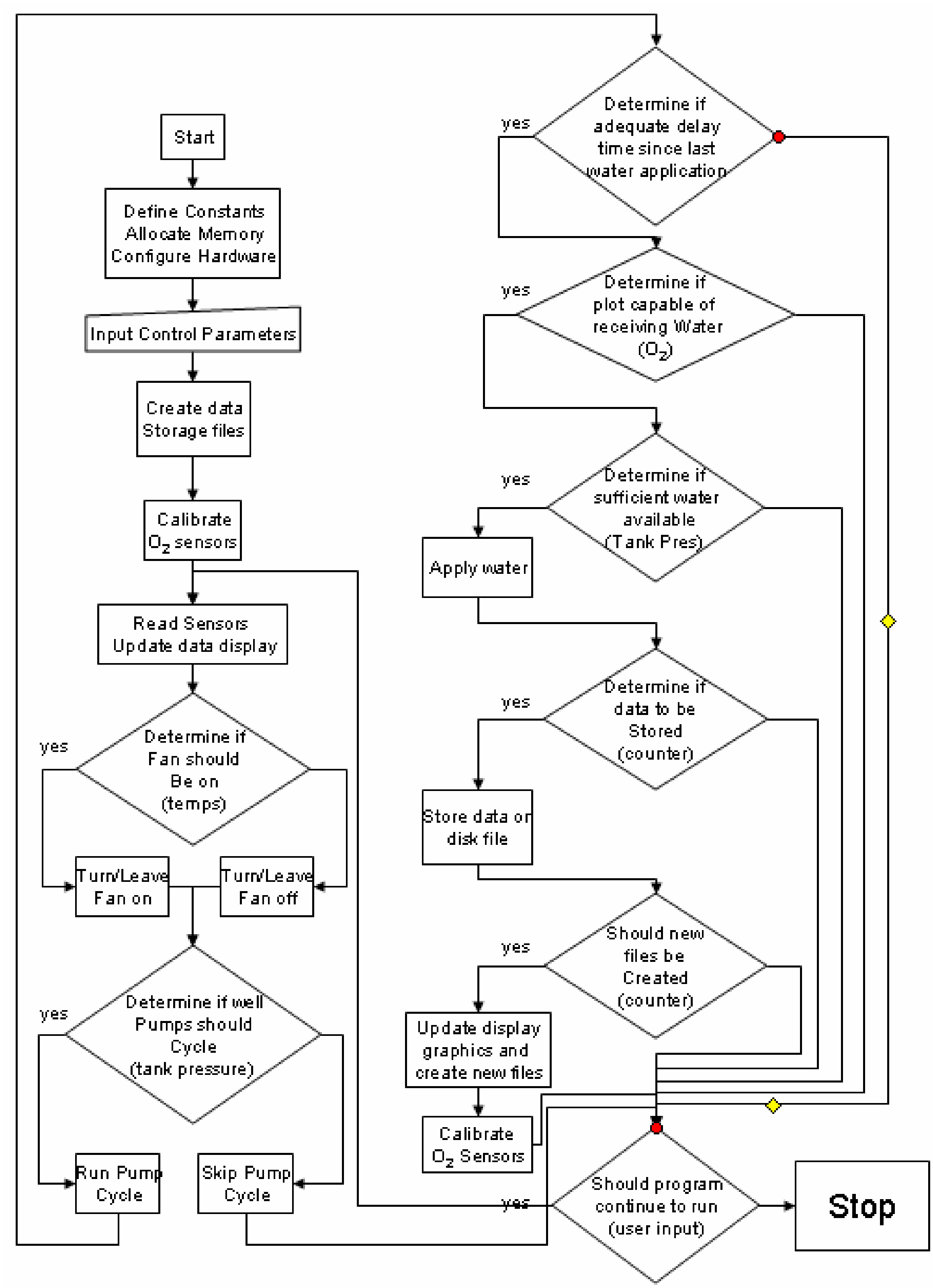


Figure 15. Flow diagram of the computer program that operated the system.

A simplified flow diagram of the control logic of a single test plot is provided in Figure 15. There were two test plots called north and south. For a given plot, the pneumatic extraction well pumps, four on each plot, were controlled by a single three wave solenoid valve and operated in parallel. During initialization the user can adjust the timing by setting the number of seconds for filling the bladder (solenoid valve vents to the atmosphere) and the number of seconds the pump discharges (solenoid valve delivers regulated pressure to the bladder). Pump pressure was regulated manually and the pressure was not recorded electronically. During the power up sequence a default value for the timing was provided. After tuning the pumping system to get a good delivery, the default values were reset to the adjusted values. This was done to avoid needing to reset the pumping sequence each time there was a power interruption.

Decision	**True Condition**
Turn on or leave on vault exhaust fan	If Vault temp > 10 ºC and Air temp < Vault temp
Cycle Well pumps	If Pressure tank pressure < 9 psi or if on previous cycle pumps were on and pressure tank pressure < 10 psi
Has there been sufficient delay since last water application	Pump cycle count since last water application ≥ 20
Is there sufficient oxygen for water application	O_2 ≥ 15%
Is there sufficient water in tank to apply water	Tank pressure > 8 psi
Should data be stored	If counter ≥ 10 then reset counter
Should new data file be created	If number of records in data file ≥ 1000
Should O_2 sensors be recalibrated	When new data file is created

There were four pressure transducers monitored; two on each of the pressure tanks. The manufactures supplied calibration curve parameters were stored in a subroutine called "parms.bas". Two different brands were used, one of each brand was placed in each of the pressure tanks. There were four temperature transducers. The temperature transducers were semiconductors with a linear output (LM35). The calibration parameters were stored in the subroutine "parms.bas". One temperature transducer was placed in each of the vaults used to deliver water to the plots, one was in the shed that stored the computer, and air compressor and one was in an air space under the shed to measure ambient air temperature. There were eight oxygen sensors, four measuring oxygen on each plot two at each sampling elevation. Initial calibration parameters were stored in the subroutine "parms.bas" but they were periodically recalibrated under program control as discussed later. The flow controllers had outputs of instantaneous flow rate that were periodically sampled and stored.

Data was stored in engineering units in three different groups of comma separated variable files with names that included the type of data included in the file. Each file name would have the form DATA month, day (number), minute, second, and either blank, cal, or flow .csv. The data file (without cal or flow) stored data in units of ºC, oxygen as %, tank pressure in psig and counters as integers. The file had the format datetime not including seconds, four temperatures (South Vault, North Vault, Shed and Air), eight oxygens (South Low 1, South Low 2, South High 1, South High 2, North Low 1, North Low 2, North High 1, North High 2), South well pump cycle counter, North well pump cycle counter, South Application counter, North Application Counter, South pressure tank pressure, and North pressure tank pressure. The data files with the suffix cal contained the calibration data for the oxygen sensors. The first column was datetime followed by the voltage output of the oxygen sensors in the same sequence as listed above. The last column gave the gas being presented to the oxygen sensors. The final two rows are the calibration parameters determined, based on linear regression.

4.2.1 Fan Control

The vaults could get very warm and go above the maximum operating temperature of the flow controllers. To minimize this impact, fans were installed in the vaults to exhaust air when needed. The fans did not run all of the time since we did not want to create conditions where condensation could occur, potentially shorting out some of the instrumentation. The fans were operated when the air temperature was above 5 ºC and the temperature of the vault was greater than the temperature of the outside air. This should avoid potentially freezing the water in the lines and avoid condensing conditions. No data was collected indicating what part of the time the fans were running.

4.2.2 Extraction Well Control

The extraction wells were operated such that the pumps would begin to cycle when the pressure in the tank fell below the lower set point (9 psi) and then stoped cycling when the pressure went above the upper set point (10 psi). The computer displayed the pressure reported by all of the pressure transducers but only one pressure data point for each pressure tank was recorded in the data file. Once a pump cycle started, the entire cycle was completed before other operations were initiated, including a called program stop.

4.2.3 Plot Water Application

Water was applied to the plots in doses similar to a mound type wastewater treatment system. The objective is to inject sufficient water to cause it to spread horizontally over the infiltration gallery, based on a premise that the horizontal hydraulic conductivity will be greater than the vertical hydraulic conductivity. An individual dose might be 1.0 cm of water. For the study plots this would be 0.05 m^3 or 50 l. The pressure tank was undersized for optimum operation. While the pressure tank should have been able to deliver the required volume, it wouldn’t have been able to supply the volume with adequate pressure to maintain the flow rate desired. The flow controller was limited to a maximum flow rate of 0.5 l/min and would have required 10 minutes of delivery for a single cycle. A less than optimum compromise was used. The application duration was reduced to a single pump cycle (1 minute) and the flow rate was reduced 0.3 l/min. The control flow rate was set in the subroutine parms.bas and the controller attempted to match the desired rate. Application occurred no more frequently than once every N cycles where N is set in the parms.bas subroutine. Water application would occur only if the pressure in the pressure tank were greater than the minimum pressure set point – 1 psi.

4.2.4 Data Storage

Every M[th] observation was stored in the data files. After O records were stored on a data file new files were created. The size of a file and the frequency of recording were set in the subroutine parms.bas.

4.2.5 Data Display

The field operator was provided with a display (Figure 16) showing the operating conditions and information about how the system was working. The display was divided

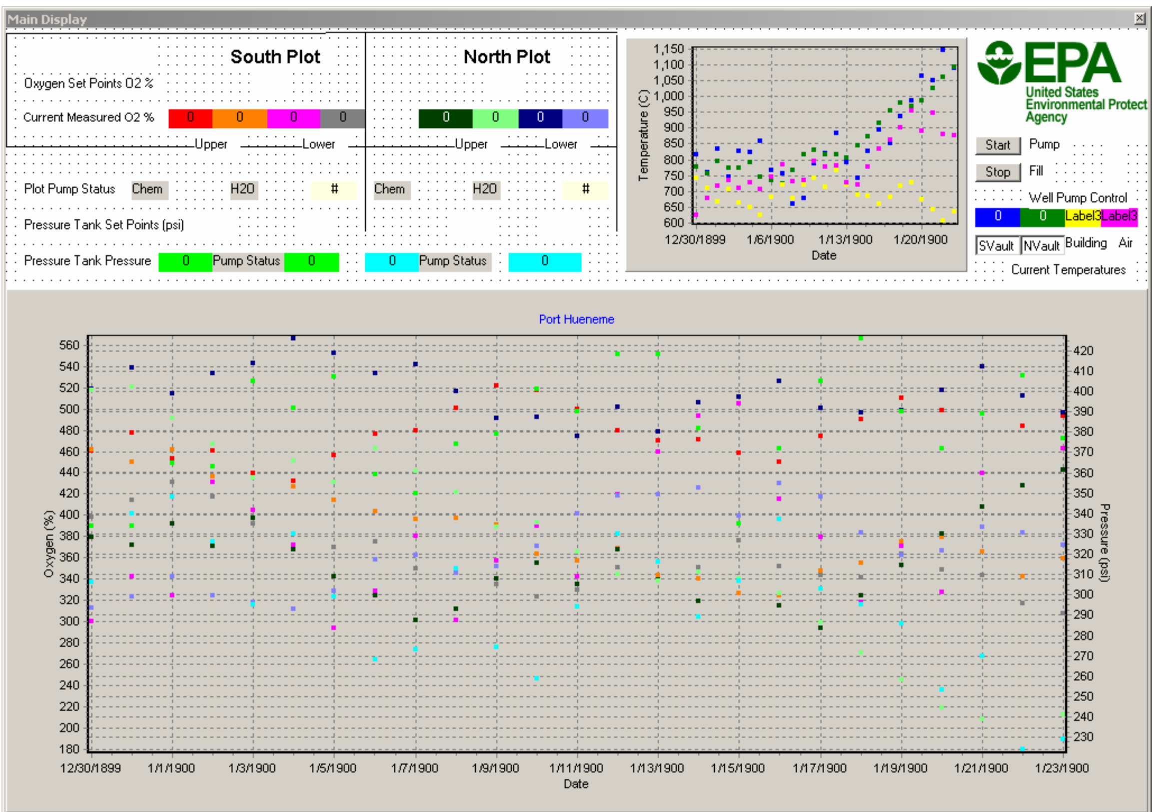


Figure 16. Computer monitor display of the program. This screen capture was obtained in programming mode. The values displayed are placeholders not data.

into several regions. In the upper left hand corner there was a flag that would indicate the system was recalibrating the oxygen sensors. The color would change indicating which sensor was being calibrated. When the oxygen sensors were being calibrated the display reported the voltage being measured rather than the oxygen percentage in the gas. After calibration the display reported oxygen in percent. There were status flags indicating conditions that might cause water delivery to be skipped, as well as the approximate amount of time before the next plot water delivery cycle. The times were updated only once every pump extraction cycle. In the upper right hand corner, the pump pressurization (pump) and venting (fill) times were input as well as the start stop buttons for the program. There were also two graphical displays that were continuously updated. One display was the temperature history and the other displayed oxygen and pressure tank pressure. Each time a data file was closed the displays were erased and then regenerated with current data. The displays were auto ranging. The values shown in the Figure 16 are not actual data, just space fillers.

5 PERFORMANCE EVALUATION

A very large amount of data was collected during the entire performance period. To permit discussing the results, we will focus on the data collected between November 19, 2006 and April 8, 2007.

5.1 Hardware Performance

The system performed as expected with few design difficulties. The pumping system delivering water from the formation to the pressure tanks worked with no down time for the entire duration of the study. The control system was able to maintain the pressure in the pressure tank with in the expected range. The redundant pressure transducers measuring the pressure in the pressure tank were not recalibrated during the test. There were a total of three pressure transducers on each tank, two were reported to the display and one was stored in the data base table continuously. They continued to track each other over the duration of the project. The third transducer located in the flow controller was checked when senior staff were at the site and gave similar values, but this information was not recorded in the data log. No changes were required in the computer code that operated these pumps. The code was designed to allow setting the pressurization and venting time by the operator. Once this timing was established it was not re-adjusted during the study. The code was modified such that the default values for pump timing in the code were the same as those determined experimentally as giving the best flow rates. In this way, if there was a power outage, the program could be restarted automatically when the computer rebooted without human intervention. The solenoid valves that supplied and discharged the compressed air functioned continuously without a problem. Power outages did take place during the project, but either the small UPS system or the auto restart function in the code kept the system running, appearently continuously.

We had one problem with the oxygen calibration system, the regulator supplying the nitrogen for calibration developed a leak, at a rate of one cylinder per week. The regulator was replaced and an additional solenoid stop valve (Figure 13) added to the system which solved the problem. Once the leak was stopped, very little nitrogen gas was consumed. The oxygen sensors showed very little drift over the period of the experiment. Calibration repeatability was within ±2% when the nitrogen was available for calibration. The loop measuring the plot oxygen initially had problems with water condensing in the line, even with a desiccant column in the loop. Water would condense as it cooled during the winter and the air pump was not able to force the water out of the lines. A heat trace all along the 1/8” copper tubing, as indicated in Figure 13, was added to the system and no further problems were observed.

The major difficulty was with the equipment that was installed below ground level. The initial installation was splash proof but was unable to withstand flooding. During one of the storm events, everything was flooded and destroyed. A different type of vault was installed where the top of the vault was above the high water level and a sump pump installed to remove water that might enter through the small penetrations in the vault walls required for electrical, pneumatic and hydraulic fittings (Figure 17). To conserve

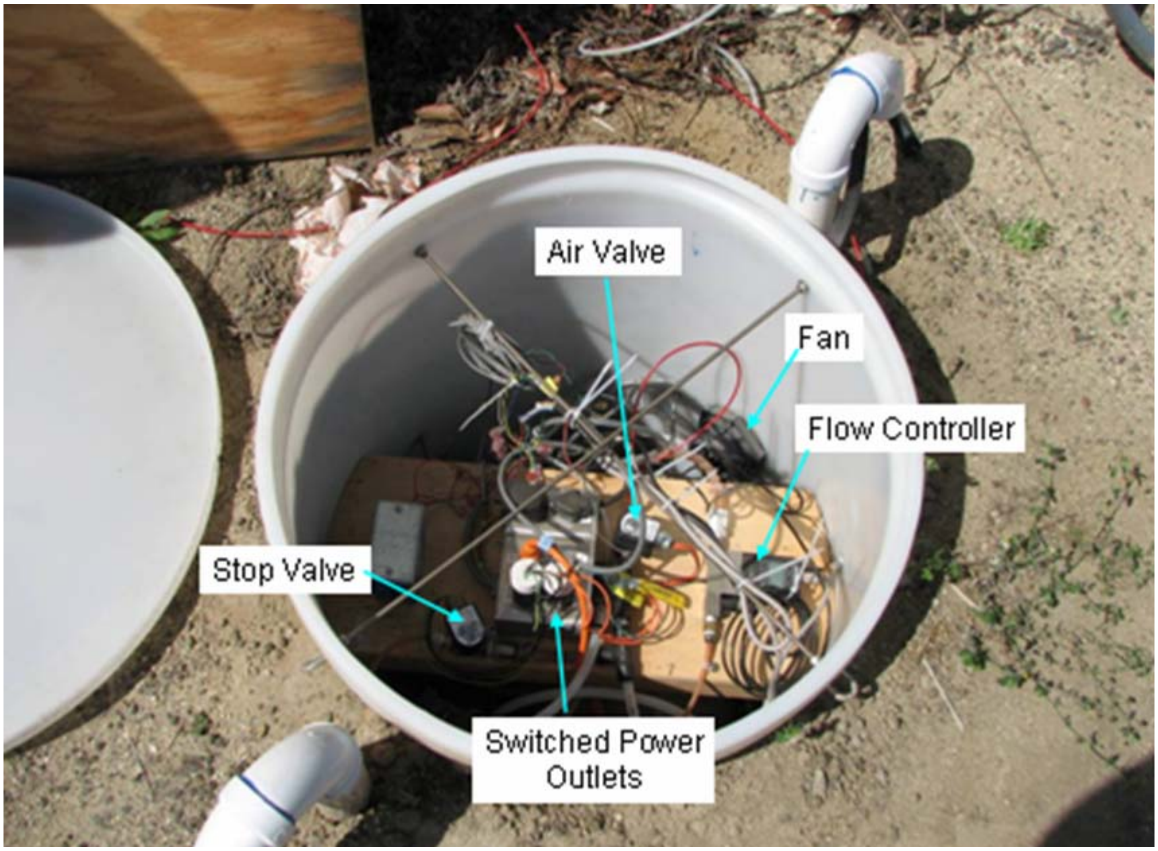


Figure 17. Equipment in the vault.

funds, a flow controller, rather than a metering pump, was used in the vault when the system was rebuilt. The flow controller relied on the pressure in the pressure tank to deliver water to the plot. There was insufficient pressure to deliver the design flow rate, even when the distribution system was above ground. To achieve the desired flow rate with these flow controllers, it would be necessary to have a greater tank pressure. The pneumatic well pumps being used were not capable of supplying this pressure, suggesting the addition of a pressure pump or using the metering pumps as initially installed in the system. Once the modifications were made, the system continued to function at a reduced flow rate for the remainder of the study period and there was little reduction in the flow through the system over time. This suggests that draining the distribution lines with compressed air after each water application was successful in controlling the biomass buildup in the distribution system. This had been a problem in earlier work where the lines were not drained. The modified system did survive major flooding events without any problems.

5.2 Chemical Performance

Prior to the installation of the system, it was believed that methyl tertiary butyl ether (MTBE) was present in the formation water. The site had been used previously for MTBE research and data was available suggesting sufficient MTBE to warrant a study to evaluate the degradation of MTBE. However, after the first few rounds of sampling it was discovered that there was little MTBE remaining and studying the chemical response would not provide definitive information as to the field application of the technology for the removal of MTBE. It was felt that it would beneficial to continue monitoring the system to verify how robust the system would be mechanically and to learn what improvements might be made to improve the viability of the technology. Laboratory studies have repeatedly shown that under oxidizing conditions, fuel components degrade in a soil environment. In this study, we have shown that a system can be reliably installed and operated remotely and when a reduction in sensed oxygen occurs the flow rate to the plots is reduced. Very few modifications were required. A study needs to demonstrate the total system performance where dissolved fuel components are present. We were not able to depress the oxygen level in the vadose zone using the ground water at this site. This is partially due to the quality of water being applied and partially due to the inability to apply excess water to the system. The infiltration gallery was scaled so that the maximum design flow would be 5% of the saturated hydraulic conductivity. It was felt, based on research on infiltration galleries with municipal waste water, the system should be capable of accepting this loading rate without biologically fouling. No biological fouling was observed. The flow controllers could not apply this rate to the infiltration gallery with the pressure available. The metering pumps initially installed could have met the design maximum flow rate, but may not have been able to pump at a sufficiently high rate to depress the oxygen in the formation since there was little oxygen demand remaining in the water.

5.3 Temperature

The temperature was monitored at four different locations: in each of the vaults, inside the shed where the computer was stored and in the airspace below the shed. The primary purpose for measuring the temperature at these locations was to determine if the temperature ever exceeded the operational range of the instrumentation (0 – 40 ºC). A sample of the experimental data is shown in Figure 18. Other data looks similar but is not shown. It was necessary to provide shading for the vaults to avoid overheating even with an exhaust fan that was used under program control whenever the air temperature below the shed was below the air temperature in the vault and above 5 ºC. This range was used to avoid condensation in the vault and potentially freezing the water distribution system. The temperature sensors, LM35s, proved to be very reliable and robust. The program did not set an alarm for out of range temperatures. This did not create a problem at Port Hueneme but might be a problem at other locations with wider temperature swings.

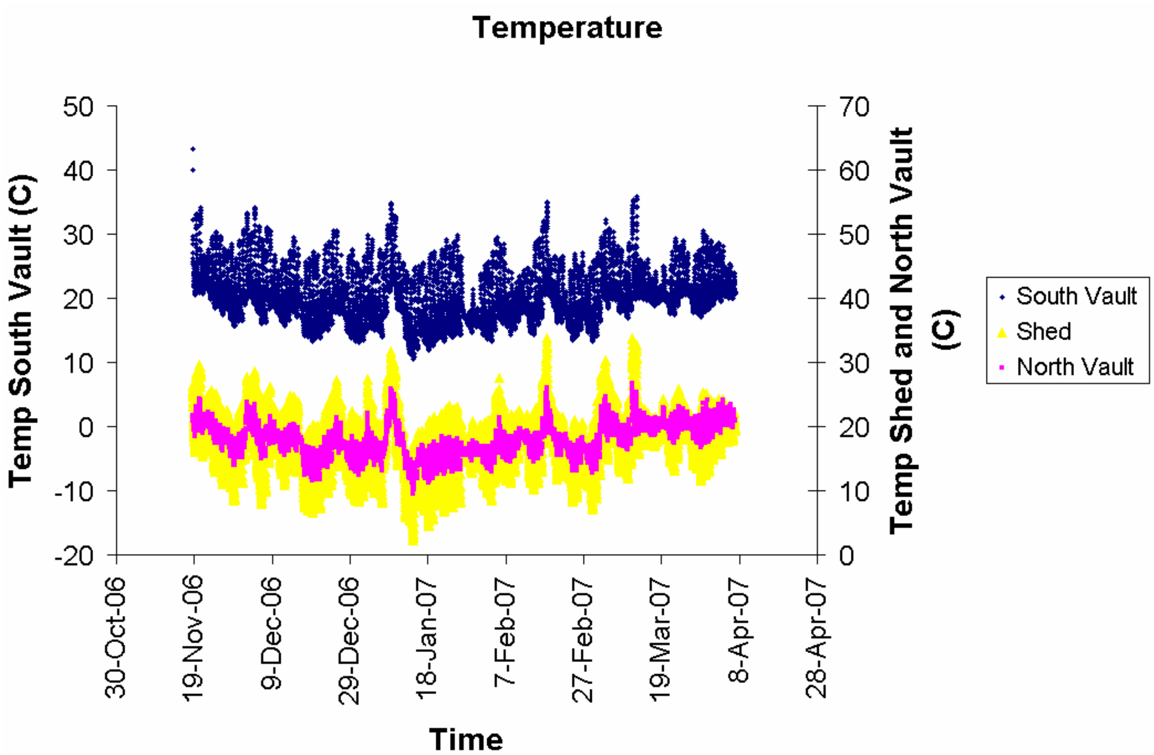


Figure 18. Temperature fluctuations at three different locations. Note, the south vault temperature fluctuated much more than the north vault. The north vault was shaded by the equipment shed much of the time.

5.4 Pumping Required to Maintain Pressure Tank Pressure

There were two check valves in each of the four pumps on each of the plots. The bottom check valve allowed water to enter the pump when the pump bladder was vented to atmosphere and stoped the flow when the pump was pressurized. The top check valve was at the top of the pump body and prevented backflow into the well from the water column and pressure tank. Under ideal conditions each pump cycle should produce the same amount of flow and, if all pumps are operating under the same head conditions, all of the pumps should discharge at the same flow rate. As a crude performance metric the number of times the pumps were cycled was recorded for each of the plots (Figure 19). The north plot showed a roughly constant flow rate of 400 to 500 cycles per day while the south plot was showing a slow decline but only about half as many cycles except for one period about February 7 2007 when both plots showed a decrease in pumping rates. The cause of this pumping rate decrease is discussed later. The maximum possible number of pump cycles per day was less than1440 or one cycle per minute minus the time used in other computer operations such as data collection, display and storage. At the end of the project when the pumps were removed from the formation there were major differences in the appearance of the pumps (Figure 20). Some of the pumps had a significant amount of what appears to be iron oxide on the outside and water that drained from the pumps was rusty in appearance. From previous studies not referenced, the formation water was known to contain a significant amount of reduced iron. From the observation, it appears that some of the pumps were leaking air into the well bore causing the iron to oxidize and fall out of solution. This may suggest that more frequent maintenance of the pumps was required and checking the seals to make sure there was no air leakage.

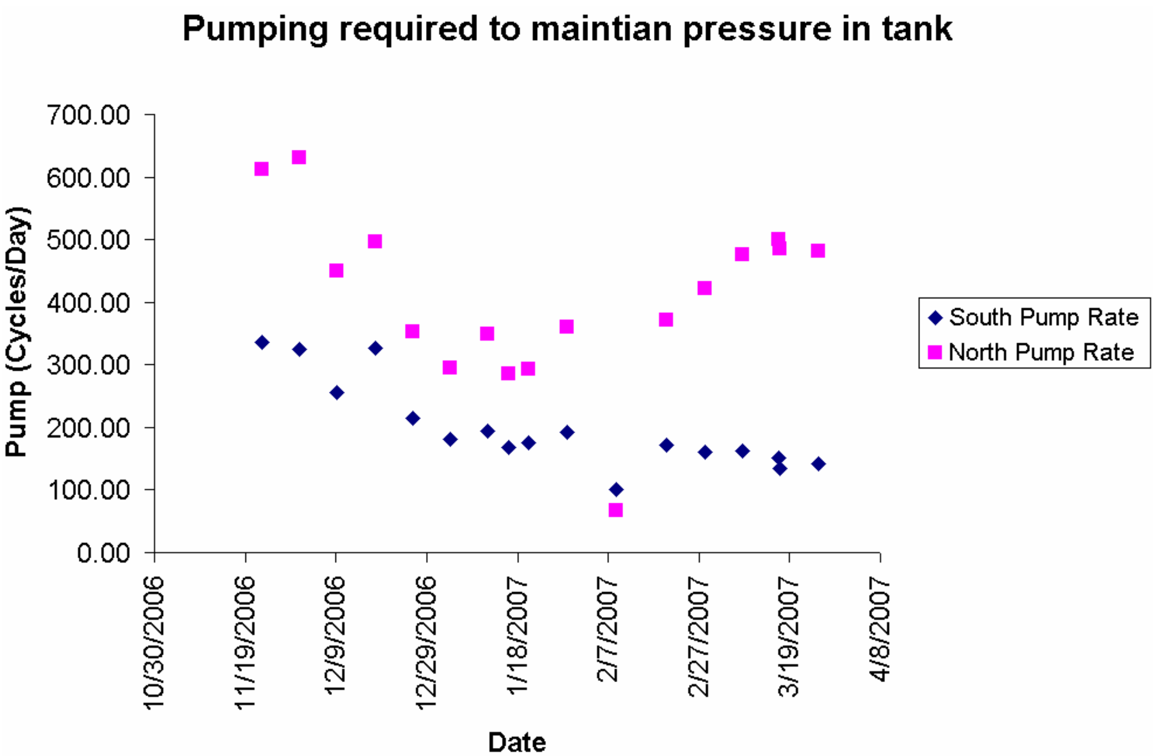


Figure 19. Average pump cycles per day required to maintain pressure in tank located in vault.

Figure 20. Pumps after approximately 18 months of continuous operation.

5.5 Oxygen Sensor Calibration.

The oxygen sensors were recalibrated after every M x O (10000) data points were collected or before each data file was created. The number of data points was fixed by parameters passed to the operating program. Figure 21 shows the changes in the calibration curves. The period of time between 11/19/2006 and 1/17/2007 was the time period when the regulator was leaking and the default oxygen calibrations were used. Thus the slope (377) and intercept (-1.06) remain constant. The repair to the system occurred 1/17/2007. After that point in time the calibrated sensors reported the same oxygen level for both plots except for the short period of time when there was apparently a small amount of dirt that made the regulator inoperative (Figure 22). During this period of time in attempting to calibrate the algorithm used made the intercept go more negative and the slope to increase dramatically yielding false data. If the there had been no drift in the oxygen sensors, both the slope of the calibration curve and the intercept of the calibration curve should have been constant. The period of time illustrated shows significant deviation for a short period of time. It is believed that a piece of dirt caused the regulator to malfunction. Touching the regulator resolved the problem. The problem was discovered almost immediately but it took some time before the technician could respond to the problem.

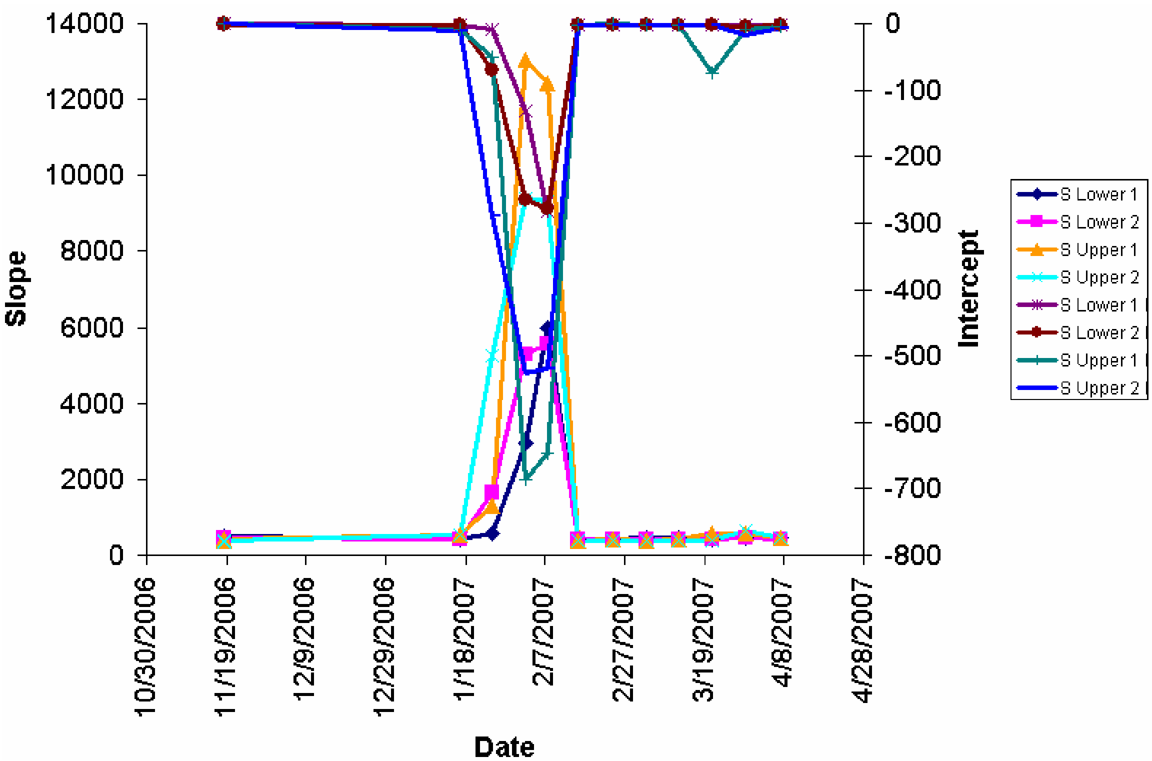


Figure 21. Oxygen calibration parameters illustrating period of time when oxygen was not available. The first four curves are the slope of the four south plot oxygen sensors and the next four curves are the respective oxygen sensors intercept. The nominal value for the intercept is -1.

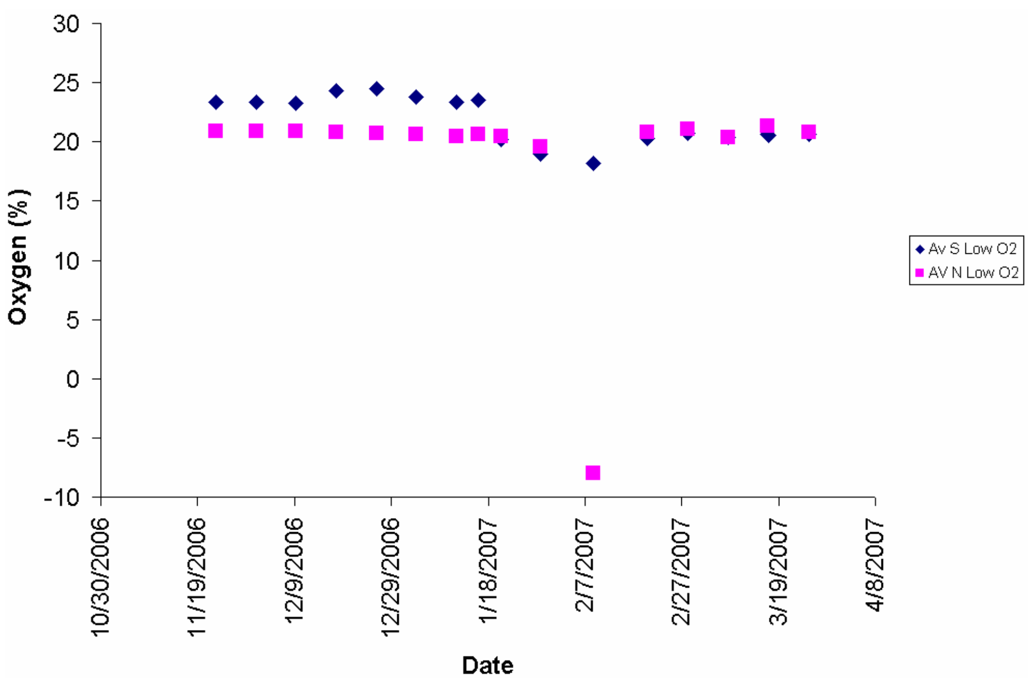


Figure 22. Measured oxygen based on calibration curves.

To improve the maintenance and reliability, a pressure transducer should be added to the nitrogen line and add an alarm that would alert the operator of a fault automatically bypassing the recalibration of the oxygen sensor until the alarm had been reset. The nitrogen cylinder should also be moved into a secure location to minimize tampering. When nitrogen was present, the oxygen measurement system calibrated with good stability. The oxygen sensors were still functioning reliability five years after the date of manufacturer. The manufacturer suggests that the life expectancy for this series of oxygen sensors is ten years. No data was collected beyond the five year period.Looking at the average oxygen measured over the same period of time (Figure 22) we see there is a major dip in the reported oxygen level. The observed apparent dip in the oxygen occurred at the same time the calibration curves were apparently in error because there was insufficient nitrogen to bring the sensor down to the reduced oxygen level. No provision had been made in the code to set off alarms or to use the default settings when there was an apparent fault in the data being collected.

5.6 Water Applied to the Plot.

One of the primary evaluation criteria was the ability to deliver water to the test plots. Previously, at Dover AFB, a test was performed but the system failed due to an accumulation of biomass in the plumbing system. The Dover AFB system failed after a few months of operation and it was not possible to continue delivering water to the plot. In this study we did not apply as much water to the plots as the initial design called for. The target upper was 72 l/d to each of the plots. We applied less than 9 l/day. There were two primary reasons for the reduced delivery. First, the extraction pumps did not deliver the amount of water anticipated. The pumps ran continuously to extract water from the formation but were not capable of maintaining the required pressure in the pressure tank to deliver water at every potential cycle. Second, there was either backflow from the pressure tank into the formation between pump cycles or a system gas leak. This was observed as a drop in pressure in the pressure tank when there was no flow. Finally, the flow controllers need more pressure to deliver the maximum flow through the distribution system. The pumps were tested in the field to see if their pumping rate was adequate. The testing pumped into an open vessel with no back pressure other than elevation head and pressure drop through the plumbing. As tested, back flow could not occur and the pumps appeared to be delivering adequate flow. A pressure flow curve was not developed and hindsight suggests that additional pumping capacity would have been beneficial. The flow controllers were unable to deliver the maximum flow even when first turned on without any potential back pressure caused by system fouling. This limitation was recognized at the beginning of the test. It was decided to continue, knowing that it would not be possible to deliver the maximum amount of water desired. If flow controllers are to be used, a pressure pump should be added to the system to increase the pressure available to deliver the water or a different type of extraction pump should be used that is capable of delivering higher pressure reliably. We had two methods of estimating flow, one based on the flow totalizer, and one based on the flow rate measurements and the theoretical duration of a flow cycle. Flow rate was measured only one time during each flow cycle and assumed to be constant throughout the cycle. If the flow controller operated as expected, the flow rate would not be a true constant as the control valve will continuously "seek" the set point and there will be overshoot and undershoot problems. The duration of the flow cycle was set by an internal clock as a fixed amount of delay. In addition to the delay there were other activities going on that required time. The actual cycle time is not known, but should have been very reproducible. It is known that the actual time is greater than the program delay (1 minute). The estimated flow, based on the flow rate will, therefore, be less that the actual total. The flow totalizer counted to 99.999 l before turning over and restarting. Technicians manually recorded the totalized flow. Measurements were made on a weekly schedule. It was frequently necessary to use judgment to decide how many times the counter had turned over between recorded values. A portion of Cumulative flow data is shown in Figure 23. The slope of the cumulative flow data is the flow rate. A linear

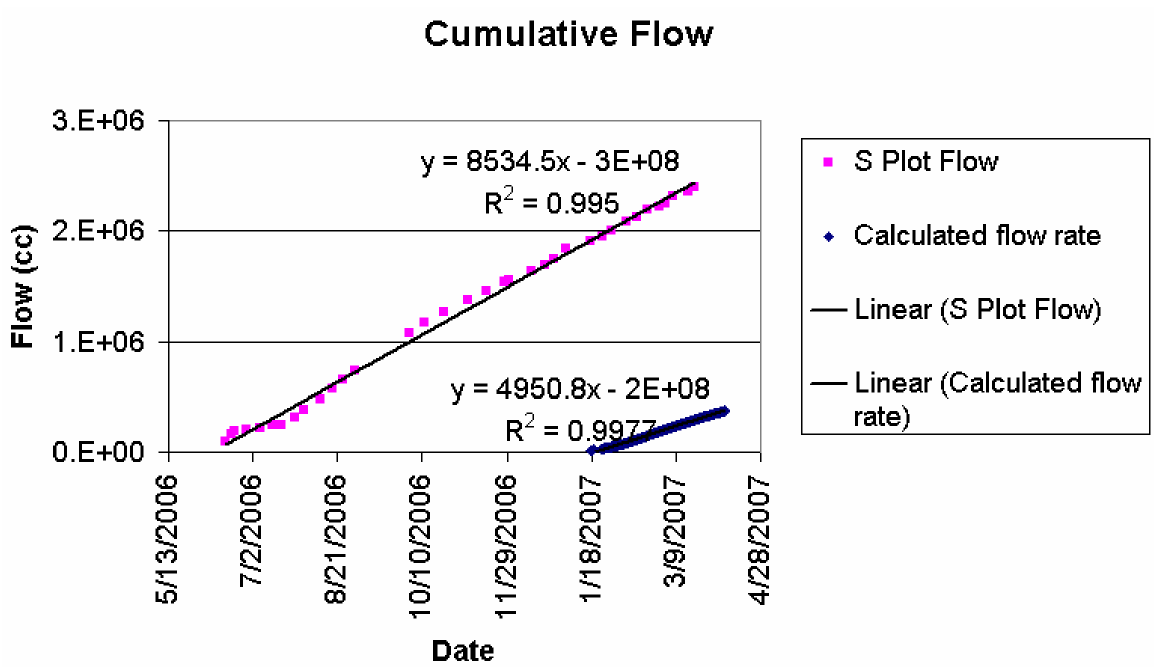


Figure 23. Cumulative water application to south plot using two different metrics.

function was fit to the experimental data and the results presented on the figure. The starting times are different so only the slopes not the cumulative flow values are significant. The totalized flow suggests an average flow rate of 8.5 l/day this is based on the measurements made using the flow totalizer that was part of the flow controller and manually recorded by a field technician. Calculating the totalized flow based on one instantaneous measurement flow rate and the theoretical time of a flow pulse suggests an average flow rate of 5 l/day. Both of the plots are essentially linear suggesting there is little if any evidence of system failure due to biological fouling. This suggests the system design which included draining the distribution lines with compressed air was successful in achieving one of the major concerns of system reliability under field operation. Only minor design adjustment would be required to achieve the desired maximum pumping rates

6 RECOMMENDED NEXT STEPS

It has been demonstrated in the field that a reliable system can be constructed and operated remotely and requires little field operator intervention. Several items can be improved over the initial installation to improve the overall system. The gas supply used for the oxygen sensor calibration should be maintained in a limited access area and a pressure transducer should be added to the system that can be monitored remotely. The control program should be modified and the old calibration curve parameters for the oxygen sensors should be kept when signals indicate calibration gas is not present. As implemented in the current design, we used a continuous loop for oxygen measurement. To avoid moisture collection in the system we used a heat trace along the copper lines to and from the plot and a dryer. The system required a significant amount of maintenance. Two alternative potential improvements are: a refrigerated dropout pot with automatic drain in the closed loop system, or a one pass system. In either case, it is likely that the heat trace would still be required to avoid condensation in the gas lines. It is also recommended that a mass flow controller be used in the gas sampling system if one pass air is used or a mass flow meter if a closed loop system is used. By controlling and measuring the flow, confidence will be gained that the gas being sampled is actually what is passing through the diffusion membrane. A controller would also be a requirement if one pass air is used to insure the appropriate residence time to obtain representative oxygen measurements.

The water delivery system needs to be capable of supplying a known flow at known flow rates with varying amounts of back pressure and supply pressure. We did not achieve these objectives in the present study. Either the flow controller should be replaced with a metering pump or a pressurizing pump should be added that can maintain adequate pressure during a delivery cycle.

Material costs should be considered when designing a future demonstration. Since it is not reasonable to make a site-independent cost estimate, it is recommended that the project be designed in a two step process, once a tentative site has been selected. The first step would be to determine how large an infiltration gallery would be required, based on areal extent of the design and the hydraulic properties of the formation. Once this information is known, it is possible to make a reasonable cost estimate. Some of the items that should be considered up front are:

1. Wells for the extraction of groundwater and the evaluation of the flow rate of the groundwater including a determination of the direction of flow.
2. A pre analysis of the flow rate at the site can be done with the wells that are installed to extract water. A test similar to a borehole dilution test would be adequate. There are several other alternatives that should be considered.
3. Well pumps – The pumps needed are low flow pumps that will provide long term service with minimum maintenance. Non-contact bladder pumps for 2” wells appear to be adequate, if the pumping head is not too great. Air lift pumps may be required for greater depths. An analysis needs to be made comparing the

difference between the cost of installing 2” monitoring wells and using more expensive pumps or 4” wells and less expensive pumps.

4. Equipment Shed for storing equipment - will heat and/or cooling be required?
5. Air compressor. - The size of the air compressor will be dependent on whether air is used just for draining the lines or also used for water extraction from the formation. An oilless pump should be used to avoid the potential transfer of oil from the compressor to the formation.
6. Automatic Drain to remove condensate will be needed for the air compressor.
7. PC - A low cost slow computer that is reliable is all that is needed. Care must be taken regarding the temperature range anticipated. Hard disk drives are generally limited to 0 – 40 ºC. In dusty environments fans can have a problem it would be better to have a unit that did not require moving parts.
8. Interface cards for the PC. - Water condensing environmental conditions may require special coatings on the computer cards at some locations.
9. Pressure tank for the extracted well water - a bladder isolated pressure tank is recommended that has a storage volume equal to two or more application cycle volumes to the infiltration gallery. It may be desirable to include a deairing valve upgradient of the pressure tank if there is any potential of introducing air into the water. Additional air will potentially encourage bacterial growth in the pressure tank and this should be avoided as much as possible.
10. Plot delivery pump and infiltration gallery size (the distance along the flow path of the groundwater) are critical items that must be scaled to the flow rates desired. This will be dependent on the ground water flow rate and system size.
11. From a design standpoint, there was no indication that there were problems applying water at 5% of the saturated conductivity of the infiltration gallery fill material. An additional constraint is that the hydraulic conductivity of the natural formation material must be capable of accepting the entire application rate of the water.
12. The silicone tubing used in the study was reinforced tubing with 1/8” nominal ID with 1/8” copper lines to the oxygen sensors It might be better to use ¼” ID tubing with ¼” copper lines to deliver the sampled gas to the oxygen sensors. This will give more time for equilibration.
13. A self regulating heat trace is considered essential to avoid water condensation in the oxygen measurement system. Manufacturers will assist in the design of an appropriate size. Take into consideration that rodents like to eat the insulation of wire. Metal conduit may be appropriate where the wire is exposed.
14. There was nothing special about the solenoid valves used. They were all 120VAC and activated using TTL input optically isolated zero voltage switches. The switches were used to isolate the valves from the computer system.

The system needs to be scaled to a full sized system and implemented at a site with existing contamination. The performance monitoring needs to be evaluated over a period of time of 18 months or more. Longer times would be desirable to make sure there is no bio-fouling of the infiltration gallery. It will likely require a few months for the system to acclimate to the water composition and then several additional months for the biomass to grow to a quasi steady state level. Scaling of the system will require different sized pressure tanks, and flow controllers. As demonstrated all of the groundwater fluids were maintained below ground surface. This is possible to accomplish, however, it would be preferable if the controls can be placed in a protected area above the ground surface where there is less likelihood of flood damage. Temperature extremes also need to be considered. Where the test was performed we did not have problems keeping equipment within its operating temperature range 0-40ºC. Most locations in the Unites States may have difficulty in the winter time maintaining temperatures above freezing unless there is some sort of heat. The computer, in addition to all of the small water lines, needs to be protected from freezing. In the South there may be problems of overheating of equipment. One alternative for the computer's hard disk is to replaced it with a solid state disk with wider environmental constraints.

7 REFERENCES

Borden, R.C., Daniel, R.A., LeBrun IV, L.E. and Davis, C.W., 1997. Intrinsic biodegradation of MTBE and BTEX in a gasoline-contaminated aquifer. Water Resources Research, 33: 1105-1115.

Bradley, P.M., Landmeyer, J.E. and Chapell, F.H., 1999. Aerobic mineralization of MTBE and tert_butyl alcohol by stream-bed sediment microorganisms. Environmental Science and Technology, 33: 1877-1879.

Bradley, P.M., Landmeyer, J.E. and Chapell, F.H., 2002. TBA Biodegradation in Surface-Water Sediments under Aerobic and Anaerobic Conditions. Environ. Sci. and Technol., 36(19): 4087-4090.

Church, C.D. et al., 1999. Effects of environmental conditions on MTBE degradation in model column aquifers. In: USGS (Editor), U.S. Geological Survey Toxic Substances Hydrology Program, Charleston, South Carolina.

Cline, P.V., Delfino, J.J. and Rao, P.S.C., 1991. Partitioning of aromatic constituents into water from gasoline and other complex solvent mixtures. Environ. Sci. and Technol., 25(5): 914-920.

Converse, J.C. and Tyler, E.J., 2000. Wisconsin mound soil absorption system: sitine, design and construction manual. University of Wisconsin.

Erickson, J. and Tyler, E.J., 2001. A model for soil oxygen delivery to wastewater infiltration surfaces.

Javanmardian, M. and Glasser, H.A., 1997. In situ biodegradation of MTBE using biosparging, Division of Environmental Chemistry. American Chemical Society, Preprint of Extended Abstracts, pp. 424.

Kaharoune, M., Kharoune, L., Lebault, J.M. and Pauss, A., 2002. Aerobic degradation of ethyl-tert-butyl ether by a microbial consortium: selection and evaluation of biodegradation ability. Environ Toxicol Chem, 21(10): 2052-2058.

Mackay, D.M. et al., 1999. Field studies of *in situ* remediation of an MTBE plume at site 60, Vandenburg Air Force Base, California. In: A. Stanley (Editor), Petroleum Hydrocarbons and Organic Chemicals in Ground Water: Prevention, Detection, and Remediation. National Ground Water Association and American Petroleum Institute, Houston, Texas, pp. 178-188.

Mackay, D.M. et al., 2000. In situ treatment of MTBE by biostimulation of native aerobic microorganisms. In: U.S.EPA (Editor), Workshop on MTBE Biodegradation. U.S. EPA, Cincinnati, Ohio, pp. 87-91.

OEHHA, 1999. Public health goal for Methyl Tertiary Butyl Ether (MTBE) in drinking water.

Schmidt, T.C., Schirmer, M., Weiß, H. and Haderlein, S., 2004. Microbial degradation of methyl tert-butyl ether and tert-butyl alcohol in the subsurface. J. Contam. Hydrol., 70(3-4): 173-203.

Tyler, E.J., 2001. Hydraulic wastewater loading rates to soil.

USGS, 2006. Volatile Organic Compounds in the Nation's ground water and drinking-water supply wells, USGS Circular 1292.

Wilson, G.J., Richter, A.P., Suidan, M.T. and Venosa, A.D., 2001. Aerobic biodegradation of gasoline oxygenates MTBE and TBA. Water Science and Technology, 43(2): 277-284.
Wilson, R.D., Mackay, D.M. and Scow, K., 2002. In situ MTBE biodegradation supported by diffusive oxygen release. Environ. Sci. and Technol., 36: 190-199.
Wilson, R.D. et al., 1999. Laboratory-scale evaluation of in situ aerobic MTBE biodegradation options for Vandenberg Air Force Base, California, Petroleum Hydrocarbons and Organic Chemicals in Ground Water: Prevention, Detection, and Remediation, Houston, Texas, pp. 167-176.

APPENDIX 1. Wiring Codes DAS6033

DAS6033 Wiring

Pin	Signal Name	Application	Range	Cable	Wire Pair	Color	Variable Name	Comment
1	LLGND							
2	CH0 In Hi	Temp South Vault	0-10 VDC		B/B	Blue	Temp(0)	
3	Ch0 In Lo			2-A	R/B	Black		
4	CH1 IN HI	Temp North Vault	0-10 VDC		B/B	Blue	Temp(1)	
5	CH1 IN LOW			4-A	R/B	Black		
6	CH2 IN HI	Temp Shed	0-10 VDC				Temp(2)	
7	CH2 IN LO							
8	CH3 In HI	Outside Temp Air	0-10 VDC				Temp(3)	
9	CH3 IN LO							
		O2 South Plot			Y/B	Yellow	O2(0)	
10	CH4 IN HI	Low Sensor 1	0-1 VDC		Y/B	Black		
11	CH4 IN LO				R/B	Red	O2(1)	
		O2 South Plot			Y/B	Black		
12	CH 5 IN HI	Low Sensor 2	0-1 VDC		R/B	Black	O2(2)	
13	CH 5 IN LO				Y/B	Black		
		O2 South Plot			W/B	White	O2(3)	
14	CH 6 IN HI	High Sensor 1	0-1 VDC		Y/B	Black		
15	CH 6 IN LO							
		O2 South Plot			G/B	Green	O2(4)	
16	CH7 IN HI	High Sensor 2	0-1 VDC		Y/B	Black		
17	CH7 IN LO				Blue/B	Blue	O2(5)	
18	LLGND				Y/B	Black		
		O2 North Plot			Blue/B	Black	O2(6)	
19	Ch 8 In HI	Low Sensor 1	0-1 VDC					
20	CH8 IN Low				Y/B	Black		
		O2 North Plot			G/B	Black	O2(7)	
21	CH9 IN High	Low Sensor 2	0-1 VDC					
22	CH9 IN LO							
		O2 North Plot						
23	CH10 IN HI	High Sensor 1	0-1 VDC					
24	CH10 IN LO							
		O2 North Plot						
25	Ch11 IN HI	HighSensor 2	0-1 VDC					
26	CH11 IN LOW			6-A	Y/B	Black		
27	CH12 IN HI	Pressure Transducer South Plot	0-10 VDC					0.5 VDC = 0 psi, 5.5 VDC = 15 psi
					G/B	Green	Pres(0)	
28	CH12 IN LO			1-A	G/B	Black		
29	CH13 IN HI	Pressure Transducer	0-10 VDC	3-A	G/B	Green	Pres(1)	

30	CH13 IN LOW				G/B	Black		
31	CH14 IN HI	Chem Feed Pump South Plot	0-10 VDC					Return Signal From Pump
32	CH14 IN LOW							
33	CH15 IN HI	Chem Feed Pump North Plot	0-10 VDC					Return Signal From Pump
34	CH15 IN LOW							
35								
36	D/A Out 0	South Flow Controler	0-5 VDC	1-A	B/B	Blue	SWaterPOut	
37	D/A Gnd				B/B	Black		
38	D/A Out 1	South Chemical Metering Pump	0-10 VDC				SChemPOut	
39	PC +5VDC							
40								
41								
42								
43								
44								
45								
46								
47								
48								
49								
50	GND							
51	LLGND							
52	CH16 IN HI	Flow Meter Rate South Plot	0-10 VDC		W/B	White		Use 500 ohm resistor
53	CH16 IN LO			1-A	W/B	Black		
54	CH17 IN HI	Flow Meter Totalizer South Plot	0-10 VDC		Y/B	Yellow		Use 500 ohm resistor
55	CH17 IN LO			1-A	Y/B	Black		
56	CH18 IN HI	Flow Meter Rate North Plot	0-10 VDC		W/B	White		Use 500 ohm resistor
57	CH18 IN LO			3-A	W/B	Black		
58	CH19 IN HI	Flow Controler Totalizer North Plot	0-10 VDC		Y/B	Yellow		Use 500 ohm resistor
59	CH19 IN LO			3-A	Y/B	Black		
60	Ch20 In HI	Old Pressure transducer	0-10 VDC	2-A	G/B	Green		
61	CH20 In LO				G/B	Black		
62	CH21 IN HI	Old Pressure transducer	0-10 VDC	4-A	G/B	Green		
63	CH21 IN LO				G/B	Black		

64						
65						
66						
67						
68	LLGND					
69						
70						
75						
76						
77						
78						
79						
80						
81						
82						
83						
84						
85	DIO0	South Vent Fan		R/B	Red	FanBit(0)
86	DIO1	North Vent Fan	5-A	R/B	Black	FanBit(1)
87	DIO2	South Compressed Air	2-A	Y/B	Yellow	PlotAirBit(0)
88	DIO3	North Compressed Air	4-A	Y/B	Yellow	PlotAirBit(1)
89	DIO4	South Stop Valve		Y/B W/B	Yellow White	PlotStopBit(0) PlotStopBit(1)
90	DIO5	North Stop Valve		G/B	Green	O2CalBit(0)
91	DIO6	Oxygen Calibration 4 valves				
92	DIO7	O2_ N2 selector 2 valves	5-A	Blue/B	Blue	O2CalBit(1)
93						
94						
95						
96	GND			Y/B,W/B,G/B ,Blue/B	Black	
97						
98						
99						
100	GND					

APPENDIX 2. Oxygen Sensor Wiring Codes

O_2 Sensors

	Sensor Color	Wire Pair	Wire	Comment
South Upper				
	Sensor Color	Wire Pair	Wire	Comment
1	Red	orange/White	Orange	1 is closest to the panel
	Black		White	
2	Red	Green/White	Green	
	Black		White	
South Lower				
1	Red	Brown/White	Brown	
	Black		White	
2	Red	Blue/White	Blue	
	Black		White	
North Upper				
1	Red	orange/White	Orange	
	Black		White	
2	Red	Green/White	Green	
	Black		White	
North Lower				
1	Red	Brown/White	Brown	
	Black		White	
2	Red	Blue/White	Blue	
	Black		White	

APPENDIX 3. Oxygen Calibrator Box Wiring Codes

Relay Box Wiring

Relay(s)	Valve	Terminal	Signal Cable	Pair	Color	Power Cable	Comment
Vault relays in vault							Comment
	Compressor Air Control Normally Closed						
1-8		1			Black	10-B,11-B	Pick Up Black In Vault @ AC Cables going to Valut + 2 Signal Cables
		2			Black		
		3	Vault Cable 2,4	Y/B	Yellow		
		4		Y/B	Black		
Vault Relay in Shed							Duplicate for North and South Plot
	Ventilating Fan						Operate fan when outside temperature is less than inside temperature and inside temperature greater than 10 deg C
2-6		1			Black	9,10A,11A	
		2			Orange	10-A,11-A	
		3		B/W,O/W	Blue, Orange		South Plot Blue, North Plot Orange
		4	5	B/W	White		
	Stop Valve Normally closed						
3-7		1			Black	9	
		2			Red	10-A,11-A	
		3		G/W,brn/w	Green, Brown		Green South, Brown North
		4	5	G/W	White		
				Orange/W	White		5 VDC
				Brown/W	White		5 VDC
Oxygen Monitor Relays in Valve Box							Insulate power terminals
	Three way valves						
	O2 Calibration 2 valves valves to switch from sample to calibration						
4		1			Black		
		2			Red		
		3		Blue/White	Blue		
		4		Blue/White	White		Digital Ground Black in Panel
	O2 Calibration 1 valve N2 or air South Upper						
5		1			Black		
		2			Orange		
		3		Brown/White	Brown		
		4					Common Digital Ground

	O2 Calibration 1 valve N2 or air South Lower			
9	1		White	
	2		Green	
	3	Green/White	Green	
	4	Green/White	White	Common Digital Ground
	O2 Calibration 1 valve N2 or air North Upper			
10	1			
	2			
	3	Orange/White	Orange	
	4			Common Digital Ground
	O2 Calibration 1 valve N2 or air North Lower			
11	1			
	2			
	3	Orange/White	White	
	4			Common Digital Ground
	N2 stop two way valve normally open to pump box			Operate this valve with the valves to switch between sample and calibrate relay 4
	% VDC Supply	Brown/White	White	

APPENDIX 4. DDA02-16 Wiring Codes

DDA02-16 Wiring

Pin	Signal Name	Application	Range	Cable	Wire Pair	Color	Variable Name	Comment
1	Vout 0	North Flow Controler	0-5 VDC		B/B	Blue	NWaterPOut	
2	Analog Gnd			3-A	B/B	Black		
3	V out 1	North Chemical Metering Pump	0-10 VDC				NChemPOut	
4	Analog Gnd							
5								
6								
7								
8								
9								
10								
11								
12								
13								
14								
15								
16								
17								
18								
19								
20								
21								
22								
23								
24								
25								
26								
27								
28								
29								
30								
31								
32								
33								
34								
35								
36								
37								
38								

39			
40			
41			
42			
43			
44			
45			
46			
47			
48			
49			
50	Digital Gnd		
51	Second Port A Bit 7		
52	Second Port A Bit 6		
53	Second Port A Bit 5		
54	Second Port A Bit 4		
55	Second Port A Bit 3		
56	Second Port A Bit 2		
57	Second Port A Bit 1	North Well Pump	WellPumpBit(1)
58	Second Port A Bit 0	South Well Pump	WellPumpBit(0)
59	Second Port B Bit 7		
60	Second Port B Bit 6		
61	Second Port B Bit 5		
62	Second Port B Bit 4		
63	Second Port B Bit 3		
64	Second Port B Bit 2		
65	Second Port B Bit 1	North Chemical Metering Pump	ChemPumpBit(1)
66	Second Port B Bit 0	South Chemical Metering Pump	ChemPumpBit(0)

67	Second Port C Bit 7				
68	Second Port C Bit 6				
69	Second Port C Bit 5				
70	Second Port C Bit 4				
71	Second Port C Bit 3				
72	Second Port C Bit 2				
73	Second Port C Bit 1				FillBit(1)
74	Second Port C Bit 0				FillBit(0)
75	First Port A Bit 7				
76	First Port A Bit 6				
77	First Port A Bit 5				
78	First Port A Bit 4	N2 North High	Or/W	Orange	
79	First Port A Bit 3	N2 North Low	Or/W	White	
80	First Port A Bit 2	N2 South High	Gr/W	Green	
81	First Port A Bit 1	N2 South Low	Br/W	Brown	
82	First Port A Bit 0	Air/Sampling valves & N2 Stop	Bl/W	Blue	
83	First Port B Bit 7				
84	First Port B Bit 6				
85	First Port B Bit 5				
86	First Port B Bit 4				
87	First Port B Bit 3				
88	First Port B Bit 2				
89	First Port B Bit 1				
90	First Port B Bit 0				

91	First Port C Bit 7		
92	First Port C Bit 6		
93	First Port C Bit 5		
94	First Port C Bit 4		
95	First Port C Bit 3		
96	First Port C Bit 2		
97	First Port C Bit 1		
98	First Port C Bit 0		
99	+5 VDC	Br/W	White
100	Digital Gnd	Bl/W,Gr/W	White

APPENDIX 5. Vault Wiring

Vault Signal Level Wiring

Cable	Color	Function	Pair	Color	Comment
	Pressure Transducer				
	Red	+15 VDC input	R/B	Red	
	Black	Gnd Input	R/B	Black	
	Green	+ output	G/B	Green	0.5 VDC = 0 psi, 5.5 VDC = 15 psi
	White	- output	G/B	Black	
	Shield				
	Flow Controller				
	Red	+ Analog In	B/B	Blue	0 - 5 VDC
	Blue	Analog In Common	B/B	Black	
	Orange	+ Analog Flow Out	W/B	White	4 ma = no flow, 20 ma = 2000 ml/min
	Violet	- Analog Flow Out	W/B	Black	
Cable 1/3-B	Yellow	+ Controller Out	Y/B	Yellow	4 - 20 ma
	Brown	- Controller Out	Y/B	Black	
	Green		B/B	Black	
	Temperature Transducer				
		Supply +5VDC	R/B	Red	
		Common	R/B	Black	
		Signal Out	B/B	Blue	
	Digital Switch Compressed Air				
		Signal	Y/B	Yellow	
		Return	Y/B	Black	
	Old Pressure transducer				
Cable 2/4-B	Green	+ Output	G/B	Green	Use 15 VDC power from Cable 1/3
	White	- Output	G/B	Black	

APPENDIX 6. Port_Hueneme_Display Code

```
 Option Explicit
 'Last revised 3/12/2006 CGE
Const Board1 = 1              ' Board number 1 is the DAS6033
Const Board2 = 2              ' Board number 2 is the DDA02
Const Direction = DIGITALOUT   ' program digital output
Const NumPoints = 2000        ' nunber of points to be scaNN*MMed to determine
        average value
Const FirstPoint = 0          ' first point in buffer to scan to array
Const CBRate = 90000          ' scan rate
Const NN = 1000               ' Number of records in a data file before creating a new file
Const MM = 10 ' Number of measurements+2 between recording in file i.e. if MM = 1
        every third recorded is recorded
Dim SAData(NumPoints)      ' Dimension of scan array to hold input data
Dim MemHandle              ' define a variable to contain the handle for memory
                           ' allociated by windows through cbWinBufAlloc%()
Dim ADData1(21) As Integer             ' raw analog input data Unit 1 16 bit binary
Dim A_Data(21)             ' Scaled to 1 analog data
Dim ADData%(NumPoints)
Dim flowcount(3) As Integer     ' Flow totalizers number of times pump turned on
Dim DataValue%                  ' Binary Number used to turn pumps on and off Board 2
        only
Dim DataValue1%
Dim ULStat%
Dim CycleCountS As Integer  'South plot counter for number of times water has been
        applied to plot
Dim CycleCountN As Integer  'North plot counter for number of times water has been
        applied to plot
Dim O20(NN * MM) As Variant    ' Oxygen Sensor O2(0) South lower sensor 1 CH4
        High pin 10 low pin 11
Dim O21(NN * MM) As Variant      ' Oxygen Sensor O2(1) South lower sensor 2 CH5
        High pin 12 low pin 13
Dim O22(NN * MM) As Variant      ' Oxygen Sensor O2(2) South upper sensor 1 CH6
        High pin 14 low pin 15
Dim O23(NN * MM) As Variant      ' Oxygen Sensor O2(3) South upper sensor 2 CH7
        High pin 16 low pin 17
Dim O24(NN * MM) As Variant      ' Oxygen Sensor O2(4) North lower sensor 1 CH8
        High pin 19 low pin 20
Dim O25(NN * MM) As Variant      ' Oxygen Sensor O2(5) North lower sensor 2 CH9
        High pin 21 low pin 22
Dim O26(NN * MM) As Variant      ' Oxygen Sensor O2(6) North upper sensor 1 CH10
        High pin 23 low pin 24
Dim O27(NN * MM) As Variant      ' Oxygen Sensor O2(7) North upper sensor 2 CH11
        High pin 25 low pin 26
```

```
Dim Pres0(NN * MM) As Variant   ' Pressure Tank Pressure South Plot Pres(0) CH12
        High pin 27 Low pin 28
Dim Pres1(NN * MM) As Variant   ' Pressure Tank Pressure North Plot Pres(1) CH13
        High pin 29 Low pin 30
Dim Temp0(NN * MM) As Variant   ' Temperature inside South Vault Temp(0) Ch 0
        High pin 2 Low pin 3
Dim Temp1(NN * MM) As Variant   ' Temperature inside North Vault Temp(1) Ch 1
        High pin 4 Low pin 5
Dim Temp2(NN * MM) As Variant   ' Temperature inside Shed Temp(2) Ch 2 High pin 6
        Low pin 7
Dim Temp3(NN * MM) As Variant   ' Temperature Air Under Shed Temp(3) Ch 3 High
        pin 8 Low pin 9
Dim FlowR0(NN * MM) As Variant  ' Flow rate FlowR(0) South Plot, FlowR(1) North
        Plot
Dim FlowR1(NN * MM) As Variant
Dim TFlow0(NN * MM) As Variant  ' Totalized Flow TFlow(0) South plot, TFlow(1)
        North plot
Dim TFlow1(NN * MM) As Variant
Dim WellPumpBit(1) As Integer   ' Bit 0 is the south plot bit 1 is the north plot
Dim ChemPumpBit(1) As Integer   ' Bit 0 is the south plot bit 1 is the north plot
Dim FillBit(1) As Integer       ' Bit to tell is pressure tank has been filled
Dim PlotPumpBit(7) As Integer   ' Bit 0,1 is the south plot bit 2,3 is the north plot
Dim FanBit(1) As Integer        ' Bit 0 South Plot, Bit 1 North Plot Ventelating fan for
        vault
Dim PlotAirBit(1) As Integer    ' bit 0 South Plot, Bit 1 North Plot To remove water form
        distribution lines
Dim PlotStopBit(1) As Integer   ' Bit 0 South Plot, Bit 1 North Plot To make sure flow
        stops when off
Dim O2CalBit(1)                 ' Bit 0 three way sample/calibrate valve, Bit 1 three way
                    '   Air/N2 valve
Dim DaTime(NN * MM) As Date
Dim Date_time As Date
Dim Run%
Dim Start
Dim Start1
Dim M As Integer                ' counter to determine when to write data
Dim N As Integer                ' counter to determine when to create a new file
Dim I As Integer
Dim J As Integer
Dim Fillsec1 As Integer
Dim AirVolt(7) As Variant       ' O2 sensor voltage in air
Dim N2Volt(7) As Variant        ' O2 sensor voltage in N2
Dim SumAdata(21)
Dim Data(NumPoints)
Dim Gain As Variant
```

```
Dim Options As Variant
Dim ans
Dim SFlowR      'South flow controller voltage
Dim NFlowR      'North flow controller voltage
Dim CycleCoutnS ' Number of times through the pump cycle south plot
Dim CycleCoutnN ' Number of times through the pump cycle north plot
Dim delay   'seconds of delay
'____________________________________________________________________________
          ____________________________________________________________
'Board 1 DAS6033    Function                    Code Variable          PinNo(s)
'DIO0             South Vent Fan              FanBit(0)             85
'DIO1             North Vent Fan              FanBit(1)             86
'DIO2             South Compressed Air          PlotAirBit(0)         87
'DIO3             North Compressed Air          PlotAirBit(1)         88
'DIO4             South Stop Valve             PlotStopBit(0)        89
'DIO5             North Stop Valve             PlotStopBit(1)        90
'DIO6             O2 Calibration Valve          O2CalBit(0)           91
'DIO7             O2/Air Calibration VAlve       O2CalBit(1)           92


'A/D Channel 0     Temp South Vault              Temp(0)              2,3
'A/D Channel 1     Temp North Vault              Temp(1)              4,5
'A/D Channel 2     Temp Shed                    Temp(2)              6,7
'A/D Channel 3     Air Temp Under Shed           Temp(3)              8,9
'A/D Channel 4     O2 South Plot Low Sensor 1      O2(0)               10,11
'A/D Channel 5     O2 South Plot Low Sensor 2      O2(1)               12,13
'A/D Channel 6     O2 South Plot High Sensor 1     O2(2)               14,15
'A/D Channel 7     O2 South Plot High Sensor 2     O2(3)               16,17
'A/D Channel 8     O2 North Plot Low Sensor 1      O2(4)               19,20
'A/D Channel 9     O2 North Plot Low Sensor 2      O2(5)               21,22
'A/D Channel 10    O2 North Plot High Sensor 1     O2(6)               23,24
'A/D Channel 11    O2 North Plot High Sensor 2     O2(7)               25,26
'A/D Channel 12    Pressure Tank South Plot        Pres(0)             27,28
'A/D Channel 13    Pressure Tank North Plot        Pres(1)             29,30
'A/D Channel 14    South Chemical Metering Pump                       31,32
'A/D Channel 15    North Chemical Metering Pump                       33,34
'A/D Channel 16    Flow Rate South Plot           FlowR(0)            52,53
'A/D Channel 17    Flow Total South Plot          TFlow(0)            54,55
'A/D Channel 18    Flow Rate North Plot           FlowR(0)            56,57
'A/D Channel 19    Flow Total North Plot          TFlow(0)            58,59
'A/D Channel 20    Pressure Tank South Plot Old                       60,61
'A/D Channel 21    Pressure Tank North Plot Old                       62,63

'D/A Out 0        South Flow Controller          SWaterOut           36,37
'D/A Out 1        South Chem Feed Pump RPM Cont   SChemOut                38,37
```

```
'Board 2 DDA02
'D/A Out 0        North Flow Controller         NWaterPOut           1,2
'D/A Out 1        N Chem Pump RPM Control        NChemPOut             3,4

'   SecondPortA
'Bit 0          North Water pump Valve         WellPumpBit(1)        58
'Bit 1                                              57
'   SecondPortB
'Bit 0          South Chemical pump bit       ChemPumpBit(0)        66
'Bit 1          North Chemical pump bit       ChemPumpBit(1)        65
'   FirstPortA
'Bit 0          N2 Stop Valve Stops N2 from Pump box               82
'Bit 1          N2 South Low                              81
'Bit 2          N2 South High                             80
'Bit 3          N2 North Low                              79
'Bit 4          N2 North High                             78
'   FirstPortB
'Bit 0          South Low Sample/Calibrate valves (2)             90
'Bit 1          South High Sample/Calibrate valves (2)            89
'Bit 2          North Low Sample/Calibrate valves (2)             88
'Bit 3          North HIgh Sample/Calibrate valves (2)            87
```

Private Sub cmdStart_Click()

```
'This is the basic timing program and is based on the timing sequence of the extraction
pumps
Call param

M = 0  'counter used to determine when to write a record to the data file
N = 1 ' counter used to store the number of records written in the file
'ReDim O20(64000)
'ReDim O21(64000)
'ReDim O22(64000)
'ReDim O23(64000)
'ReDim Pres0(64000)
'ReDim Pres1(64000)
'ReDim Temp0(64000)
'ReDim Temp1(64000)
'ReDim DaTime(64000)
Run% = 0

  ' declare revision level of Universal Library

  ULStat% = cbDeclareRevision(5.52!)
  If ULStat% <> 0 Then Stop

  ULStat% = cbErrHandling(PRINTALL, DONTSTOP)
```

```
  If ULStat% <> 0 Then Stop

  ' If cbErrHandling% is set for STOPALL or STOPFATAL during the program
  ' design stage, Visual Basic will be unloaded when an error is encountered.
  ' We suggest trapping errors locally until the program is ready for compiling
  ' to avoid losing unsaved data during program design.  This can be done by
  ' setting cbErrHandling options as above and checking the value of ULStat%
  ' after a call to the library. If it is not equal to 0, an error has occurred.
  ULStat% = cbDConfigBit(Board1, AUXPORT, 0, Direction%) ' configure Bit 0 as
        output
  If ULStat% <> 0 Then Stop
  ULStat% = cbDConfigBit(Board1, AUXPORT, 1, Direction%) ' configure Bit 1 as
        output
  If ULStat% <> 0 Then Stop
  ULStat% = cbDConfigBit(Board1, AUXPORT, 2, Direction%) ' configure Bit 2 as
        output
  If ULStat% <> 0 Then Stop
  ULStat% = cbDConfigBit(Board1, AUXPORT, 3, Direction%) ' configure Bit 3 as
        output
  If ULStat% <> 0 Then Stop
  ULStat% = cbDConfigBit(Board1, AUXPORT, 4, Direction%) ' configure Bit 4 as
        output
  If ULStat% <> 0 Then Stop
  ULStat% = cbDConfigBit(Board1, AUXPORT, 5, Direction%) ' configure Bit 5 as
        output
  If ULStat% <> 0 Then Stop
  ULStat% = cbDConfigBit(Board1, AUXPORT, 6, Direction%) ' configure Bit 6 as
        output
  If ULStat% <> 0 Then Stop
  ULStat% = cbDConfigBit(Board1, AUXPORT, 7, Direction%) ' configure Bit 7 as
        output
  If ULStat% <> 0 Then Stop

  ULStat% = cbDConfigPort(Board2, FIRSTPORTA, Direction%) ' configure FirstPortA
        as output
  If ULStat% <> 0 Then Stop
  ULStat% = cbDConfigPort(Board2, FIRSTPORTB, Direction%) ' configure FirstPortB
        as output
  If ULStat% <> 0 Then Stop
  ULStat% = cbDConfigPort(Board2, SECONDPORTA, Direction%) ' configure
        SecondPortA as output
  If ULStat% <> 0 Then Stop
  ULStat% = cbDConfigPort(Board2, SECONDPORTB, Direction%) ' configure
        SecondPortB as output
  If ULStat% <> 0 Then Stop
'********
```

```
' initialize digital output ports to a digital 1 of off for all but the well pumps
'*********
      DataValue% = 1 + 2
        ULStat% = cbDOut(Board2, SECONDPORTA, DataValue%)
               If ULStat% <> 0 Then Stop

      DataValue% = 1 + 2
        ULStat% = cbDOut(Board2, SECONDPORTB, DataValue%)
               If ULStat% <> 0 Then Stop

      DataValue% = 1 + 2 + 4 + 8 + 16
        ULStat% = cbDOut(Board2, FIRSTPORTA, DataValue%)
               If ULStat% <> 0 Then Stop
      ULStat% = cbDBitOut(Board1%, AUXPORT, 0, 1)
      If ULStat% <> 0 Then Stop
      ULStat% = cbDBitOut(Board1%, AUXPORT, 1, 1)
      If ULStat% <> 0 Then Stop
      ULStat% = cbDBitOut(Board1%, AUXPORT, 2, 1)
      If ULStat% <> 0 Then Stop
      ULStat% = cbDBitOut(Board1%, AUXPORT, 3, 1)
      If ULStat% <> 0 Then Stop
      ULStat% = cbDBitOut(Board1%, AUXPORT, 4, 1)
      If ULStat% <> 0 Then Stop
      ULStat% = cbDBitOut(Board1%, AUXPORT, 5, 1)
      If ULStat% <> 0 Then Stop
      ULStat% = cbDBitOut(Board1%, AUXPORT, 6, 1)
      If ULStat% <> 0 Then Stop
      ULStat% = cbDBitOut(Board1%, AUXPORT, 7, 1)
      If ULStat% <> 0 Then Stop


'**********
' initialize pumping rates
'**********
ULStat% = cbFromEngUnits(Board1, UNI10VOLTS, SChemPOut, DataValue%)
   If ULStat% <> 0 Then Stop
    ULStat% = cbAOut(Board1, 1, UNI10VOLTS, DataValue%) ' South Chem pump
            RPM output
   If ULStat% <> 0 Then Stop
ULStat% = cbFromEngUnits(Board2, UNI10VOLTS, NChemPOut, DataValue%)
   If ULStat% <> 0 Then Stop
    ULStat% = cbAOut(Board2, 1, UNI10VOLTS, DataValue%) ' North Chem pump
            RPM output
   If ULStat% <> 0 Then Stop

  '  MsgBox (SWaterPOut%)
```

```
'******************
' Initialize oxygen set point
'******************
O2SetS.Caption = Format(O2Set(0), "0.0")
O2SetN.Caption = Format(O2Set(1), "0.0")
'******************
' Initialize pressure tank and pump settings
'******************
SouthHiPresSet.Caption = Format(SouthHiPSet, "0.0")
SouthLoPresSet.Caption = Format(SouthLoPSet, "0.0")
NorthHiPresSEt.Caption = Format(NorthHiPSet, "0.0")
NorthLoPresSet.Caption = Format(NorthLoPSet, "0.0")
PressureSec.Caption = Format(PresSec, "0.") ' Number of seconds for presurazation or
        pump seconds
FillSec.Caption = Format(FilSec, "0.")      ' Number of seconds for filliing
'MsgBox ("test")



'**********
' Initialize flow counters flow counters initialized each time a new file is opened
' Flowcount(0) is the South well cluster, Flowcount(1) is the North well cluster
' Flowcount(2) = South Plot Pump, Flowcount(3) = North plot pump
'**********
For I = 0 To 3
   flowcount(I) = 0
Next I
  CmdStart.Enabled = 0
'**********
' Initialize Well pump bits
'**********
FillBit(0) = 0
FillBit(1) = 0
CycleCountS = 0
CycleCountN = 0
'MsgBox ("test")
'**********
' Calibrate O2 Sensors
'**********
'Call O2Cal
'***********
' Create a files for storing data
'**********
  Date = Now
  Time = Now
```

```
  fname = "c:\temp\DATA" + Format$(Date, "mmm dd ") + Format$(Time, "hh mm ss")
        + ".csv"
  Open fname For Output As #1
  Close
  Date = Now
  Time = Now

  fname3 = "c:\temp\DATA" + Format$(Date, "mmm dd ") + Format$(Time, "hh mm
        ss") + "flow.csv"
  Open fname3 For Output As #3
  Close
'  Channel 14 = South Chem Metering Pump Hi Pin 31, Low Pin 32
'  Channel 15 = North Chem Metering Pump Hi Pin 33, Low Pin 34
'  Channel 16 = South Flow Meter Rate Hi Pin 52, Low Pin 53
'  Channel 17 = South Flow Totalizer Hi Pin 54, Low Pin 55
'  Channel 18 = North Flow Meter Rate Hi Pin 56, Low Pin 57
'  Channel 19 = North Flow Totalizer Hi Pin 58, Low Pin 59
         Open fname3 For Append As #3
        Write #3, Format("Date Time"), Format("South Chem Pump"), Format("N Chem
        Pump"), _
           Format("S Flow Rate"), Format("S Flow Total"), _
           Format("N Flow Rate"), Format("N FlowTotal")
        Close

Do While Run% <> -1

If FillSec <= 1 Then
   Fillsec1 = 0
   Else: Fillsec1 = FillSec - 1
End If

'*************
' Obtain sensor data
'*************
Call Analog_In
Call Analog_In4
'***********
' Fan Control
'***********
Call Fan
'*************
' Set up well pump control
'*************
'MsgBox ("Well Pump Cont IN")
Call WellPumpCtrl
'MsgBox ("Well Pump Cont out")
```

```
'*************
' Determine if plots need water and turn on appropriate pumps Note: make sure there is
        water in pressure tank
' before turning on pumps. If Tank pressure >= low set pressure - 1 assume water is in
        tank.
' Delay chemical feed by one pump cycle and maintain water feed by ten pump cycles
        after turning
' off the chemical feed pump.
' The objective is to reduce the carbon source in the feed line to minimize the potential for
        biological growth in the
' distribution lines.
'*************
'ULStat% = cbDBitOut(Board1, AUXPORT, 2, 0) ' South Air Bit
'   If ULStat% <> 0 Then Stop
'   Call VentTime
'ULStat% = cbDBitOut(Board1, AUXPORT, 2, 1) ' South Air Bit
'   If ULStat% <> 0 Then Stop

'MsgBox ("Before South Air")

Call PlotPumpCtrl
'*******
'Write to the digital output channels the 6052 will be used for the South plot
' and the DDA02 will be used for the north plot the 6052 is configured bitwies and
' the DDA02 is configured port wise
'*******
   '**********
   'Board 1 (6052) Output to control plot pumps and solenoid valves South plot bit 0
        water pump
   '  (pin 85), bit 1 solenoid stop valve (pin 86)
   '   bit 2 chemical feed pump (pin 87),  bit 3 three way soelnoid valve for air flush (pin
        88),
   '   and bit 4 the south water well pump bit(0) (pin 89)
   '**********
'     ULStat% = cbDBitOut(Board1%, AUXPORT, 0, PlotPumpBit(0))
'     If ULStat% <> 0 Then Stop
'     ULStat% = cbDBitOut(Board1%, AUXPORT, 1, PlotPumpBit(1))
'     If ULStat% <> 0 Then Stop
'     ULStat% = cbDBitOut(Board1%, AUXPORT, 2, PlotPumpBit(2))
'     If ULStat% <> 0 Then Stop
'     ULStat% = cbDBitOut(Board1%, AUXPORT, 3, PlotPumpBit(3))
'     If ULStat% <> 0 Then Stop

   '**********
```

```
'Port B Board2 Output port to control plot pumps and solenoid valves North plot bit 4
      water pump (pin 32), bit 5 solenoid stop valve (pin 33)
'   bit 6 chemical feed pump (pin 34), and bit 7 three way soelnoid valve for air flush
      (pin 35)
'**********
'       DataValue1% = PlotPumpBit(4) + 2 * PlotPumpBit(5) + 4 * PlotPumpBit(6) + 8 *
          PlotPumpBit(7)
'       ULStat% = cbDOut(Board2%, FIRSTPORTB, DataValue1%)
'             If ULStat% <> 0 Then Stop

'**********
'Bit 4 Board 1 south plot, Port A Unit 2 North plot Output port to control well pumps
      only 1 bits used  (pin 21)
'**********
'          MsgBox (WellPumpBit(0))
  If WellPumpBit(0) = 1 Or WellPumpBit(1) = 1 Then ' Pumps are to run a pressure
      cycle followed by a fill cycle
    DataValue% = WellPumpBit(0) + 2 * WellPumpBit(1)

      ULStat% = cbDOut(Board2, SECONDPORTA, DataValue%)
            If ULStat% <> 0 Then Stop

     '    MsgBox (DataValue1%)
      Call PressureTime
      DataValue% = 0 ' Vent pressure lines
      ULStat% = cbDOut(Board2, SECONDPORTA, DataValue%)
            If ULStat% <> 0 Then Stop
      Call FillTime ' Don't turn switches back on until sufficient time to fill bladder
          'MsgBox (DataValue1%)
'
  Else: Call SleepTime ' This is the time for one pump cycle

  End If

      DoEvents   ' Yield to other processes
      Refresh    ' required to update the screen


'*************
' determine how to store data it is not necessary to plot or record every data point.
' Data is recorded after M counter observations
' it is also desirable to limit the size of a given record.  A new record is started after
' a fixed number (N counter) of events are recorded
'*************
If M > MM Then ' Write date after (MM +2) sampling events
   Date_time = Now
```

```
   Open fname For Append As #1
   Write #1, Format(Date_time), Format(A_Data(0), "#0.00"), Format(A_Data(1),
         "#0.00"), Format(A_Data(2), "#0.00"), Format(A_Data(3), "#0.00"), _
   Format(A_Data(4), "#0.00"), Format(A_Data(5), "#0.00"), Format(A_Data(6),
         "#0.00"), Format(A_Data(7), "#0.00"), _
   Format(A_Data(8), "#0.00"), Format(A_Data(9), "#0.00"), Format(A_Data(10),
         "#0.00"), Format(A_Data(11), "#0.00"), _
   Format(flowcount(0), "#00000"), Format(flowcount(1), "#00000"),
         Format(flowcount(2), "#00000"), _
   Format(flowcount(3), "#00000"), Format(A_Data(20), "#0.000"), Format(A_Data(21),
         "#0.000")
   Close
   M = 0 'reset the counter for recording and ploting after each data point is recorded
'***********
' place the recorded data in arrays for ploting
'***********
   O20(N) = A_Data(4)
   O21(N) = A_Data(5)
   O22(N) = A_Data(6)
   O23(N) = A_Data(7)
   O24(N) = A_Data(8)
   O25(N) = A_Data(9)
   O26(N) = A_Data(10)
   O27(N) = A_Data(11)
   Pres0(N) = A_Data(12)
   Pres1(N) = A_Data(13)
   Temp0(N) = A_Data(0)
   Temp1(N) = A_Data(1)
   Temp2(N) = A_Data(2)
   Temp3(N) = A_Data(3)
   DaTime(N) = Date_time

'*********
' update the plot
'*********

   Call Plot2
   N = N + 1

'*********
' determine if it is necessary to create a new file
'*********

   If N > NN Then 'Create a new file name after (NN) samples are recorded
      N = 1 'reset the counter
      Date = Now
```

```
        Time = Now

        fname = "c:\temp\DATA" + Format$(Date, "mmm dd ") + Format$(Time, "hh mm
            ss") + ".csv"
        Open fname For Output As #1
        Close
'********
' reset the flow counters
'********
        For I = 0 To 3
            flowcount(I) = 0
        Next I
        'Call O2Cal
    End If
' MsgBox DaTime(N)


 Else: M = M + 1
 End If


Loop
End Sub
```

Private Sub Fan()

```
' This subroutine is to operate the fan in the faults.  The objective is to cool the
' vault and reduce the humidity in the vault.  The sump pump will only remove part of
' the water and the vault would probably remain at near saturated conditions causing
' corrosion and other problems with the instrumentation.  The Fans will be programed to
            be
' on when the outside temperature is cooler than the inside temperature as long as the
            inside
' temperature is above 10 C. Need to leave some cussion to avoid freezing conditions.

If A_Data(3) < A_Data(0) And A_Data(0) > 10 Then
    FanBit(0) = 0
    Else: FanBit(0) = 1
End If
If A_Data(3) < A_Data(1) And A_Data(1) > 10 Then
    FanBit(1) = 0
    Else: FanBit(1) = 1
End If
      ULStat% = cbDBitOut(Board1, AUXPORT, 0, FanBit(0)) 'Fan Control
      If ULStat% <> 0 Then Stop
        Start = Timer
      ULStat% = cbDBitOut(Board1, AUXPORT, 1, FanBit(1)) 'Fan Control
```

```
    If ULStat% <> 0 Then Stop
     Start = Timer

End Sub
```

Private Sub O2Cal()

```
Dim AirVolt(7)
Dim N2Volt(7)
'**********
' Calibrate O2 Sensors first time throuth routine. Sensor reported to respons 90% in 60
        sec
'   Obtain Initial sensor data
'**********
'***********
' Create a file for storing Calibration data
'**********
Label2.BackColor = &HFF& ' red
Label2.Caption = "Calibrating"
  Date = Now
  Time = Now

  fname2 = "c:\temp\DATA" + Format$(Date, "mmm dd ") + Format$(Time, "hh mm
        ss") + "cal.csv"
  Open fname2 For Output As #2
  Close
'*******
' First obtain air reading all valves at the same time
'*******
   DataValue% = 1 * 1 + 1 * 2 + 1 * 4 + 1 * 8 + 1 * 16 + 1 * 32 + 1 * 64 + 1 * 128
    ULStat% = cbDOut(Board2, FIRSTPORTA, DataValue%) 'Do not activate N2 Stop
        valve to pump box Bit 0
    If ULStat% <> 0 Then Stop
   DataValue% = 0 * 1 + 0 * 2 + 0 * 4 + 0 * 8 + 1 * 16 + 1 * 32 + 1 * 64 + 1 * 128
    ULStat% = cbDOut(Board2, FIRSTPORTB, DataValue%) 'Activate All
        Sample/calibrate valves
    If ULStat% <> 0 Then Stop
     Start1 = Timer

Do While Timer < (Start1 + CalTime)
     Call Analog_In3
        Date_time = Now
     Open fname2 For Append As #2
     Write #2, Format(Date_time), Format(A_Data(4), "#0.000"), Format(A_Data(5),
        "#0.000"), _
     Format(A_Data(6), "#0.000"), Format(A_Data(7), "#0.000"), _
```

```
        Format(A_Data(8), "#0.000"), Format(A_Data(9), "#0.000"), Format(A_Data(10),
            "#0.000"), _
        Format(A_Data(11), "#0.0000"), Format(" Air")
        Close
    O2Val(0).Caption = Format$(A_Data(4), "0.000")
'   MsgBox (A_Data(4))

    O2Val(1).Caption = Format$(A_Data(5), "0.000")
    O2Val(2).Caption = Format$(A_Data(6), "0.000")
    O2Val(3).Caption = Format$(A_Data(7), "0.000")
    O2Val(4).Caption = Format$(A_Data(8), "0.000")
    O2Val(5).Caption = Format$(A_Data(9), "0.000")
    O2Val(6).Caption = Format$(A_Data(10), "0.000")
    O2Val(7).Caption = Format$(A_Data(11), "0.000")
        DoEvents   ' Yield to other processes
        Refresh    ' required to update the screen
    Call SleepTime

Loop
For I = 0 To 7
AirVolt(I) = A_Data(I + 4)
Next I
'*********
' Supply N2 only to south low stop N2 from pump box
'*********
    DataValue% = 0 * 1 + 0 * 2 + 1 * 4 + 1 * 8 + 1 * 16 + 1 * 32 + 1 * 64 + 1 * 128
      ULStat% = cbDOut(Board2, FIRSTPORTA, DataValue%) 'Activate N2 stop valve
            and Air/N2 valve for N2 calibration
      If ULStat% <> 0 Then Stop
    DataValue% = 0 * 1 + 1 * 2 + 1 * 4 + 1 * 8 + 1 * 16 + 1 * 32 + 1 * 64 + 1 * 128
      ULStat% = cbDOut(Board2, FIRSTPORTB, DataValue%) 'Activate
            Sample/Calibrate valves for South Low
      If ULStat% <> 0 Then Stop
'MsgBox ("South Low ")
        Start1 = Timer
Label2.BackColor = &HFF00& ' green
Refresh
Do While Timer < (Start1 + CalTime)
        Call Analog_In3
'           A_Data(4) = 0 'comment line out when not testing
'           A_Data(5) = 0 'comment line out when not testing
            Date_time = Now
        Open fname2 For Append As #2
        Write #2, Format(Date_time), Format(A_Data(4), "#0.000"), Format(A_Data(5),
            "#0.000"), _
        Format(""), Format(""), _
```

```
        Format(""), Format(""), Format(""), _
        Format(""), Format("N2")
        Close
    O2Val(0).Caption = Format$(A_Data(4), "0.000")
    O2Val(1).Caption = Format$(A_Data(5), "0.000")
        DoEvents   ' Yield to other processes
        Refresh    ' required to update the screen
    Call SleepTime
Loop
For I = 0 To 1
    N2Volt(I) = A_Data(I + 4)
Next I
'Begin South High supplying N2
    DataValue% = 0 * 1 + 1 * 2 + 0 * 4 + 1 * 8 + 1 * 16 + 1 * 32 + 1 * 64 + 1 * 128
      ULStat% = cbDOut(Board2, FIRSTPORTA, DataValue%) 'Activate N2/Air Valve
            for N2 stop valve for
            'South High channels 6 and 7
      If ULStat% <> 0 Then Stop
    DataValue% = 1 * 1 + 0 * 2 + 1 * 4 + 1 * 8 + 1 * 16 + 1 * 32 + 1 * 64 + 1 * 128
      ULStat% = cbDOut(Board2, FIRSTPORTB, DataValue%) 'Activate
            Sample/Calibrate for
            'South High channels 6 and 7
      If ULStat% <> 0 Then Stop
'MsgBox ("South High 1")
        Start1 = Timer

Do While Timer < (Start1 + CalTime)
        Call Analog_In3
'           A_Data(6) = 0 'comment line out when not testing
'           A_Data(7) = 0 'comment line out when not testing
            Date_time = Now
        Open fname2 For Append As #2
        Write #2, Format(Date_time), Format(""), Format(""), _
        Format(A_Data(6), "#0.000"), Format(A_Data(7), "#0.000"), _
        Format(""), Format(""), Format(""), _
        Format(""), Format("N2")
        Close
    O2Val(2).Caption = Format$(A_Data(6), "0.00")
    O2Val(3).Caption = Format$(A_Data(7), "0.00")
        DoEvents   ' Yield to other processes
        Refresh    ' required to update the screen
    Call SleepTime
Loop
For I = 0 To 1
    N2Volt(I + 2) = A_Data(I + 6)
Next I
```

```
'Begin North Low N2 Calibration
    DataValue% = 0 * 1 + 1 * 2 + 1 * 4 + 0 * 8 + 1 * 16 + 1 * 32 + 1 * 64 + 1 * 128
      ULStat% = cbDOut(Board2, FIRSTPORTA, DataValue%) 'Activate N2/Air Valve
            and N2 stop valve for
            'North Low Channels 8 and 9
      If ULStat% <> 0 Then Stop
    DataValue% = 1 * 1 + 1 * 2 + 0 * 4 + 1 * 8 + 1 * 16 + 1 * 32 + 1 * 64 + 1 * 128
      ULStat% = cbDOut(Board2, FIRSTPORTB, DataValue%) 'Activate
            Sample/Calibrate valves
            'North Low Channels 8 and 9
      If ULStat% <> 0 Then Stop
        Start1 = Timer

Do While Timer < (Start1 + CalTime)
        Call Analog_In3
'            A_Data(8) = 0 'comment line out when not testing
'            A_Data(9) = 0 'comment line out when not testing
            Date_time = Now
        Open fname2 For Append As #2
        Write #2, Format(Date_time), Format(""), Format(""), _
        Format(""), Format(""), _
        Format(A_Data(8), "#0.000"), Format(A_Data(9), "#0.000"), Format(""), _
        Format(""), Format("N2")
        Close
    O2Val(4).Caption = Format$(A_Data(8), "0.00")
    O2Val(5).Caption = Format$(A_Data(9), "0.00")
        DoEvents   ' Yield to other processes
        Refresh    ' required to update the screen
    Call SleepTime
Loop
For I = 0 To 1
    N2Volt(I + 4) = A_Data(I + 8)
Next I
' Begin North High N2 Calibration
    DataValue% = 0 * 1 + 1 * 2 + 1 * 4 + 1 * 8 + 0 * 16 + 1 * 32 + 1 * 64 + 1 * 128
      ULStat% = cbDOut(Board2, FIRSTPORTA, DataValue%) 'Activate N2/Air Valve
            and N2 stop valve
            'North High Channels 10 and 11
      If ULStat% <> 0 Then Stop
    DataValue% = 1 * 1 + 1 * 2 + 1 * 4 + 0 * 8 + 1 * 16 + 1 * 32 + 1 * 64 + 1 * 128
      ULStat% = cbDOut(Board2, FIRSTPORTB, DataValue%) 'Activate
            Sample/Calibrate valves for
            'North High Channels 10 and 11
      If ULStat% <> 0 Then Stop
        Start1 = Timer
```

```
Do While Timer < (Start1 + CalTime)
      Call Analog_In3
'          A_Data(10) = 0 'comment line out when not testing
'          A_Data(11) = 0 'comment line out when not testing
         Date_time = Now
      Open fname2 For Append As #2
      Write #2, Format(Date_time), Format(""), Format(""), _
      Format(""), Format(""), _
      Format(""), Format(""), Format(A_Data(10), "#0.000"), _
      Format(A_Data(11), "#0.000"), Format("N2")
      Close
   O2Val(6).Caption = Format$(A_Data(10), "0.00")
   O2Val(7).Caption = Format$(A_Data(11), "0.00")
      DoEvents   ' Yield to other processes
      Refresh    ' required to update the screen
   Call SleepTime
Loop
For I = 0 To 1
   N2Volt(I + 6) = A_Data(I + 10)
Next I
For I = 0 To 7

Slope(I + 4) = 20.9 / (AirVolt(I) - N2Volt(I))
Intercept(I + 4) = (-1) * Slope(I + 4) * N2Volt(I)

Next I
      Open fname2 For Append As #2
      Write #2, Format("Slopes"), Format(Slope(4), "#0.000"), Format(Slope(5),
         "#0.000"), _
      Format(Slope(6), "#0.000"), _
      Format(Slope(7), "#0.000"), Format(Slope(8), "#0.000"), Format(Slope(9),
         "#0.000"), _
      Format(Slope(10), "#0.000"), Format(Slope(11), "#0.000")
      Close
      Open fname2 For Append As #2
      Write #2, Format("Intercepts"), Format(Intercept(4), "#0.000"), Format(Intercept(5),
         "#0.000"), _
      Format(Intercept(6), "#0.000"), _
      Format(Intercept(7), "#0.000"), Format(Intercept(8), "#0.000"), Format(Intercept(9),
         "#0.000"), _
      Format(Intercept(10), "#0.000"), Format(Intercept(11), "#0.000")
      Close

   DataValue% = 1 * 1 + 1 * 2 + 1 * 4 + 1 * 8 + 1 * 16 + 1 * 32 + 1 * 64 + 1 * 128
    ULStat% = cbDOut(Board2, FIRSTPORTA, DataValue%) 'Denergize N2/Air Valve
    If ULStat% <> 0 Then Stop
```

```
Label2.BackColor = &HFFFFFF
Label2.Caption = ""
       DoEvents   ' Yield to other processes
       Refresh    ' required to update the screen

End Sub
```

Private Sub PlotPumpCtrl()

```
Dim AVGO2
'********
' fluids are pumped only part of the time.  This allows draining the lines in an attempt to
' minimize the biological growth in the pipeing.  pulses of flow are delivered to the plot
    as
' one part out of the variable cyclecount cyclecount must be 3 or greater
'********

  If A_Data(12) > SouthLoPSet - 1 Then ' Sufficient water pressure to operate pump
    'AVGO2 = (A_Data(4) + A_Data(5)) / 2
    AVGO2 = 20 'for testing purposes
    If AVGO2 > O2Set(0) Then ' Sufficient oxygen to apply water
      Cnt1.Caption = Format$((cyclecount - CycleCountS) * (FilSec + PresSec), "0")
      CycleCountS = CycleCountS + 1 ' used to determine when to pump water vs flush
      with air
      'MsgBox (CycleCountS)
      If CycleCountS = 1 Then 'Turn on plot pump chemical pump and open solenoid
      valve
        flowcount(2) = flowcount(2) + 1 ' Increment South water flow counter
        SFlowR = SWaterPOut ' south plot water flow controller activated
        PlotStopBit(0) = 0 ' solenoid stop valve activated (opened) normally closed
      valve
        ChemPumpBit(0) = 0 ' south chemical pump bit turned on pin 34  Note a zero
      is
                      ' on for this pump
        PlotAirBit(0) = 1 ' south vent valve is off
        lblPump(0).BackColor = &HFF00& 'green
        lblPump(1).BackColor = &HFF00& 'green

      ElseIf CycleCountS = 2 Then ' Flush line with Air everything off but air
        SFlowR = 0 ' stop flow through flow controller
        PlotStopBit(0) = 1 ' closes the solenoid stop valve (normally closed valve)
        ChemPumpBit(0) = 1 ' a 1 turning off south chemical pump bit turned off
        'PlotAirBit(0) = 0 'south vent valve is on
        lblPump(0).BackColor = &HFFFF& 'yellow
        lblPump(1).BackColor = &HFFFF& 'Yellow

      ElseIf CycleCountS > 2 And CycleCountS < cyclecount Then ' everything off
```

```
            SFlowR = 0 ' no flow thrugh flow controller
            PlotStopBit(0) = 1 ' close solenoid stop valve (normally closed valve)
            ChemPumpBit(0) = 1 ' south chemical pump bit turned off
            PlotAirBit(0) = 1 'south vent valve off (normally closed valve)
            lblPump(0).BackColor = &HFF& 'red
            lblPump(1).BackColor = &HFF& 'red

         ElseIf CycleCountS >= cyclecount Then 'last cycle before a pulse everything off
            CycleCountS = 0
            SFlowR = 0 ' no flow thrugh flow controller
            PlotStopBit(0) = 1 ' close solenoid stop valve (normally closed valve)
            ChemPumpBit(0) = 1 ' south chemical pump bit turned off
            PlotAirBit(0) = 1 'south vent valve off (normally closed valve)
            lblPump(0).BackColor = &HFF& 'red
            lblPump(1).BackColor = &HFF& 'red
         Else:     ans = MsgBox("Continue?", vbYesNo + vbQuestion, "Problem in Plot
         Pump Control Logic ")

         End If
      Else:   'insuficient oxygen to deliver water
            Cnt1.Caption = Format$("O2 ??")
            SFlowR = 0 ' no flow thrugh flow controller
            PlotStopBit(0) = 1 ' close solenoid stop valve (normally closed valve)
            ChemPumpBit(0) = 1 ' south chemical pump bit turned off
            PlotAirBit(0) = 1 'south vent valve off (normally closed valve)
            lblPump(0).BackColor = &HFF& 'red
            lblPump(1).BackColor = &HFF& 'red
      End If
   Else:   'Insuficient water in pressure tank to run a cycle
         Cnt1.Caption = Format$("H2O ??")
            SFlowR = 0 ' no flow thrugh flow controller
            PlotStopBit(0) = 1 ' close solenoid stop valve (normally closed valve)
            ChemPumpBit(0) = 1 ' south chemical pump bit turned off
            PlotAirBit(0) = 1 'south vent valve off (normally closed valve)
         lblPump(0).BackColor = &H8000000F   'grey
         lblPump(1).BackColor = &H8000000F   'grey
   End If
'*****************************************************
   If A_Data(13) > NorthLoPSet - 1 Then ' Sufficient water pressure to operate pump
      AVGO2 = (A_Data(8) + A_Data(9)) / 2
      AVGO2 = 20 'for testing purposes
      If AVGO2 > O2Set(1) Then ' Sufficient oxygen to apply water
         Cnt2.Caption = Format$((cyclecount - CycleCountN) * (FilSec + PresSec), "0")
         CycleCountN = CycleCountN + 1
         If CycleCountN = 1 Then
            flowcount(3) = flowcount(3) + 1 ' Increment South water flow counter
```

```
        NFlowR = NWaterPOut ' north plot water pump bit turned on pin 32 Note a
    zero is on for this pump
        PlotStopBit(1) = 0 ' solenoid stop valve activated (opened) normally closed
        ChemPumpBit(1) = 0 ' north chemical pump bit turned on
        PlotAirBit(1) = 1 ' north vent valve not activated
        lblPump(2).BackColor = &HFF00& 'green
        lblPump(3).BackColor = &HFF00& 'green
    ElseIf CycleCountN = 2 Then
        NFlowR = 0 ' a 1 is turning off the South plot water
        PlotStopBit(1) = 1 ' closes the solenoid stop valve
        ChemPumpBit(1) = 1 ' north chemical pump bit turned off
        'PlotAirBit(1) = 0 'north compressed air vent valve
        lblPump(2).BackColor = &HFFFF& 'Yellow
        lblPump(3).BackColor = &HFFFF& 'Yellow
    ElseIf CycleCountN > 2 And CycleCountN < cyclecount Then
        NFlowR = 0 ' a 1 is turning off the flow controller
        PlotStopBit(1) = 1 ' solenoid stop valve
        ChemPumpBit(1) = 1 ' north chemical pump bit turned off
        PlotAirBit(1) = 1 'south compressed air vent valve
        lblPump(2).BackColor = &HFF& 'red
        lblPump(3).BackColor = &HFF& 'red
    ElseIf CycleCountN >= cyclecount Then 'last cycle before a pulse
        CycleCountN = 0
        NFlowR = 0 ' a 1 is turning off the flow controller
        PlotStopBit(1) = 1 ' solenoid stop valve
        ChemPumpBit(1) = 1 ' north chemical pump bit turned off
        PlotAirBit(1) = 1 'south compressed air vent valve
        lblPump(2).BackColor = &HFF& 'red
        lblPump(3).BackColor = &HFF& 'red
    Else:     ans = MsgBox("Continue?", vbYesNo + vbQuestion, "Problem in Plot
    Pump Control Logic ")
    End If
  Else:   'insuficient oxygen to deliver water
        Cnt2.Caption = Format$("O2 ??")
        NFlowR = 0 ' a 1 is turning off the flow controller
        PlotStopBit(1) = 1 ' solenoid stop valve
        ChemPumpBit(1) = 1 ' north chemical pump bit turned off
        PlotAirBit(1) = 1 'south compressed air vent valve
        lblPump(2).BackColor = &HFF& 'red
        lblPump(3).BackColor = &HFF& 'red
  End If
Else:   'Insuficient water in pressure tank to run a cycle
    Cnt2.Caption = Format$("H2O ??")
        NFlowR = 0 ' a 1 is turning off the flow controller
        PlotStopBit(1) = 1 ' solenoid stop valve
        ChemPumpBit(1) = 1 ' north chemical pump bit turned off
```

```
            PlotAirBit(1) = 1 'south compressed air vent valve
          lblPump(2).BackColor = &H8000000F   'grey
          lblPump(3).BackColor = &H8000000F   'grey
    End If
'**************************************************
'MsgBox ("test")
'*********
' first set the analog outputs then the Digital outputs
'*********
'ULStat% = cbFromEngUnits(Board2, UNI10VOLTS, NFlowR, DataValue%)
'    If ULStat% <> 0 Then Stop
'    ULStat% = cbAOut(Board2, 0, UNI10VOLTS, DataValue%)
'    If ULStat <> 0 Then Stop
    'MsgBox (NFlowR)

'ULStat% = cbFromEngUnits(Board1, UNI10VOLTS, SFlowR, DataValue%)
'    If ULStat% <> 0 Then Stop
'    ULStat% = cbAOut(Board1, 0, UNI10VOLTS, DataValue%)
'    If ULStat <> 0 Then Stop

DataValue% = ChemPumpBit(0) + 2 * ChemPumpBit(1)
ULStat% = cbDOut(Board2, SECONDPORTB, DataValue%) ' chemical flow bits
    If ULStat% <> 0 Then Stop
ULStat% = cbDBitOut(Board1, AUXPORT, 4, PlotStopBit(0)) 'south stop valve
    If ULStat% <> 0 Then Stop
ULStat% = cbDBitOut(Board1, AUXPORT, 5, PlotStopBit(1)) 'north stop valve
    If ULStat% <> 0 Then Stop
ULStat% = cbDBitOut(Board1, AUXPORT, 2, PlotAirBit(0)) 'south air valve
    If ULStat% <> 0 Then Stop
ULStat% = cbDBitOut(Board1, AUXPORT, 3, PlotAirBit(1)) 'north air valve
    If ULStat% <> 0 Then Stop
'*******
' measure flows
'*******
    If PlotStopBit(0) = 0 Or PlotStopBit(1) = 0 Then
          delay = 10
          Call DTime(delay)
          Call Analog_In2
          Date_time = Now
           Open fname3 For Append As #3
         Write #3, Format(Date_time), Format(A_Data(14), "#0.0000"),
          Format(A_Data(15), "#0.000"), _
             Format(A_Data(16), "#0.000"), Format(A_Data(17), "#0.000"),
          Format(A_Data(18), "#0.000"), _
             Format(A_Data(19), "#0.000")
          Close
```

```
          Call DTime(delay)
          Call Analog_In2
          Date_time = Now
           Open fname3 For Append As #3
         Write #3, Format(Date_time), Format(A_Data(14), "#0.0000"),
          Format(A_Data(15), "#0.000"), _
              Format(A_Data(16), "#0.000"), Format(A_Data(17), "#0.000"),
          Format(A_Data(18), "#0.000"), _
              Format(A_Data(19), "#0.000")
          Close
          Call DTime(delay)
          Call Analog_In2
          Date_time = Now
           Open fname3 For Append As #3
         Write #3, Format(Date_time), Format(A_Data(14), "#0.0000"),
          Format(A_Data(15), "#0.000"), _
              Format(A_Data(16), "#0.000"), Format(A_Data(17), "#0.000"),
          Format(A_Data(18), "#0.000"), _
              Format(A_Data(19), "#0.000")
          Close
          delay = FilSec + PresSec - delay * 3
          Call DTime(delay)
     End If
If CycleCountN = 2 And CycleCountS = 2 Then
    ULStat% = cbDBitOut(Board1, AUXPORT, 2, 0)
       If ULStat% <> 0 Then Stop
    ULStat% = cbDBitOut(Board1, AUXPORT, 3, 0)
       If ULStat% <> 0 Then Stop
    Call VentTime
    ULStat% = cbDBitOut(Board1, AUXPORT, 2, 1)
       If ULStat% <> 0 Then Stop
    ULStat% = cbDBitOut(Board1, AUXPORT, 3, 1)
       If ULStat% <> 0 Then Stop
    Call VentTime
    ULStat% = cbDBitOut(Board1, AUXPORT, 2, 0)
       If ULStat% <> 0 Then Stop
    ULStat% = cbDBitOut(Board1, AUXPORT, 3, 0)
       If ULStat% <> 0 Then Stop
    Call VentTime
    ULStat% = cbDBitOut(Board1, AUXPORT, 2, 1)
       If ULStat% <> 0 Then Stop
    ULStat% = cbDBitOut(Board1, AUXPORT, 3, 1)
       If ULStat% <> 0 Then Stop
          Call Analog_In2
          Date_time = Now
           Open fname3 For Append As #3
```

```
            Write #3, Format(Date_time), Format(A_Data(14), "#0.0000"),
             Format(A_Data(15), "#0.000"), _
                Format(A_Data(16), "#0.000"), Format(A_Data(17), "#0.000"),
             Format(A_Data(18), "#0.000"), _
                Format(A_Data(19), "#0.000")
             Close
ElseIf CycleCountN = 2 And CycleCountS <> 2 Then
       ULStat% = cbDBitOut(Board1, AUXPORT, 2, 0)
          If ULStat% <> 0 Then Stop
       Call VentTime
       ULStat% = cbDBitOut(Board1, AUXPORT, 2, 1)
          If ULStat% <> 0 Then Stop
       Call VentTime
       ULStat% = cbDBitOut(Board1, AUXPORT, 2, 0)
       If ULStat% <> 0 Then Stop
       Call VentTime
       ULStat% = cbDBitOut(Board1, AUXPORT, 2, 1)
       If ULStat% <> 0 Then Stop
          Call Analog_In2
          Date_time = Now
           Open fname3 For Append As #3
            Write #3, Format(Date_time), Format(A_Data(14), "#0.0000"),
             Format(A_Data(15), "#0.000"), _
                Format(A_Data(16), "#0.000"), Format(A_Data(17), "#0.000"),
             Format(A_Data(18), "#0.000"), _
                Format(A_Data(19), "#0.000")
             Close
ElseIf CycleCountS = 2 And CycleCountN <> 2 Then
    ULStat% = cbDBitOut(Board1, AUXPORT, 3, 0)
       If ULStat% <> 0 Then Stop
    Call VentTime
    ULStat% = cbDBitOut(Board1, AUXPORT, 3, 1)
       If ULStat% <> 0 Then Stop
    Call VentTime
    ULStat% = cbDBitOut(Board1, AUXPORT, 3, 0)
       If ULStat% <> 0 Then Stop
    Call VentTime
    ULStat% = cbDBitOut(Board1, AUXPORT, 3, 1)
       If ULStat% <> 0 Then Stop
          Call Analog_In2
          Date_time = Now
           Open fname3 For Append As #3
            Write #3, Format(Date_time), Format(A_Data(14), "#0.0000"),
             Format(A_Data(15), "#0.000"), _
                Format(A_Data(16), "#0.000"), Format(A_Data(17), "#0.000"),
             Format(A_Data(18), "#0.000"), _
```

```
            Format(A_Data(19), "#0.000")
          Close
Else:

End If
'   If WellPumpBit(0) = 0 Or WellPumpBit(1) = 0 Then ' Pumps are to run a pressure
          cycle followed by a fill cycle
'      DataValue% = WellPumpBit(0) + 2 * WellPumpBit(1)

'      ULStat% = cbDOut(Board2, SECONDPORTA, DataValue%)
'             If ULStat% <> 0 Then Stop

          'MsgBox (DataValue1%)
'       Call PressureTime
'       If CycleCoutnS = 1 Or CycleCountN = 1 Then

'           DataValue% = 3 ' Vent pressure lines
'           ULStat% = cbDOut(Board2, SECONDPORTA, DataValue%)
'             If ULStat% <> 0 Then Stop
'           Call FillTime ' Don't turn switches back on until sufficient time to fill bladder
          'MsgBox (DataValue1%)
'       End If
'   Else: Call SleepTime ' This is the time for one pump cycle

'   End If
DataValue% = 3
ULStat% = cbDOut(Board2, SECONDPORTB, DataValue%) ' chemical flow bits
   If ULStat% <> 0 Then Stop
ULStat% = cbDBitOut(Board1, AUXPORT, 4, 1) 'south stop valve
   If ULStat% <> 0 Then Stop
ULStat% = cbDBitOut(Board1, AUXPORT, 5, 1) 'north stop valve
   If ULStat% <> 0 Then Stop
ULStat% = cbDBitOut(Board1, AUXPORT, 2, 1) 'south air valve
   If ULStat% <> 0 Then Stop
ULStat% = cbDBitOut(Board1, AUXPORT, 3, 1) 'north air valve
   If ULStat% <> 0 Then Stop
      Call Analog_In

      DoEvents   ' Yield to other processes
      Refresh    ' required to update the screen
End Sub
```

Private Sub WellPumpCtrl()

```
'****************
' Determine if well pumps should cycle.  Since the well pumps cycle faster than the plot
          pumps,
```

```
' the well pump cycle will control the sampling frequency
'****************
'
'
'MsgBox (WellPumpBit(0))



    If A_Data(12) <= SouthLoPSet Then
       WellPumpBit(0) = 1
       FillBit(0) = 0
       lblPump(4).BackColor = &HFFFF&          'Yellow
       flowcount(0) = flowcount(0) + 1
       'MsgBox ("If 1")
    ElseIf A_Data(12) > SouthHiPSet And FillBit(0) = 1 Then  '
       WellPumpBit(0) = 0
       lblPump(4).BackColor = &HFF& 'Red
       'MsgBox ("If loop 2")
    ElseIf A_Data(12) > SouthLoPSet And FillBit(0) = 0 And A_Data(12) <
       SouthHiPSet Then '
       WellPumpBit(0) = 1
       lblPump(4).BackColor = &HFF00& 'green
       flowcount(0) = flowcount(0) + 1
       'MsgBox ("If loop 3")
    ElseIf A_Data(12) >= SouthHiPSet Then
       WellPumpBit(0) = 0
       FillBit(0) = 1
       lblPump(4).BackColor = &HFF& 'red
       'MsgBox ("If loop 4")
    Else:    ans = MsgBox("Continue?", vbYesNo + vbQuestion, "Problem in Pump
       Control Logic ")


    End If

    If A_Data(13) <= NorthLoPSet Then
       WellPumpBit(1) = 1
       FillBit(1) = 0
       lblPump(5).BackColor = &HFFFF&     'Yellow
       flowcount(1) = flowcount(1) + 1
       'MsgBox ("If 1")
    ElseIf A_Data(13) > NorthLoPSet And FillBit(1) = 1 Then
       WellPumpBit(1) = 0
       lblPump(5).BackColor = &HFF& 'Red
       'MsgBox ("If loop 2")
```

```
    ElseIf A_Data(13) > NorthLoPSet And FillBit(1) = 0 And A_Data(13) <
      NorthHiPSet Then
      WellPumpBit(1) = 1
      lblPump(5).BackColor = &HFF00& 'green
      flowcount(1) = flowcount(1) + 1
      'MsgBox ("If loop 3")
    ElseIf A_Data(13) >= NorthHiPSet Then
      WellPumpBit(1) = 0
      FillBit(1) = 1
      lblPump(5).BackColor = &HFF& 'red
      'MsgBox ("If loop 4")
    Else:     ans = MsgBox("Continue?", vbYesNo + vbQuestion, "Problem in Pump
      Control Logic ")

    End If

'   MsgBox ("test")



End Sub
```

Private Sub CmdStop_Click()

```
'**********
' set output lines to off when system is stoped
'**********

   DataValue% = 3
    ULStat% = cbDOut(Board2, SECONDPORTB, DataValue%)
         If ULStat% <> 0 Then Stop

   DataValue% = 1 + 2 + 4 + 8 + 16
    ULStat% = cbDOut(Board2, FIRSTPORTA, DataValue%)
         If ULStat% <> 0 Then Stop
   ULStat% = cbDBitOut(Board1%, AUXPORT, 0, 1)
   If ULStat% <> 0 Then Stop
   ULStat% = cbDBitOut(Board1%, AUXPORT, 1, 1)
   If ULStat% <> 0 Then Stop
   ULStat% = cbDBitOut(Board1%, AUXPORT, 2, 1)
   If ULStat% <> 0 Then Stop
   ULStat% = cbDBitOut(Board1%, AUXPORT, 3, 1)
   If ULStat% <> 0 Then Stop
   ULStat% = cbDBitOut(Board1%, AUXPORT, 4, 1)
   If ULStat% <> 0 Then Stop
   ULStat% = cbDBitOut(Board1%, AUXPORT, 5, 1)
   If ULStat% <> 0 Then Stop
```

```
    ULStat% = cbDBitOut(Board1%, AUXPORT, 6, 1)
    If ULStat% <> 0 Then Stop
    ULStat% = cbDBitOut(Board1%, AUXPORT, 7, 1)
    If ULStat% <> 0 Then Stop
'********
' initialize states of the output ports for Board 2 DDA02
'********
     DataValue1% = 1 + 2 * 1 + 4 * 1 + 8 * 1 + 16 * 1 + 32 * 1
'      ULStat% = cbDOut(Board2%, FIRSTPORTB, DataValue1%)
'          If ULStat% <> 0 Then Stop
     ULStat% = cbDOut(Board2%, FIRSTPORTA, DataValue1%)
         If ULStat% <> 0 Then Stop
     ULStat% = cbDOut(Board2%, SECONDPORTA, 0)
         If ULStat% <> 0 Then Stop

Run% = -1
Close 1
Close 2
Close 3

Close
  CmdStart.Enabled = True


End Sub
```

Private Sub Analog_In() ' used for channel 0 thru 13

```
'**********
' Analog Channel inputs PCI-DAS6031 Board 1
'  Channel 0 = South Vault Temperature Sensor Hi Pin 2, Low Pin 3
'  Channel 1 = North Vault Temperature Sensor Hi Pin 4, Low Pin 5
'  Channel 2 = Shed Temperature Sensor Hi Pin 6, Low Pin 7
'  Channel 3 = Outside Air Under Shed Temperature Sensor Hi Pin 8, Low Pin 9
'  Channel 4 = South Plot Low O2 Sensor 1 Hi Pin 10, Low Pin 11
'  Channel 5 = South Plot Low O2 Sensor 2 Hi Pin 12, Low Pin 13
'  Channel 6 = South Plot Hi O2 Sensor 1 Hi Pin 14, Low Pin 15
'  Channel 7 = South Plot Hi O2 Sensor 2 Hi Pin 16, Low Pin 17
'  Channel 8 = North Plot Low O2 Sensor 1 Hi Pin 19, Low Pin 20
'  Channel 9 = North Plot Low O2 Sensor 2 Hi Pin 21, Low Pin 22
'  Channel 10 = North Plot Hi O2 Sensor 1 Hi Pin 23, Low Pin 24
'  Channel 11 = North Plot Hi O2 Sensor 2 Hi Pin 25, Low Pin 26
'  Channel 12 = South Pressure Transducer Hi pin 27, Low Pin 28
'  Channel 13 = North Pressure Transducer Hi Pin 29, Low Pin 30
'  Channel 14 = South Chem Metering Pump Hi Pin 31, Low Pin 32
'  Channel 15 = North Chem Metering Pump Hi Pin 33, Low Pin 34
'  Channel 16 = South Flow Meter Rate Hi Pin 52, Low Pin 53
```

```
'   Channel 17 = South Flow Totalizer Hi Pin 54, Low Pin 55
'   Channel 18 = North Flow Meter Rate Hi Pin 56, Low Pin 57
'   Channel 19 = North Flow Totalizer Hi Pin 58, Low Pin 59
'   Channel 20 = South Old Pressure transducer Hi pin 60, Low Pin 61
'   Channel 21 = North Old Pressure transducer Hi Pin 62, Low Pin 63
'********
'ReDim SumAdata(13)
'ReDim Data(NumPoints)
For I = 0 To 13
   MemHandle = cbWinBufAlloc(NumPoints)
   If MemHandle = 0 Then Stop

'LowChannel = i
'HighChannel = i
'CBCount& = NumPoints% ' Total number of data points to collect
'CBRate& = 90000          'Sampling rate in HZ per channel
      Gain = Voltrange(I)
'Options = CONVERTDATA
   ULStat% = cbAInScan(Board1, I, I, NumPoints, CBRate&, Gain, MemHandle,
         Options)
   If ULStat% = 91 Then
      ULStat% = cbErrHandling(DONTPRINT, DONTSTOP)
   ElseIf ULStat% <> 0 Then
      Stop
   End If
' transfer values from Windows buffer to data array used by VBA
   ULStat% = cbWinBufToArray(MemHandle, ADData(0), FirstPoint, NumPoints)
   ULStat% = cbWinBufFree(MemHandle)      ' Free up memory for use by
   If ULStat% <> 0 Then Stop               ' other programs

   If ULStat% <> 0 Then Stop
   SumAdata(I) = 0
   For J = 0 To NumPoints
      ULStat = cbToEngUnits(Board1, Gain, ADData%(J), Data(J))
      SumAdata(I) = SumAdata(I) + Data(J)
   Next J
   A_Data(I) = SumAdata(I) / NumPoints
'MsgBox (A_Data(i%))

Next I
'********
' Convert voltages to real world values
'********
For I = 0 To 13
   A_Data(I) = (A_Data(I) - Intercept(I)) * Slope(I)
Next I
```

```
'*********
' Create Display captions
'*********

'A_Data(4) = 9
'A_Data(5) = 9

For J = 0 To 3
   TempVal(J).Caption = Format$(A_Data(J), "0.00")
Next J
For J = 0 To 7
   O2Val(J).Caption = Format$(A_Data(J + 4), "0.00")
Next J
For J = 0 To 1
   TankPres(J).Caption = Format$(A_Data(J + 12), "0.00")
Next J

Refresh            ' required to update the screen

End Sub
```

Private Sub Analog_In2() ' used for channel 14 thru 19

```
'**********
' Analog Channel inputs PCI-DAS6031 Board 1
'   Channel 14 = South Chem Metering Pump Hi Pin 31, Low Pin 32
'   Channel 15 = North Chem Metering Pump Hi Pin 33, Low Pin 34
'   Channel 16 = South Flow Meter Rate Hi Pin 52, Low Pin 53
'   Channel 17 = South Flow Totalizer Hi Pin 54, Low Pin 55
'   Channel 18 = North Flow Meter Rate Hi Pin 56, Low Pin 57
'   Channel 19 = North Flow Totalizer Hi Pin 58, Low Pin 59
'********
'ReDim SumAdata(13)
'ReDim Data(NumPoints)
For I = 14 To 19
   MemHandle = cbWinBufAlloc(NumPoints)
   If MemHandle = 0 Then Stop

'LowChannel = i
'HighChannel = i
'CBCount& = NumPoints% ' Total number of data points to collect
'CBRate& = 90000          'Sampling rate in HZ per channel
      Gain = Voltrange(I)
'Options = CONVERTDATA
   ULStat% = cbAInScan(Board1, I, I, NumPoints, CBRate&, Gain, MemHandle,
         Options)
   If ULStat% = 91 Then
```

```
      ULStat% = cbErrHandling(DONTPRINT, DONTSTOP)
   ElseIf ULStat% <> 0 Then
      Stop
   End If
' transfer values from Windows buffer to data array used by VBA
   ULStat% = cbWinBufToArray(MemHandle, ADData(0), FirstPoint, NumPoints)
   ULStat% = cbWinBufFree(MemHandle)      ' Free up memory for use by
   If ULStat% <> 0 Then Stop              ' other programs

   If ULStat% <> 0 Then Stop
   SumAdata(I) = 0
   For J = 0 To NumPoints
      ULStat = cbToEngUnits(Board1, Gain, ADData%(J), Data(J))
      SumAdata(I) = SumAdata(I) + Data(J)
   Next J
   A_Data(I) = SumAdata(I) / NumPoints
'MsgBox (A_Data(i%))

Next I


Refresh            ' required to update the screen

End Sub
```

Private Sub SleepTime() 'Time between measurements when water is not needed

```
         Start = Timer
            Do While Timer < Start + FilSec + PresSec
               If Timer < Start Then Start = Start - 86400 'Timer crossed midnight
               DoEvents   ' Yield to other processes
            Loop

End Sub
```

Private Sub DTime(delay) 'Delay time

```
         Start = Timer
            Do While Timer < Start + delay
               If Timer < Start Then Start = Start - 86400 'Timer crossed midnight
               DoEvents   ' Yield to other processes
            Loop

End Sub
```

Private Sub Analog_In3() ' used for channel 4 thru 11

```
'**********
' Analog Channel inputs PCI-DAS6031 Board 1
'   Channel 4 = South Plot Low O2 Sensor 1 Hi Pin 10, Low Pin 11
'   Channel 5 = South Plot Low O2 Sensor 2 Hi Pin 12, Low Pin 13
'   Channel 6 = South Plot Hi O2 Sensor 1 Hi Pin 14, Low Pin 15
'   Channel 7 = South Plot Hi O2 Sensor 2 Hi Pin 16, Low Pin 17
'   Channel 8 = North Plot Low O2 Sensor 1 Hi Pin 19, Low Pin 20
'   Channel 9 = North Plot Low O2 Sensor 2 Hi Pin 21, Low Pin 22
'   Channel 10 = North Plot Hi O2 Sensor 1 Hi Pin 23, Low Pin 24
'   Channel 11 = North Plot Hi O2 Sensor 2 Hi Pin 25, Low Pin 26
'********
'ReDim SumAdata(13)
'ReDim Data(NumPoints)
For I = 4 To 11
   MemHandle = cbWinBufAlloc(NumPoints)
   If MemHandle = 0 Then Stop

'LowChannel = i
'HighChannel = i
'CBCount& = NumPoints% ' Total number of data points to collect
'CBRate& = 90000           'Sampling rate in HZ per channel
      Gain = Voltrange(I)
'Options = CONVERTDATA
   ULStat% = cbAInScan(Board1, I, I, NumPoints, CBRate&, Gain, MemHandle,
         Options)
   If ULStat% = 91 Then
      ULStat% = cbErrHandling(DONTPRINT, DONTSTOP)
   ElseIf ULStat% <> 0 Then
      Stop
   End If
' transfer values from Windows buffer to data array used by VBA
   ULStat% = cbWinBufToArray(MemHandle, ADData%(0), FirstPoint, NumPoints)
   ULStat% = cbWinBufFree(MemHandle)      ' Free up memory for use by
   If ULStat% <> 0 Then Stop              ' other programs

   If ULStat% <> 0 Then Stop
   SumAdata(I) = 0
   For J = 0 To NumPoints
      ULStat = cbToEngUnits(Board1, Gain, ADData%(J), Data(J))
      SumAdata(I) = SumAdata(I) + Data(J)
   Next J
   A_Data(I) = SumAdata(I) / NumPoints
'MsgBox (A_Data(i%))
Next I
```

```
Refresh          ' required to update the screen

End Sub
```

Private Sub Analog_In4() 'used for channel 20 thru 21

```
'**********
' Analog Channel inputs PCI-DAS6031 Board 1
'   Channel 20 = South Old Pressure transducer Hi pin 60, Low Pin 61
'   Channel 21 = North Old Pressure transducer Hi Pin 62, Low Pin 63
'********
'ReDim SumAdata(13)
'ReDim Data(NumPoints)
For I = 20 To 21
   MemHandle = cbWinBufAlloc(NumPoints)
   If MemHandle = 0 Then Stop

'LowChannel = i
'HighChannel = i
'CBCount& = NumPoints% ' Total number of data points to collect
'CBRate& = 90000          'Sampling rate in HZ per channel
      Gain = Voltrange(I)
'Options = CONVERTDATA
   ULStat% = cbAInScan(Board1, I, I, NumPoints, CBRate&, Gain, MemHandle,
         Options)
   If ULStat% = 91 Then
      ULStat% = cbErrHandling(DONTPRINT, DONTSTOP)
   ElseIf ULStat% <> 0 Then
      Stop
   End If
' transfer values from Windows buffer to data array used by VBA
   ULStat% = cbWinBufToArray(MemHandle, ADData%(0), FirstPoint, NumPoints)
   ULStat% = cbWinBufFree(MemHandle)     ' Free up memory for use by
   If ULStat% <> 0 Then Stop             ' other programs

   If ULStat% <> 0 Then Stop
   SumAdata(I) = 0
   For J = 0 To NumPoints
      ULStat = cbToEngUnits(Board1, Gain, ADData%(J), Data(J))
      SumAdata(I) = SumAdata(I) + Data(J)
   Next J
   A_Data(I) = SumAdata(I) / NumPoints
'MsgBox (A_Data(i%))
Next I
'********
' Convert voltages to real world values
'********
```

```
For I = 20 To 21
   A_Data(I) = (A_Data(I) - Intercept(I)) * Slope(I)

Next I
For J = 2 To 3
   TankPres(J).Caption = Format$(A_Data(J + 18), "0.00")
Next J

Refresh            ' required to update the screen

End Sub
```

Private Sub FillTime() 'Time to fill pumps

```
         Start = Timer
            Do While Timer < Start + FilSec 'FillSec it the seconds
               If Timer < Start Then Start = Start - 86400 'Timer crossed midnight
               DoEvents   ' Yield to other processes
            Loop
End Sub
```

Private Sub PressureTime() 'Time to fill pumps

```
         Start = Timer
            Do While Timer < Start + PresSec 'PressureSec it the seconds
               If Timer < Start Then Start = Start - 86400 'Timer crossed midnight
               DoEvents   ' Yield to other processes
            Loop
End Sub
```

Private Sub VentTime() 'Time to pressure vent lines

```
   Start = Timer
       Do While Timer < Start + VentSec
        If Timer < Start Then Start = Start - 86400 'Timer crossed midnight
         DoEvents   ' Yield to other processes
       Loop
End Sub
```

Private Sub Plot2()

```
  Dim Graph() As Single
  Dim x As Integer
  Dim myXarray() As Double
  Dim myYArray0() As Double
  Dim myYArray1() As Double
  Dim myYArray2() As Double
  Dim myYArray3() As Double
  Dim myYArray4() As Double
```

```
    Dim myYArray5() As Double
    Dim myYArray6() As Double
    Dim myYArray7() As Double
    Dim myYArray8() As Double
    Dim myYArray9() As Double
    Dim myYArray10() As Double
    Dim myYArray11() As Double
    Dim myYArray12() As Double
    Dim myYArray13() As Double

'   Dim myYArray1()
    ReDim myXarray(N)
    ReDim myYArray0(N)
    ReDim myYArray1(N)
    ReDim myYArray2(N)
    ReDim myYArray3(N)
    ReDim myYArray4(N)
    ReDim myYArray5(N)
    ReDim myYArray6(N)
    ReDim myYArray7(N)
    ReDim myYArray8(N)
    ReDim myYArray9(N)
    ReDim myYArray10(N)
    ReDim myYArray11(N)
    ReDim myYArray12(N)
    ReDim myYArray13(N)
    'Generate some x y data.
    myXarray(0) = DaTime(1)
    myYArray0(0) = O20(1)
    myYArray1(0) = O21(1)
    myYArray2(0) = O22(1)
    myYArray3(0) = O23(1)
    myYArray4(0) = O24(1)
    myYArray5(0) = O25(1)
    myYArray6(0) = O26(1)
    myYArray7(0) = O27(1)
    myYArray8(0) = Pres0(1)
    myYArray9(0) = Pres1(1)
    myYArray10(0) = Temp0(1)
    myYArray11(0) = Temp1(1)
    myYArray12(0) = Temp2(1)
    myYArray13(0) = Temp3(1)

'   MsgBox N
    For x = 1 To N
      myXarray(x) = DaTime(x) 'value for X-axis
```

```
    myYArray0(x) = O20(x) 'value for Y-axis
    myYArray1(x) = O21(x)
    myYArray2(x) = O22(x)
    myYArray3(x) = O23(x)
    myYArray4(x) = O24(x)
    myYArray5(x) = O25(x)
    myYArray6(x) = O26(x)
    myYArray7(x) = O27(x)
    myYArray8(x) = Pres0(x)
    myYArray9(x) = Pres1(x)
    myYArray10(x) = Temp0(x)
    myYArray11(x) = Temp1(x)
    myYArray12(x) = Temp2(x)
    myYArray13(x) = Temp3(x)

  Next x
'  .Series(0).AddXY .Series(0).XValues.Last + 1, .Series(0).YValues.Last /
        .Series(0).YValues.Last - 1 + (Rnd(100) - (100 / 2)), "", clTeeColor


  With TChart1
   .AddSeries scPoint
   .Series(0).AddArray UBound(myYArray0), myYArray0(), myXarray()
   .Series(0).XValues.DateTime = True
'   .AddSeries scPoint
   .Series(1).AddArray UBound(myYArray1), myYArray1(), myXarray()
   .Series(1).XValues.DateTime = True
   .Series(2).AddArray UBound(myYArray2), myYArray2(), myXarray()
   .Series(2).XValues.DateTime = True
   .Series(3).AddArray UBound(myYArray3), myYArray3(), myXarray()
   .Series(3).XValues.DateTime = True
   .Series(4).AddArray UBound(myYArray4), myYArray4(), myXarray()
   .Series(4).XValues.DateTime = True
   .Series(5).AddArray UBound(myYArray5), myYArray5(), myXarray()
   .Series(5).XValues.DateTime = True
   .Series(6).AddArray UBound(myYArray6), myYArray6(), myXarray()
   .Series(6).XValues.DateTime = True
   .Series(7).AddArray UBound(myYArray7), myYArray7(), myXarray()
   .Series(7).XValues.DateTime = True
   .Series(8).AddArray UBound(myYArray8), myYArray8(), myXarray()
   .Series(8).XValues.DateTime = True
   .Series(9).AddArray UBound(myYArray9), myYArray9(), myXarray()
   .Series(9).XValues.DateTime = True

  End With
```

```
    With TChart2
    .AddSeries scPoint
    .Series(0).AddArray UBound(myYArray10), myYArray10(), myXarray()
    .Series(0).XValues.DateTime = True
    .Series(1).AddArray UBound(myYArray11), myYArray11(), myXarray()
    .Series(1).XValues.DateTime = True
    .Series(2).AddArray UBound(myYArray12), myYArray12(), myXarray()
    .Series(2).XValues.DateTime = True
    .Series(3).AddArray UBound(myYArray13), myYArray13(), myXarray()
    .Series(3).XValues.DateTime = True

    End With



     Refresh

End Sub
```

APPENDIX 7. Parm Code

```
Option Explicit
Public Slope()
Public Intercept()
Public Range()
Public Voltrange()
'Public MM As Integer ' Number of measurements+2 between recording in file i.e. if MM
        = 1 every third recorded is recorded
'Public NN As Integer ' Number of records in a file before creating a new file
Public N As Integer
Public O2Set(1)
Public O2()
Public A_Data()
Public fname
Public fname2
Public fname3
Public SouthHiPSet
Public SouthLoPSet
Public NorthHiPSet
Public NorthLoPSet
Public SWaterPOut  'South Flow Controller Anaog Output 0-5 VDC
Public SChemPOut  ' South Chemical Metering pump Control Voltage 0-10 VDC
Public NWaterPOut  ' North Flow Controler Analog ZOutput 0-5 VDC
Public NChemPOut  ' North Chemicla metering Pump Control Voltate 0-10 VDC
Public PresSec
Public FilSec
Public cyclecount
Public VentSec
Public CalTime


Public Sub param()
ReDim Slope(21)
ReDim Intercept(21)
'ReDim Range(19)
ReDim Voltrange(21)
CalTime = 240   'number of seconds allowed to equibilarate for calibration
'MM = 10 'number of measurements between recorded values
'NN = 1000 ' number of records in a data file
'********
' Set Pump Speed Variables
'*********
SWaterPOut = 0.5
SChemPOut = 0.194 '0.59 ml/min arm on pump right side 1 no ethanol
NWaterPOut = 0.5
```

```
NChemPOut = 0.5535 ' 1.5ml/min = 1 volt input target flow rate 0.83 ml/min measured
        0.847 ml/min
             ' arm on pump right side 4 with ethanol
cyclecount = 20 ' number of cycles with no plot flow must be greater than or equal to 3
        the actual
        ' pumping rate must be greater than the average desired application rate this was
        ' necessary to allow drying of the piping to minimize chemical degredation in the
        ' delivery system and plugging due to biological growth in the piping system
        average
        ' flow rate = pumping rate/cyclecount estimate the time for a pluse
        ' by taking the difference in times data are recorded divided by MM
'*******
' Well water pump timing variables
'*******
PresSec = 60 ' number of seconds to pressurize pumps
FilSec = 20 'number of seconds to vent the pumps
VentSec = 3 ' number of seconds to air purge distribution lines

O2Set(0) = 15 ' Oxygen set point South Plot
O2Set(1) = 15 ' Oxygen set point North Plot
SouthHiPSet = 10 ' Pressure set opint for water tank high pressure South Plot
SouthLoPSet = 7.5 ' Pressure set point for water tank low pressure south plot
NorthHiPSet = 10 ' Pressure set point of water pressure tank North plot
NorthLoPSet = 7.5 ' Lo pressure set point North plot
'*******
' Settings for the A/D operations
'*******
Slope(0) = 100 ' Temp South Vault
Slope(1) = 100 ' Temp North Vault
Slope(2) = 100 ' Temp Shed
Slope(3) = 100 ' Temp Air Under Shed
Slope(4) = 370 ' Preliminary Estimate of O2 South Lower 1
Slope(5) = 370   ' O2 South Lower 2
Slope(6) = 370   'O2 South Upper 1
Slope(7) = 370   'O2 South Upper 2
Slope(8) = 370   'O2 North Lower 1
Slope(9) = 370   'O2 North Lower 2
Slope(10) = 370  'O2 North Upper 1
Slope(11) = 370  'O2 North upper 2
Slope(12) = 7.5    'South pres
Slope(13) = 7.5    'North pressure
Slope(14) = 1
Slope(15) = 1
Slope(16) = 1
Slope(17) = 1
Slope(18) = 1
```

```
Slope(19) = 1
Slope(20) = 5 'South Old Pressure transducer
Slope(21) = 5     ' North Old Pressure transducer
Intercept(0) = 0
Intercept(1) = 0
Intercept(2) = 0
Intercept(3) = 0
Intercept(4) = 0
Intercept(5) = 0
Intercept(6) = 0
Intercept(7) = 0
Intercept(8) = 0
Intercept(9) = 0
Intercept(10) = 0
Intercept(11) = 0
Intercept(12) = 1
Intercept(13) = 1
Intercept(14) = 0
Intercept(15) = 0
Intercept(16) = 0
Intercept(17) = 0
Intercept(18) = 0
Intercept(19) = 0
Intercept(20) = 0.444
Intercept(21) = 0.61
Voltrange(0) = UNI5VOLTS ' 0 - 5 volt range Temp Sensor
Voltrange(1) = UNI5VOLTS ' 0 - 5 volt range
Voltrange(2) = UNI5VOLTS ' 0 - 5 volt range
Voltrange(3) = UNI5VOLTS ' 0 - 5 volt range
Voltrange(4) = UNIPT1VOLTS ' 0 - 0.1 volt range oxygen Sensor
Voltrange(5) = UNIPT1VOLTS ' 0 - 0.1 volt range oxygen Sensor
Voltrange(6) = UNIPT1VOLTS ' 0 - 0.1 volt range oxygen Sensor
Voltrange(7) = UNIPT1VOLTS ' 0 - 0.1 volt range oxygen sensor
Voltrange(8) = UNIPT1VOLTS ' 0 - 0.1 volt range oxygen sensor
Voltrange(9) = UNIPT1VOLTS ' 0 - 0.1 volt range oxygen sensor
Voltrange(10) = UNIPT1VOLTS ' 0 - 0.1 volt range oxygen sensor
Voltrange(11) = UNIPT1VOLTS ' 0 - 0.1 volt range oxygen sensor
Voltrange(12) = UNI10VOLTS ' 0 - 10 volt range pressure transducer
Voltrange(13) = UNI10VOLTS ' 0 - 10 volt range pressure transducer
Voltrange(14) = UNI10VOLTS ' 0 - 10 volt range Soutn Chemical Feed pump
Voltrange(15) = UNI10VOLTS ' 0 - 10 volt range North Chemical Feed pump
Voltrange(16) = UNI10VOLTS ' 0 - 10 volt range South Flow Rate
Voltrange(17) = UNI10VOLTS ' 0 - 10 volt range South Flow Totalizer
Voltrange(18) = UNI10VOLTS ' 0 - 10 volt range North Flow Rate
Voltrange(19) = UNI10VOLTS ' 0 - 10 volt range North Flow Totalizer
Voltrange(20) = UNI10VOLTS ' 0 - 10 volt range South Old pressure Transducer
```

```
Voltrange(21) = UNI10VOLTS ' 0 - 10 volt range North Old Pressure Transducer

End Sub
```

APPENDIX 8. Main Form 1 (Code)

```
Private Sub MDIForm_Load()
Port_Hueneme_Display.Show
MainForm1.AutoShowChildren = True
Port_Hueneme_Display.CmdStart.Value = True
End Sub
```

APPENDIX 9. Initial Work Plan

PROPOSAL AND WORK PLAN

Evaluation of Enhancement of Natural Attenuation of BTEX, MTBE and ethanol using infiltration galleries

Carl Enfield
National Risk Management Research Laboratory
U.S. Environmental Protection Agency, Cincinnati, OH
enfield.carl@epa.gov
513-569-7489

Ernie Lory
NFCSC, U.S. Navy
loryee@nfesc.navy.mil
805-982-1299

INTRODUCTION

Port Hueneme NFESC has been studying the MTBE-BTEX plume emanating from the NEX service station near Heritage Park for a number of years. The base has hosted a number of studies and demonstrations of innovative treatment systems and monitoring techniques for remediation both in situ and above ground.

In this work plan, we propose, under new EPA funding, to expand experimental activities previously completed at the Envirogen demonstration area and evaluate the efficacy of infiltration galleries to treat contaminant plumes from fuel sources that have had their oxygen depleted due to natural attenuation process or the properties of the formation geology. There is a large body of evidence that most fuel components are degradable under oxidizing conditions. We believe that infiltration galleries can introduce oxygen into the formation through atmospheric and/or membrane diffusion and thereby provide a potential low cost remedial technology that could be implemented at service stations to limit the transport of contaminants beyond property boundaries; such a system, if proven to work at Port Huemene, it might be useful at the other fuel-contaminated site at Navel facilities. We also believe the experimental design can be conducted with appropriate safeguards to ensure that there is no lasting adverse impact from this experimental activity at the Site. We thus request approval to proceed as proposed in detail below.

The general approach will be to extract reduced groundwater (groundwater with low dissolved oxygen) immediately upgradient of the infiltration gallery and reinject the water in the vadose zone without contact with the air. After establishing and documenting the hydraulic flow field, and demonstrating to ability to control the oxidization status of the infiltrated ground water with or without amendments, fuel components would slowly be added to the injected water to simulate ground water near a leaking storage tank. The injections would be staged initially injecting easily degradable compounds (i.e. ethanol - selected because of California's intent to replace MTBE with ethanol) followed by ethanol combined with BTEX and then combined with MTBE.

THE SITE

Figure 1 presents a map of the BTEX and MTBE plume several hundred feet down gradient of the source area (the service station). We know from years of monitoring that the BTEX concentrations at this site have been strongly attenuated by natural processes and MTBE fluxes have been substantially reduced by *In situ* remedial treatment systems. We plan to utilize the infrastructure previously used by Envirogen shown in Figure 2 to study infiltration galleries as a potential low cost treatment system.

EXPERIMENTAL PLAN

We propose to evaluate the utility and effectiveness of infiltration galleries to enhance the natural attenuation of ethanol, BTEX and MTBE. The motivation for this work is the certainty that such plumes, with or without ethanol, will continue to be present and require management at service stations throughout the country. EPA has already initiated a similar study limited to MTBE and BTEX on the East Coast (Dover AFB) and hopes to use the proposed Port Hueneme test as the example for the West Coast.

We propose to conduct two side-by-side experiments simultaneously. The proposed locations of the two infiltration galleries are illustrated in Figure 3. The proposed location was selected to utilize the infrastructure of the previous Envirogen study and not interfere with ongoing activities at the Port. The propane and oxygen injection wells would be removed or cut off below grade and plugged with bentonite to make room for the infiltration galleries. Selected MTBE/tracer injection wells would be used as extraction wells.

Figure 4 presents vertical schematics of the proposed infiltration gallery experiments. A backhoe will be used to cut two holes through the blacktop after the injection wells were removed. The holes will be outfitted with various water release and oxygen monitoring/releasing devices, etc, and backfilled with non-native coarse sand. Groundwater will be extracted from the aquifer via the former MTBE/tracer injection wells and released into the infiltration galleries. Initially, only groundwater and tracers will be released into the infiltration galleries. This activity will be continued until hydraulic and oxygen control can be established. Then it will be necessary to spike the influent to the infiltration galleries because neither BTEX nor ethanol are present in groundwater near the transect. It may also be necessary to spike the influent with MTBE if insufficient MTBE is present in the groundwater immediately upgradient of the infiltration galleries (which is likely given our past monitoring data). The infiltration galleries will be kept aerobic by controlling the rate at which the influent is added to each gallery, as described in more detail later. It is expected that the chemicals will be degraded during downward infiltration, thus leading to the release of treated water back into the aquifer. Treatment will be confirmed by monitoring wells previously installed for this purpose for the Envirogen study.

The general idea is to conduct two experiments simultaneously, but under different oxidizing conditions. The methods employed allow for rapid and complete cessation of chemical release at the end of the experiments or in case release concentrations or other aspects of the experiment have exceeded contingency criteria discussed below.

Table 1 summarizes the approximate release concentrations that would be typical of aqueous concentrations near a leaking underground storage tank. These concentrations have been utilized at other studies at e.g. Vandenburg AFB to simulate a gasohol release. The goals are: 1) to provide a stable aerobic environment *in situ*, 2) to meter the contaminated water into the infiltration gallery in such a way that the aerobic environment is sustained while the contaminants are degraded by native microbial populations, 3) to provide influent contaminant concentrations that are stable and high enough to allow reliable analysis and quantification, 4) to keep concentrations and total mass low enough so that they will naturally attenuate or can easily be remediated upon the end of the experiment.

We have estimated the total mass of the chemicals to be released. These estimates assume that each 3-m wide infiltration gallery will be dosed with the groundwater migrating through a 3-m wide portion of the sand aquifer immediately upgradient of it. We assume, for the area of these tests a interstitial groundwater velocity of approximately 0.006 m/d (0.194 f/d), a porosity of 0.25, and a saturated thickness of 7 m. The groundwater will be released uniformly across the 1 m by 3 m surface opening (areal extent) of each infiltration gallery. Total duration of the test will be roughly 1.5 years, in order to allow time for final shutdown of the experiments, cleanup of contaminants if required, etc. Prior to final design a pump test will be performed at the site to verify the flux of water passing by the extraction wells. Adjustments will be made in the design based on these measurements.

Table 1. Target characteristics of chemical release

Chemical	Approximate release concentration (mg/L)	Approximate total mass per unit area to be released over 1.5 years (kg/m^2)	Approximate total volume (liquid equivalent) per unit area to be released over 1.5 years ($gallons/m^2$)
Ethanol	500	1.1	0.37
Benzene	5	0.02	0.007
Toluene	15	0.07	0.02
o-Xylene	5	0.02	0.007
MTBE	10	0.04	0.015
Pentaflorobenzoic acid	3	0.001	NA

(tracer)			
Bromide (tracer)	300	0.13	NA

The infiltration galleries will initially be operated to maintain a fixed oxygen concentration (15%) in the soil gas at the one meter depth within each gallery (called a “control plane”). This oxygen level is required to obtain significant MTBE degradation. If the oxygen level drops below the set point the injection pumps will be turned off until oxygen has returned to the desired levels (confirmed by automated *in situ* monitoring). A second oxygen sensor will monitor the oxygen status 30 cm below the water injection as an internal check on system performance.

We propose to build the infiltration galleries in spring 2004, turn on their pumps and monitoring devices, and run groundwater through them for at least 3 weeks. This will allow time for the native microbes to populate the non-native coarse sand backfill (consistent with prior work by UC Davis and also column work completed in prior years with waters from various source areas). It will also allow us to verify our ability to supply sufficient oxygen to oxidize the native groundwater by diffusion alone. If it is not possible to obtain a reasonable flow (flow rate to be determined later) the upper oxygen monitoring plane will be switched to an oxygen supply plane and oxygen will be supplied through the diffusion membrane to supply required oxygen. Flows through the infiltration galleries will be compared to evaluate the variability in the infiltration galleries. We then will introduce ethanol into both lanes for 3-5 weeks. Ethanol is readily degradable. The initial use of ethanol alone permits making sure the oxygen sensing system is functioning properly prior to injecting potentially more slowly degradable contaminants. After this additional shake down period, ethanol will be removed from one of the lanes and the source water source water for both lanes will be changed to include both BTX and MTBE. One lane will represent a service station that has converted to gasohol but had a previous leak containing MTBE and the other a station that did not convert to MTBE. Tracers will be added to both of the lanes periodically and this will be continued for at least 10 months, but no longer than 18 months. During tracer and organic release, we will monitor the source water and downgradient sampling network in two ways: 1) snapshot sampling of all points every month, (the primary information that will be used to evaluate the efficacy of the infiltration galleries), and 2) weekly or more frequent sampling of key points. The anticipated sample load is outlined below.

				Frequency				
Test Phase	Source	Wells	Depths	Daily	Weekly	Monthly	Duration Weeks	Total Samples
Initial tracer	2			1			3	30
		24	3		2		3	432
Ethanol	2				1		5	10
		24	3			3	5	360
MTBE, BTEX, TBA,	2				1		40	80

Ethanol and tracers								
		24	3			1	40	720

EPA will perform the ground water flux measurement. During the flux measurement ground water will be produced (approximately 500 gal) requiring discharge. NFCSC will supply the tank for water collection and arrange for discharge. EPA will install the infiltration galleries and control systems. NFCSC will assume responsibility for sampling, shipping samples to the Robert S. Kerr Environmental Research Center for analysis, and monitoring system operation. EPA will perform all of the chemical analyses. Periodically cocktails will need to be mixed to supply contaminants to the infiltration gallery. This activity will be performed by NFCSC. NFCSC will either forward files stored on the operating computer to the principal investigator or supply a telephone line to permit downloading the data remotely. System failures should be reported to EPA. EPA will be responsible for making system repairs.

CONTINGENCIES

The primary contingency for this work is the ability to shut off groundwater flow and chemical release automatically if the oxygen concentrations in either of the galleries falls below the value needed to sustain biotreatment. If either infiltration gallery is found to be unable to sustain treatment, its use will be discontinued. The monitoring wells can be used to extract whatever small amount of released chemical mass has escaped either infiltration gallery. At most, we expect to create very small plumes of BTX and MTBE in groundwater beneath the infiltration galleries. By far the majority of the mass released will be treated in the infiltration gallery prior to arriving at the water table. The infiltration galleries, however, will give rise to tracer concentrations that are higher than currently present in the groundwater in the area. We expect that only the tracers will migrate significantly beyond the monitoring well field. Finally, we would expect that the originally efficient *in situ* natural attenuation of the BTEX compounds would be reestablished very quickly after we stop releasing chemicals into the infiltration galleries. Thus it is highly likely that no additional measures would be necessary after cessation of the infiltration gallery tests, other than monitoring for a month or two, as there would be no fugitive BTX or ethanol mass *in situ* to manage.

Nevertheless, there is always a chance, however small, of the unexpected happening, i.e. more significant elongation of plumes, more significant concentrations of contaminants in groundwater than anticipated, and/or accidental unplanned releases of the BTX compounds as a result of human error or instrument failure. Thus we have planned two contingencies to be ready, when and if necessary, to handle various levels of problems that might conceivably be encountered. The following table summarizes the two contingencies, which are described in more detail below.

Contingency	What	When	Criterion
1	Pump and treat	During the	Contaminant concentrations in

	using monitoring wells downgradient of experimental zone	experiment	the monitoring wells exceed 10% of the injected concentration of the non tracer compounds (using average values over 2 months of monitoring)
2	Pump and treat using supply and monitoring wells up and downgradient of the experimental zone	After the experiment	Contaminant concentrations exceed pre-injections concentrations by 20% of the non tracer compounds referring to averages of concentrations in all wells (monitoring and supply) utilized for this work

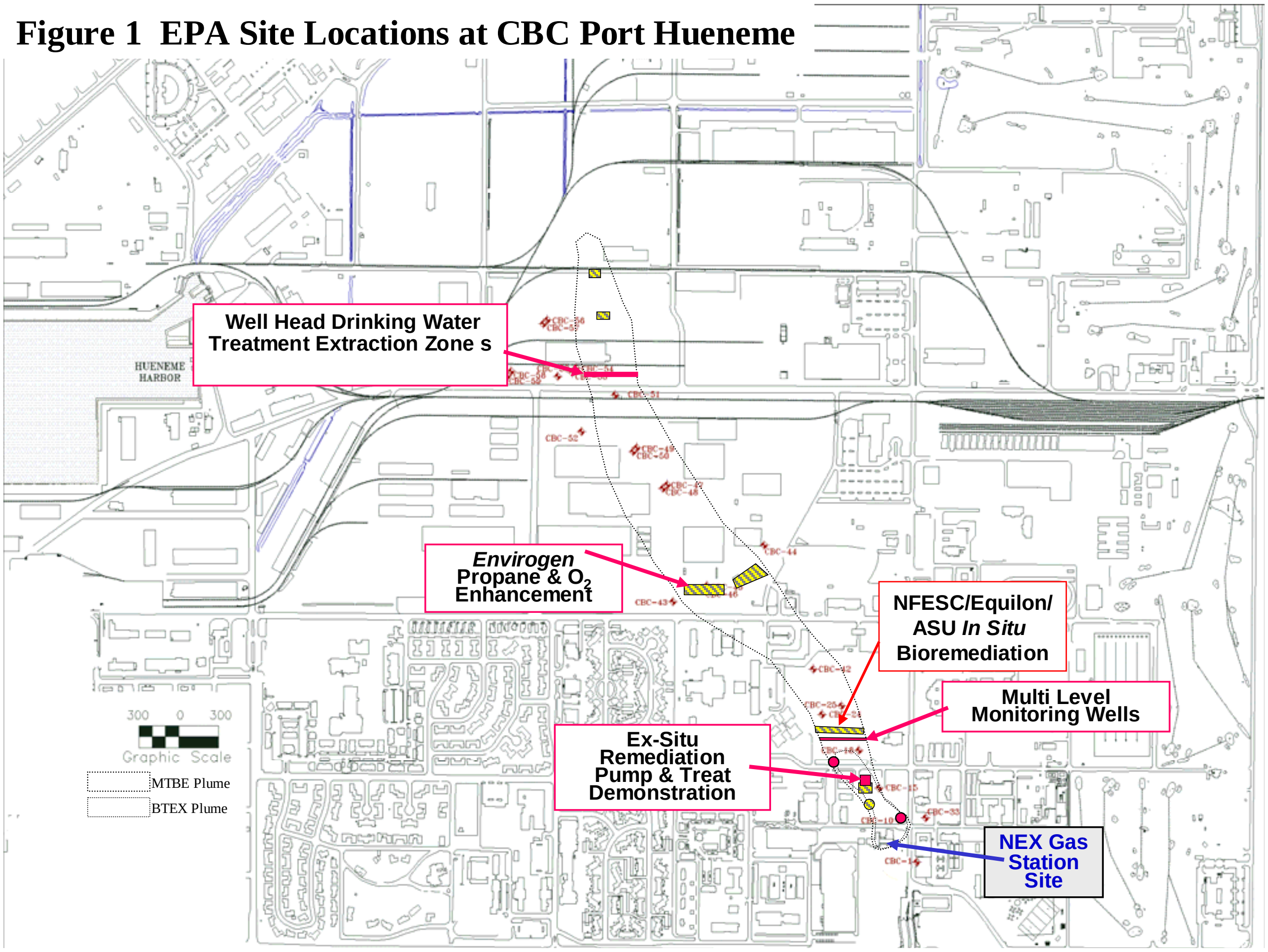


Figure 1. Map showing various features of NEX contamination plume with selected studies that have already been performed

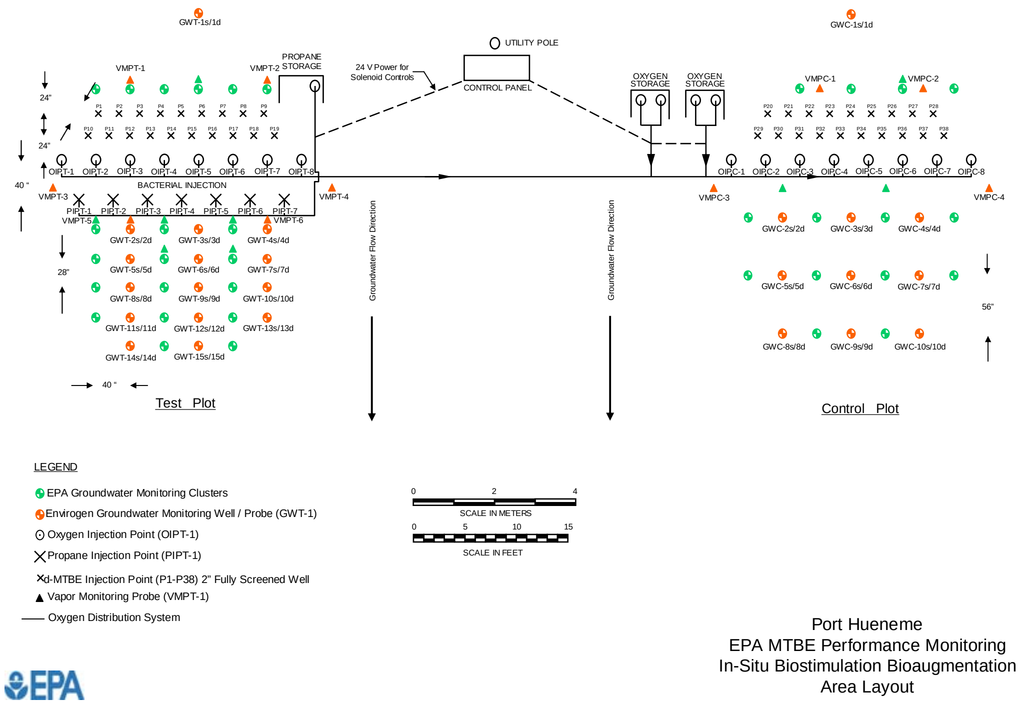
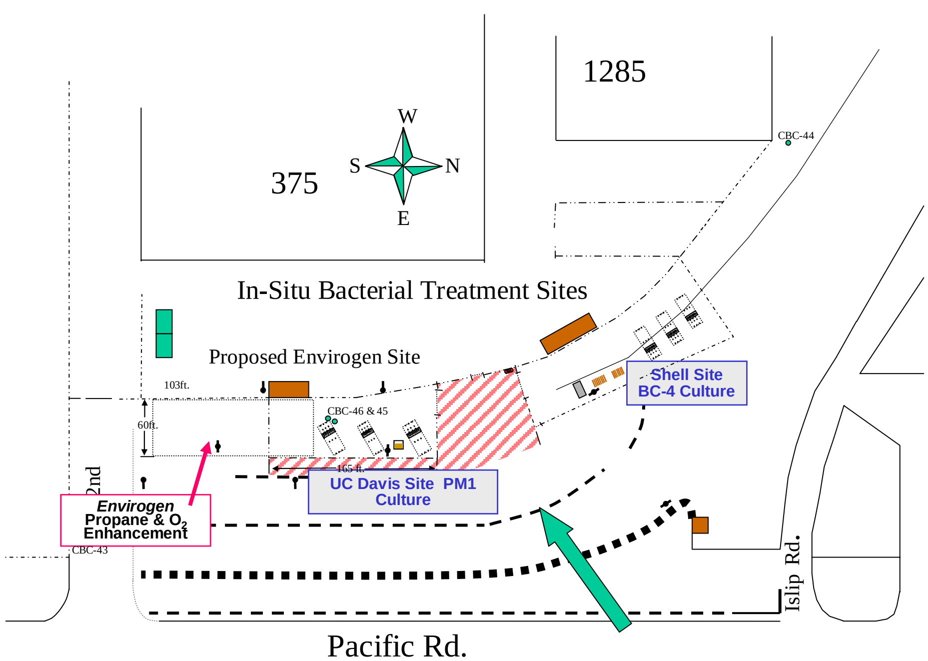


Figure 2 EPA Site Location, Middle of Plume



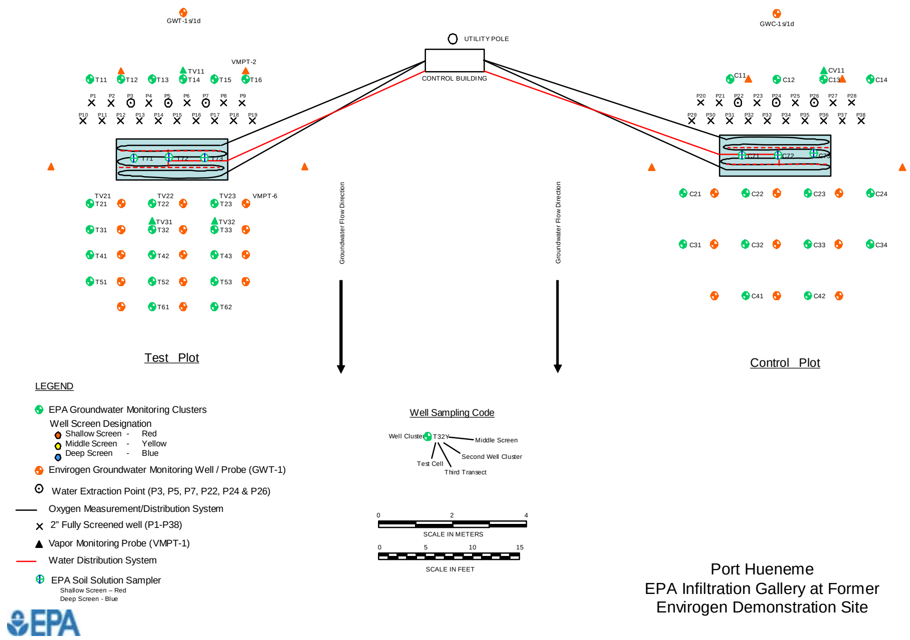


Figure 3. Illustration of the concept for the side-by-side infiltration galleries. A backhoe will be used to cut two holes into the Vadose Zone. The holes will be outfitted with various distribution and monitoring devices, etc. (shown only in the plan view), and backfilled with coarse sand. Groundwater will be extracted from the aquifer via the supply wells, spiked with chemicals of interest (see text) and released into the infiltration galleries. The infiltration galleries will be kept aerobic as described in the text. It is expected that the chemicals will be degraded during downward infiltration, thus leading to release of treated water back into the aquifer. Treatment will be confirmed by monitoring wells previously installed for this purpose. The general idea is to conduct two experiments simultaneously, but under different conditions, i.e. one with release of small amounts of BTX and MTBE and tracers, and the other with the same plus ethanol.

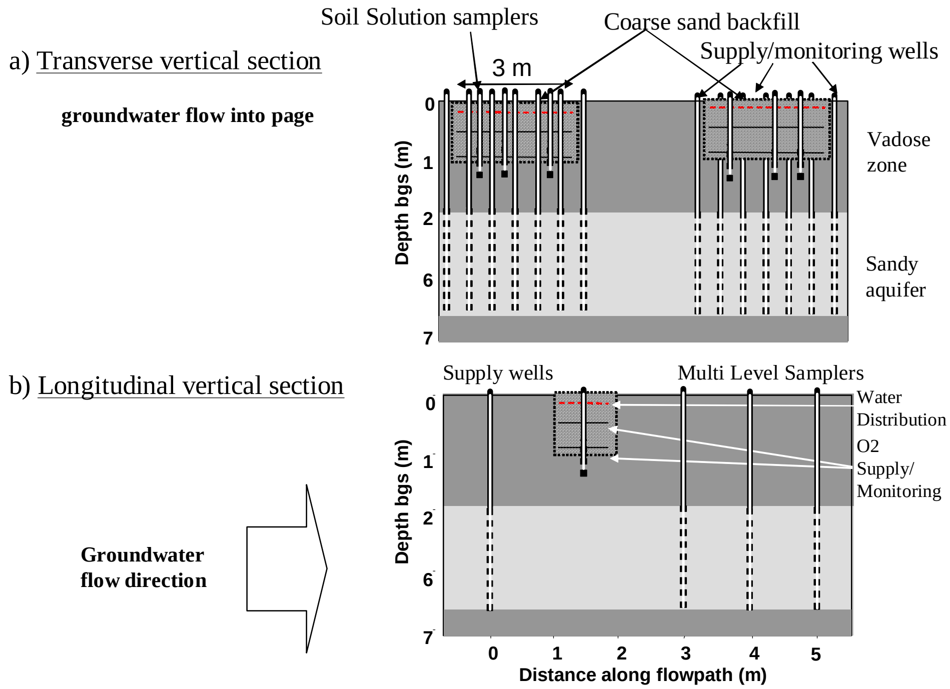


Figure 4 vertical section of proposed infiltration gallery.

APPENDIX 10. Partial Parts List

Part No.	Desciption	Source	Unit Price	Quantity	Cost	Comments
A-07193-00	Drying Column	Cole Parmer	62	4	$248.00	Gas Sampling lines
A-07193-05	Indicating Drierite desicant	Cole Parmer	121	1	$121.00	
9989K53	1/3 hp sump pump with diaphram switch	McMaster Carr	108.83	2	$217.66	Need 18" diameter by 24" depth sump for operation
5272K234	Panel Mount Coupling 1/8" OD	McMaster Carr	5.01	2	$10.02	Air supply to empty distribution pipe
5272K233	Panel Mount Coupling 1/2" OD	McMaster Carr	9.18	6	$55.08	to water lines and pump dicharge from sump pump
5272K296	1/2" Tube x 3/4" Male Pipe	McMaster Carr	6.56	2	$13.12	Sump pump Adaptor
4596K428	1 1/2 x 3/4 Reducing bushing	McMaster Carr	3.36	2	$6.72	Sump pump Adaptor
1976K67	3.15" square fan 115VAC	McMaster Carr	22.73	2	$45.46	Ventalating fan
36895K116	2" bulkhead fitting	McMaster Carr	20.91	4	$83.64	piping adaptor for vent
2389K213	2" solvent weld street ell	McMaster Carr	1.32	12	$15.84	vent through side of tank. No holes in top
8111K94	120VAC Buna-N 3 way universal solenoid	McMaster Carr	81.6	12	$979.20	Gas Sampling claibration valves
8077K48	120VAC solenoid	McMaster Carr	98.89	2	$197.78	Water line shutoff normally closed
8077K42	120VAC solenoid	McMaster Carr	38.73	2	$77.46	Air valve in distribution system
7219K25	2 Gang Weather Proof Box	McMaster Carr	8.58	2	$17.16	For sump pumpand vent fan
7219K64	Duplex receptical cover	McMaster Carr	11.05	2	$22.10	
7310K12	Liqid tight Cord Grip	McMaster Carr	1.75	4	$7.00	
7310K14	Liqid tight Cord Grip	McMaster Carr	2.51	2	$5.02	
PCI-DAS6033	64Channel 16 Data I/O Card	Measurement Computing®	1125	1	$1,125.00	
02F5345	5 conductor 16 AWG Cable 100 ft spool OD = 0.504"	Newark	68.35	1	$68.35	Power except for pump
02F4871	5 pair shielded 24 AWG Cable 100 ft spool OD 0.289"	Newark	35.64	1	$35.64	

02F4063	18AWG 3 Conductor Cable 100' OD 0.265"	Newark	89.9	1	$89.90	
PXM41MD0-1.60BARG5V	Heavy duty metric pressure transducer with 0 to 1.6 Bar range, G 1/4 male fitting, 3 meter cable and 0.5 to 5.5 Vdc output	Omega Engineering	500	2	$1,000.00	12 Week delivery
FL-3607G	5.77 cc/min of air .07 cc/min of water, Aluminum Frame, 65mm tube	Omega Engineering	119	5	$595.00	
FLV2011	Flow controler 20000 sccm	Omega Engineering	2095	2	$4,190.00	Don't know fitting size will need to get after received
4155	85 gal Polyethylene tank with external flange	US Plastics	159.64	2	$319.28	For Sump will be used to replace one of the valve boxes will come above grade but instuments will be laced below grade
4205	Cover for tank	US Plastics	29.06	2	$58.12	bolts on should be water tight
	Air chuck				$0.00	On pressure tank to pressurize
				Total	$9,603.55	