35

Acoustic Boiling Detection

by

George Daniel Doney

B.S. Systems Engineering US Naval Academy 1992
B.S. Economics US Naval Academy 1992

Submitted to the Department of Nuclear Engineering in Partial Fulfillment of the Requirements for the Degree of

Master of Science

at the

Massachusetts Institute of Technology

February 1994 [JUNE]

Signature of Author ______________________________
Department of Nuclear Engineering
February 28,1994

Certified by ______________________________
Professor John E. Meyer
Thesis Supervisor

Certified by ______________________________
Professor David D. Lanning
Thesis Reader

Approved by ______________________________
Professor Allan F. Henry
Chairman, Departmental Committee on Graduate Students

Acoustic Boiling Detection

by

G. Daniel Doney

Submitted to the Department of Nuclear Engineering
on February 28, 1994 in partial fulfillment of the
requirements for the Degree of Master of Science
in Nuclear Engineering

Abstract

Experiments were performed to gather data on the effects of changing bulk temperature and power levels on the sound of boiling. This information was formatted and used to train a neural network to recognize boiling. The trained network was then used in an executable program to provide an indication of boiling.

The trained network is an detector predictor of boiling under the experimental conditions. More work must be done before the detector can be reliably used in the reactor. With more development, the detector could possibly be used to provide important coolant information for Pressurized or Boiling Water Reactors.

Thesis Supervisor: **Dr. John E. Meyer**
Title: Professor of Nuclear Engineering

Thesis Reader: **Dr. David D. Lanning**
Title: Professor of Nuclear Engineering

Acknowledgments

I would like to extend a special thanks to the following people who contributed much of their time to help me with my thesis:

Professor **William Clement**: Professor Clement spent a lot of his time helping me to learn C this past summer. His ideas helped shape my work into its current format. He helped debug my programs and allowed me to use the neural network program he had written, DTRC93. He also helped me get the hardware used in the experiment.

Professor **John Meyer**: Professor Meyer was instrumental in helping me to express my ideas. He helped me change my thesis from a garbled mass to somewhat refined project (in a very short time). He also helped me to develop some of the ideas used in the detector.

Professor **David Lanning**: Professor Lanning kept me funded for my stay here at MIT. Without money, I couldn't have done anything. He also helped me to revise my thesis.

Professor **Martin Nelson**: Professor Nelson was very enthusiastic and helpful in offering his facilities at the Naval Academy for my summer work. It was there that I did my first work with highly subcooled boiling (and destroyed my first heater).

Robert Catalano and **Benthos, Inc**: Robert sent two free hydrophones to be used with my experiment. These hydrophones served their purpose extremely well by providing excellent frequency resolution (the best things in life are free).

Hernan Trosman: Hernan helped me to revise my thesis at the end when things became hectic. I'm hoping he'll learn to use Borland C++ so he can advise whoever may come along next to take up the project. He also ran to get me a 9V battery (even though I didn't need it).

Table of Contents

Table of Contents

List of Figures

List of Figures

Introduction

In the next few years, MIT may need to relicense its nuclear research reactor. In order to meet the research needs of the institution into the 21st century, several proposals have been presented which allow the reactor to achieve higher neutron flux. One such proposal is to modify current MITR-II core and raise its power to 10MW. Studies are currently underway to determine if the current core can handle this higher power level. It may be possible, by increasing the flow rate and by reducing uncertainties associated with the current safety system, to allow safe operation of the reactor at higher power with few alterations to the current core.

Reactor limitations are set not just by the physical properties of the reactor but also by the detectability of the conditions in the reactor. If a variable must be estimated through calculations, there will be uncertainties. In order to assure safety, operating limits must be set to take into account the most conservative estimation of the variable. In a system with a number of uncertainties, the limitation may be set far below the physical limitation of the system in order to assure safety. This is the case with the high temperature scram signals of the reactor. Thermocouples located in the reactor hot leg piping are used to determine the average exit temperature of the coolant that leaves the reactor. If the temperature, e.g. 60 degrees C, indicated in the Technical Specifications has been exceeded a scram signal is produced since under certain core circumstances this is an indication that the reactor is approaching a damaging operating condition. Damage to the core is imminent if initiation of boiling causes a flow instability to occur in a channel. A safe operating temperature limit was calculated using correlations. A correlation relating the fixed reactor parameters such as coolant channel thickness with variables such as flow rate, power density, coolant temperature, and pressure is used to determine the onset of flow instability. The expected coolant temperature at the outlet of the channel in which the flow instability may occur can then be determined as a function of channel power by setting the pressure constant (pool type reactor) and using a minimum acceptable core flow rate (1800 gpm). If the maximum acceptable outlet temperature is known, the outlet temperature of the other channels can be determined if the flux distribution is known throughout the core. The temperature of the mixed fluid can be compared to the measured average exit temperature detected in the hot leg piping. This process may produce a scram signal. The temperature for safe operation is set much lower than the physical limitations due to uncertainty in channel conditions (flow rates, cladding thickness, etc. caused by manufacturing errors and changing conditions in the core), disparity between channels (nonuniform flux distribution), uncertainty of data (errors or lack of coverage in instrumentation), and variable recognition time (transport time of coolant).

The onset of flow instability for the MITR core is preceded by boiling in the channel. Therefore, if an occurrence of boiling can be accurately detected soon enough, the reactor can be safely shut down. This is a better method for preventing flow instability than temperature detection, since the boiling information is explicit. For water cooled reactors, when under-cooling conditions occur, boiling precedes dangerous reactor conditions, however, under certain conditions (such as channel blockage) the bulk coolant temperature may not exceed specifications while damage to the core is occurring.

The uncertainty with a boiling detection system is whether or not the onset of boiling is accurately established. With an accurate boiling detection system, therefore, the power can be raised much closer to the physical limitations, allowing better performance and more adequate safety. The engineering challenge is to produce an accurate boiling detector. It must not give false alarms which result in costly shutdowns and it must not miss a boiling occurrence that results in damage to the core.

Many methods have been developed to detect boiling but are not applicable for detection of boiling in the core, however. The method must:

1. detect boiling at any location in the core;
2. work in the harsh environment of the reactor;
3. fit in the limited space of the core;
4. not interfere with the operation of the reactor; and
5. be a financially attractive option.

For example, one method of boiling detection would be to measure temperature fluctuations in the cladding temperature. But, boiling can occur at any point along the core axially or radially, so thermocouples would have to be placed at many points in the core, thus interfering with the operation of the reactor.

Another possibility is neutron flux noise detection. When boiling occurs in the reactor, voiding in the moderator/coolant causes fluctuations in the neutron population in the core. These fluctuations can be used to indicate boiling. Neutron noise boiling detection requires accurate measurement of these fluctuations. Additionally these fluctuations must be measured by multiple detectors axially to verify that the fluctuations are caused by bubbles in the flow. This is a costly alternative due to the cost of neutron detectors.

Acoustic boiling detection meets all of the conditions listed. Since sound travels through the coolant, an acoustic boiling detector would be able to detect boiling at any location. With the proper hydrophone, the detector can work in high flux and high temperature conditions. The hydrophone can be placed in a location so it can be easily replaced in the event that its performance becomes degraded. The hydrophone is small so it does not interfere with reactor operations. Since it does not need to be located in the core, it is not difficult to find a location in the hot or cold leg for it. The detector system is inexpensive since the only hardware is a hydrophone, a data acquisition board, and a computer (the costs of these items are listed in Appendix A).

Acoustic boiling detection is not a new idea. Experiments to measure the acoustic noise have been done on sodium for liquid metal cooled reactors[1], methanol[2], and water cooled reactors[3] with the possibility for use for detection in mind. This paper furthers the research done in the area of boiling detection by developing new techniques for measurement and attempting to apply these techniques to provide a working detector. The method for separating the signal of interest has been improved and a neural network was used to analyze boiling signal. These improvements allow detection and analysis at varying levels of background noise characteristic of nuclear reactors over a range of pressures,

---

1 Srinivasan,G.S.and O.P Singh, "New Statistical Features Sensitive to Sodium Boiling Noise", *Ann. Nucl. Energy*, Vol.17, No. 3, pp.135-138, 1990.

2 Westwater, J.W, A.J. Lowery, Jr., and F.S. Pramuk, "Sound of Boiling", *Science*, Vol 122, 332-333, 1955.

3 Koski, J.A., A.G. Beattie, J.B. Whitley, and C.D. Croessmann, "Experimental Verification of Subcooled Flow Boiling for Tokamak Pump Limiter Designs", The American Society of Mechanical Engineers papers, 345 E. 47 St., New York, N.Y. 10017, 1987.

temperatures, flow rates, and power levels. Potentially, the detector may allow the operator to determine these parameters at the point where boiling is occurring.

Chapter 1: Boiling

In order to anticipate the signal received under different conditions in the core, knowledge of the origin and sound of boiling is required. Initially, subcooled boiling aids the ability to carry away energy from the surface. But as power is increased in a low pressure, plate type reactor, boiling leads to flow instability which may reduce the energy removal capability of the coolant.

There are a number of boiling regimes described in two phase flow books. Two of these regimes, nucleate boiling and film boiling were studied in other investigations to determine the heat transfer characteristics and noise magnitude. One study used methanol as the coolant and produced curves which related the degree of subcooling, the heat flux, and the overall magnitude of the boiling noise (**Figure 1.1**).

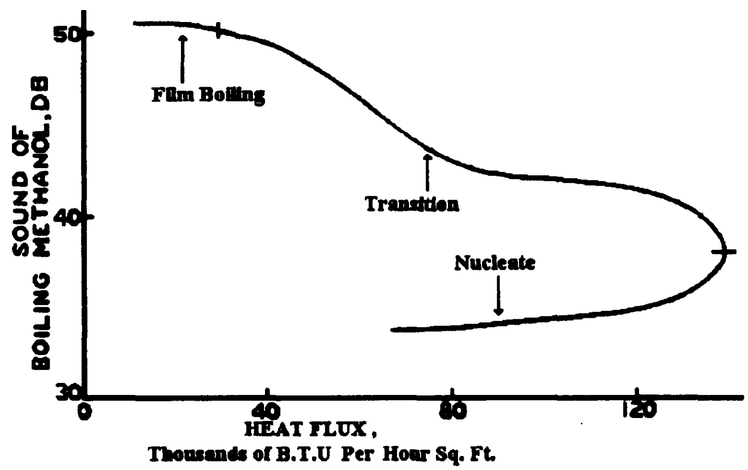


Figure 1.1: Methanol Boiling Curve[4]

The results show that during nucleate boiling, the amount of heat removed dramatically increases with a small increase in the amount of subcooling. Also an increase

---

4 Westwater, ibid.

in heat flux only results in small increases in boiling noise. But at a certain point, the boiling regime undergoes a transition to film boiling. At this transition point (indicated by a dash in **Figure 1.1**), the boiling noise increases dramatically and the heat removal capabilities of the heater are reduced. This causes the temperature to continue to go up and causes a further reduction of the ability of the coolant to remove the heat. At this point, if power is not reduced, damage to the heating element is likely. In the present study, experiments involving film boiling were not made in order to avoid damage to the heating element.

In this chapter, different types of nucleate boiling will be investigated to give a better idea of what to look for when attempting to detect boiling.

1.1 The Ideal Bubble

Boiling is a thermodynamic process in which vapor is produced to form a bubble in the surrounding liquid. In the process of expanding and contracting, the bubble gives off pressure pulses (sound) which travel through the water. An object immersed in water which expands and contracts emits in the far field a pressure wave, P_{ac}:

$$P_{ac} = \frac{\rho}{4\pi r} V'' \qquad \textbf{Eqn. 1.1}^{5}$$

where:

ρ= density of the liquid;
V''= second time derivative of the emitting body volume evaluated at time t-r/c with
c= speed of sound in media; and
r= distance from source

---

5 Nesis, Y. I., "Acoustic Noise of a Boiling Liquid", Heat Transfer - Soviet Research, Vol. 22, No. 6, 1990. Translated from Two-Phase Flow: Heat Transfer and Unsteady-State Processes in Turbomachinery. USSR Acad. Sci., pp. 68-74.

Therefore the creation and collapse of vapor bubbles will create an associated pressure pulse which can be picked up as sound.

If all other parameters are kept constant, as the power introduced to the liquid is raised, distinct changes in the type of bubble formed will occur, causing changes in the sound associated with boiling. Understanding these changes is important for detection of boiling.

1.2 Boiling Types

Three types of boiling are listed in a paper entitled "Acoustic Noise of a Boiling Liquid"[6]. First, vapor oscillation which occurs as microscopic gas bubbles perform free radially-symmetric pulsations with a cyclic frequency in the bulk of the fluid is discussed. These oscillations create very low magnitude pressure pulses and are not important to this analysis.

As the amount of the heat transfer to the liquid is increased, the near wall layer adjacent to the heating surface will experience a sharper temperature gradient. If the temperature gradient is large enough the liquid at the wall surface will reach saturation temperature. This situation results in subcooled boiling since the bulk temperature is still below saturation.

---

6 Nesis, ibid.

Bergles-Rohsenow Eqn:[7]

$$q''_{ONB} = 15.6 \bullet p^{1.156}\left[\Delta T_{SAT}\right]^{\frac{2.3}{p^{.0234}}} \qquad \textbf{Eqn 1.2}$$

where:

q''_{ONB} (BTU/hr m^2)= Heat flux necessary for onset of nucleate boiling (ONB);
P (psi)= pressure at the boiling site; and
ΔT_{SAT} (deg F)= T_{WALL}-T_{SAT}..

Notice that the greater the degree of bulk liquid subcooling, the more heat flux is required to cause boiling. Therefore, as the bulk temperature increases, less power is needed to cause boiling.

The rest of Chapter 1.1 is a summary of information presented in "Acoustic Noise of a Boiling Liquid"[8] The equations listed are taken from this paper.

Subcooled boiling is characterized by rapid growth and separation of small vapor bubbles that exist at the heat source. As these bubbles depart from the heating surface, first they increase in size due to the evaporation of superheated or saturated liquid near wall layers. At some point, the bubble will come in contact with cooler liquid and the expansion will stop with the bubble reaching some maximum radius, R_m. The bubble will then collapse as its energy is absorbed into the surrounding liquid. The time that the bubble takes to reach cooler water will decrease if the temperature gradient is higher. The

---

[7] Todreas, N.E., Kazimi M.S. Nuclear Systems I. Hemisphere Publishing Corporation, 1990, page 534, Eq. 12.16

[8] Nesis, ibid.

temperature gradient, remember, is a function of the subcooling, $\Delta T_{sub} = T_{sat}-T_{bulk}$. The bubble lifetime, τ_0, is inversely proportional to its maximum size, R_m.

$$\tau_0 = \frac{k}{\Delta T_{sub}}, \qquad \textbf{Eqn 1.3}$$

$$R_m = k_2 * \Delta T_{sub} \qquad \textbf{Eqn 1.4}$$

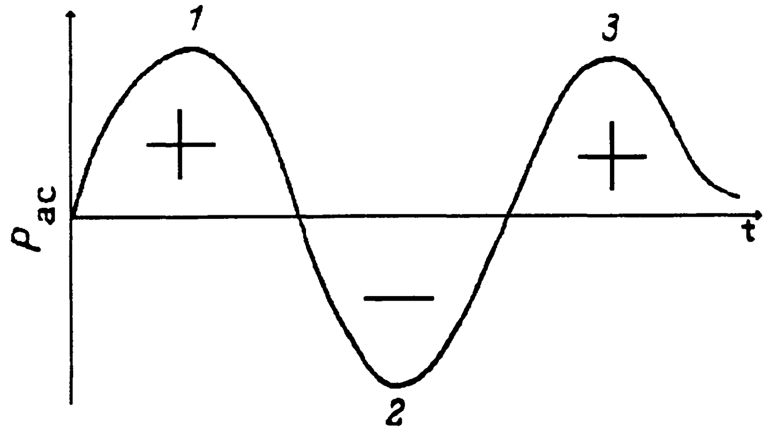
The field generated by a multitude of such bubbles is the sound of subcooled boiling. The bubbles may vary slightly in size, but there will be a mean size, the variation about which depends on the uniformity of the boundary layer. In order to determine the frequency spectrum of subcooled boiling, the spectrum created by a single bubble of mean size will be investigated. Assuming that the bubbles are spherical in shape, the bubble volume as a function of radius (R) can be inserted into **Equation 1.1** yielding:

$$P_{ac} = \frac{\rho}{r}R[2(R')^2 + (R * R'')] \qquad \textbf{Eqn 1.5}$$

Analysis of this equation gives an idea of the acoustic pulse associated with the ideal bubble (**Figure 1.2**). With subcooled boiling, the radius of the bubble increases from 0 to R_m, then decreases again to 0. At small values of R, the first term in brackets on the RHS of **Equation 1.5** is obviously greater than the second term. Therefore, early in the life of the bubble, while it is small but growing, a compression pulse will be emitted (Condition 1). When the bubble is close to its maximum radius, the first term virtually disappears. At this point the second term is negative, therefore the bubble emits a rarefaction pulse

(Condition 2). As the bubble begins to collapse, the collapse decelerates causing another compression pulse (Condition 3). This conclusion matches the experimental measurements of a single bubble in a subcooled liquid.

Figure 1.2: Pressure Pulse of Bubble in Subcooled Liquid[9]



Changing the conditions, under which boiling occurs will change the magnitude and may alter the period of the pressure pulse, but the pulse will not change qualitatively.

The spectral density S(ω) of the pulse can be found by taking the Fourier integral:

$$S(\omega) = \int_{-\infty}^{\infty} P_{ac}(t) \bullet e^{i\omega t} dt \qquad \textbf{Eqn 1.6}$$

The frequency (ω_{max}) of maximum amplitude, $|S(\omega)|_{max}$, produced by the a subcooled bubble with the $P_{ac}(t)$ shown in **Figure 1.2** is:

---

9 Nesis, ibid.

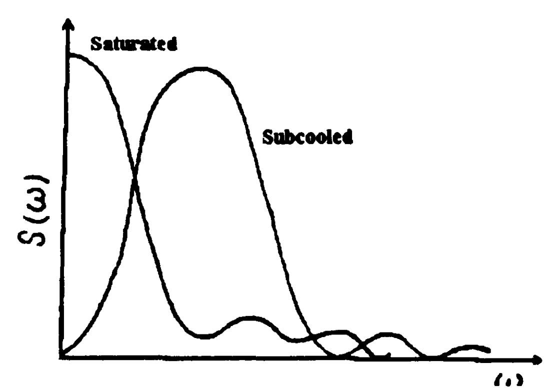
$$\omega_{max} = \frac{2}{3 \times \tau_0}, \qquad \textbf{Eqn 1.7}$$

since there are 3 sign changes in the pulse.

The sound which results from a multitude of bubbles (see **Figure 1.3**) with a maximum radius, R_m, which is distributed about a radius, $R_{m,mean}$, can be easily verified since the spectrum of the sum of the signals is equal to the sum of the spectra of each signal taken separately.

Under conditions of subcooled boiling there is no strict repetition of pressure pulses emitted by bubbles. Therefore, the resulting spectrum is continuous with a broad maximum. Also, equidistant narrow maxima can be isolated that correspond to the periodic repetition of the origination and collapse of bubbles (see Figure 1.3).

Figure 1.3: Spectral Density of Pressure Pulse in Subcooled and Saturated Liquid



If the bulk temperature of the fluid is at saturation, a third type of bubble will be formed. This bubble is characterized by its large size. An investigation of the pressure pulse (**Equation 1.5**) reveals a radically different shape. At the surface of the heater, the bubble is very small but growing. As the bubble continues to grow, the pressure pulse gets larger. After a time, a temperature decrease at the interface with the liquid becomes noticeable and the rate of growth decreases:

$$R'(t) \sim \sqrt{t}$$

Substituting R(t) into **Equation 1.5** results in a pressure pulse tapering to zero. But, since the bubble never collapses, there is no rarefaction pulse. Notice that a major portion of the frequency spectrum associated with boiling is in the infrasonic range (frequency below zero) (see **Figure 1.3**), and therefore cannot be heard.

Subcooled boiling noise is the best indicator of trouble necessitating shutdown. The temperature of the coolant in the MIT reactor is such that saturated boiling is not likely to occur. Other methods of shutdown, such as temperature detection prevent bulk temperatures from getting to this point. Also, since subcooled boiling will likely precede boiling which causes instability in the channel, detection of subcooled boiling is enough. Detecting saturated boiling alone does not exclude the possibility of damage to a fuel element since damage to an element can occur without the bulk temperature ever reaching saturation. Therefore, it is clear that detection of subcooled boiling is the case of interest in the present study.

Fortunately, subcooled boiling is both a better indicator of trouble and easiest to detect (not only because it is louder, but also because its characteristic frequencies are above the range of most commonly occurring background noises). Low frequencies, like those in which saturated boiling occurs, are cluttered with background noise.

1.3 Experimental Results

The data gathered in the experiments of the present study can be analyzed with the background information from the previous section. Altering the conditions in which boiling occurs changes the characteristics of the bubble which in turn affects the sound of boiling. Understanding the differences is essential for detection in the reactor since the detector may be required to identify boiling under any of these conditions.

Experiments were performed using the setup shown in **Figure A.1**. For a detailed description of the hardware see Appendix A. First, the heater was turned on to a low power setting (50 kW/m^2) and background samples were taken. This provides a better background frequency spectrum (see Chapter 3.2 FFT) since there is noise while the heater is running even if no boiling was occurring. The background spectrum consists of the average of 50 spectra taken at low power. The background was updated periodically throughout the experiment to account for changes.

The power was then increased and spectra were taken in bursts periodically. Each recorded spectrum is actually the average of 50 spectra minus the background spectrum. 5-20 spectra were recorded in each burst.

When each burst was taken, a value was assigned to each spectrum in the burst to tell whether and what type of boiling was occurring. These values could be changed later on further inspection. Also the bulk temperature of the fluid, measured approximately 15 cm from the heater, was taken in order to demonstrate the effects of changing temperature on the boiling noise.

The hydrophone placement was changed with each experiment to make sure that the spectra obtained had no directional dependence. In order to simulate flow conditions the heater was shaken in a wide path. The estimated velocity of the heater was 1 m/sec. Also, all frequencies below 0.5 kHz were clipped in each of the plots shown. Frequencies lower than this are dominated by background noise like amplifier bias and flow noise. The frequency domain of the voltages produced by the amplifier are plotted. The relative amplitude between plots is partially dependent on the amplifier gain which may change from day to day. Any plots with more than one spectrum were taken at nearly the same time so the amplifier gain will not be a factor.

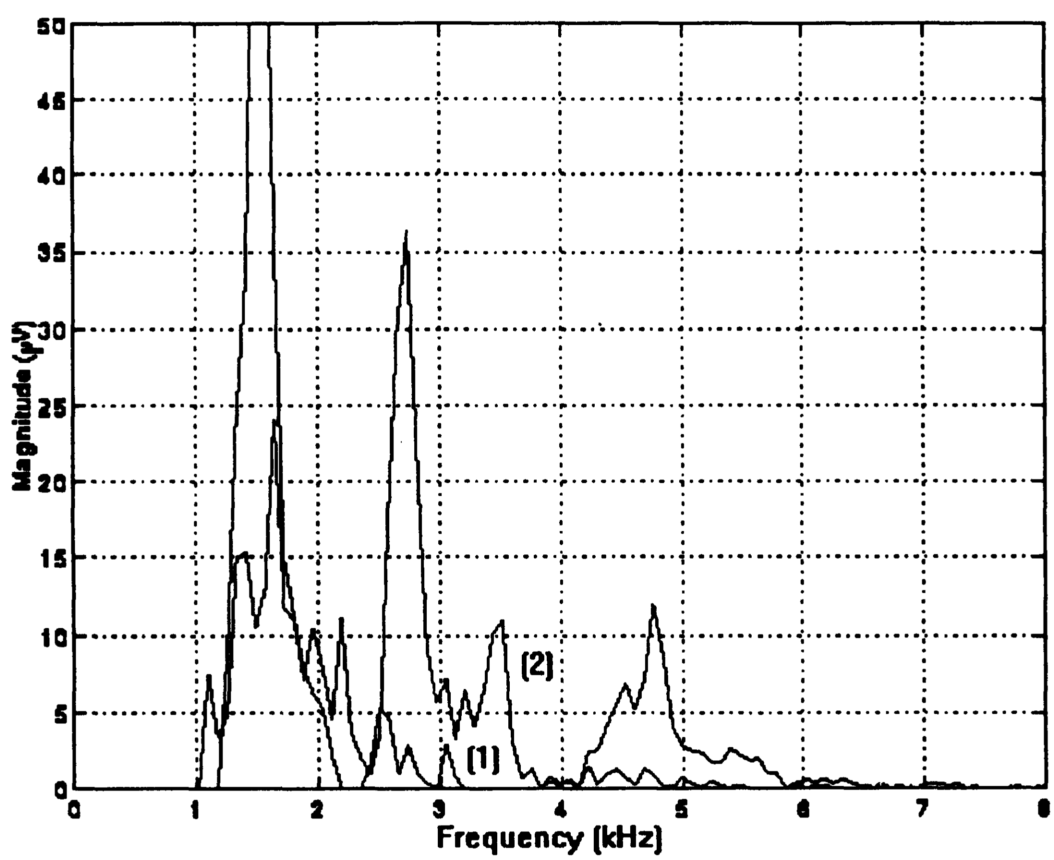
An increase in the subcooling decreases the lifetime of the bubble **(Equation 1.3)**. The more violent growth and collapse results in a louder peak sound which is centered about a higher frequency. Also, there is less variation in the bubble lifetimes, since the boundary layer is thinner, resulting in a more defined frequency peak.

Increasing the pressure also affects the vapor bubble. A higher pressure with a given bulk temperature increases the amount of subcooling. Also, the thermodynamic equilibrium is changed resulting in a bubble which is differently shaped giving a different frequency spectrum. Studies must be done at high pressures to determine the exact correlation between the frequency spectrum and the pressure.

Impurities in the coolant will have a slight effect on the frequency spectrum. As dissolved non-volatiles come out of solution, they provide a means for vapor bubbles to form. These bubbles are short lived (except the non-volatile part) and may skew the spectrum slightly towards higher frequencies. Impurities affect the saturation temperature as well and increase the degree of subcooling.

The coolant flow rate past the heated surface also affects the frequency spectrum. Turbulent flow reduces the boundary layer at the surface of the heater. This means the bubbles will be short lived. Therefore, the bubbles grow and collapse with tremendous ferocity resulting in a very loud peak which can be over 10 times louder than the no flow peak (**Figure 1.4**). Since the boundary layer is thin, the bubble lifetime is fairly uniform so the peak is very narrow. Interestingly, the center frequency of the peak is very close to the center frequency under no flow conditions (with the same liquid subcooling and pressure).

Figure 1.4 Boiling at 46 deg C, 500 kW/m^2, no flow (1) and ~1 m/sec flow (2)

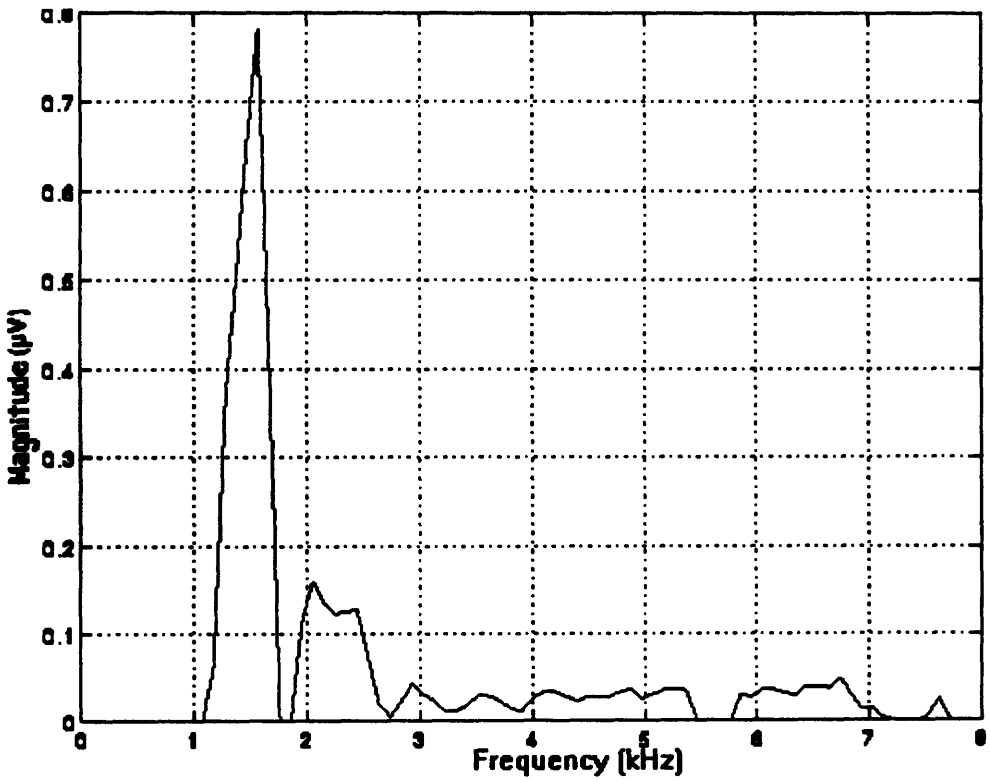


1.4 Subcooled Boiling Stages

Subcooled boiling has been observed to occur in several distinct stages as the heat flux is increased. The onset of nucleate boiling (ONB) occurs after the wall temperature has exceeded the saturation temperature by some amount. This stage of subcooled boiling is marked by a single frequency peak as mentioned in Chapter 2.1.2. Many observations were taken of boiling near its onset. One example of an observed spectrum, **Figure 1.5,** is very similar to the multiple ideal bubble spectrum shown in **Figure 1.3**. .

With the heater at constant power and the temperature nearly constant (observation taken over a period of a few seconds), the magnitude of the peak frequency fluctuated. This is probably due to oscillations in natural circulation past the heated surface. These oscillations had a period on the order of a second.

Figure 1.5 Boiling at 57 deg C, 200 kW/m², no flow

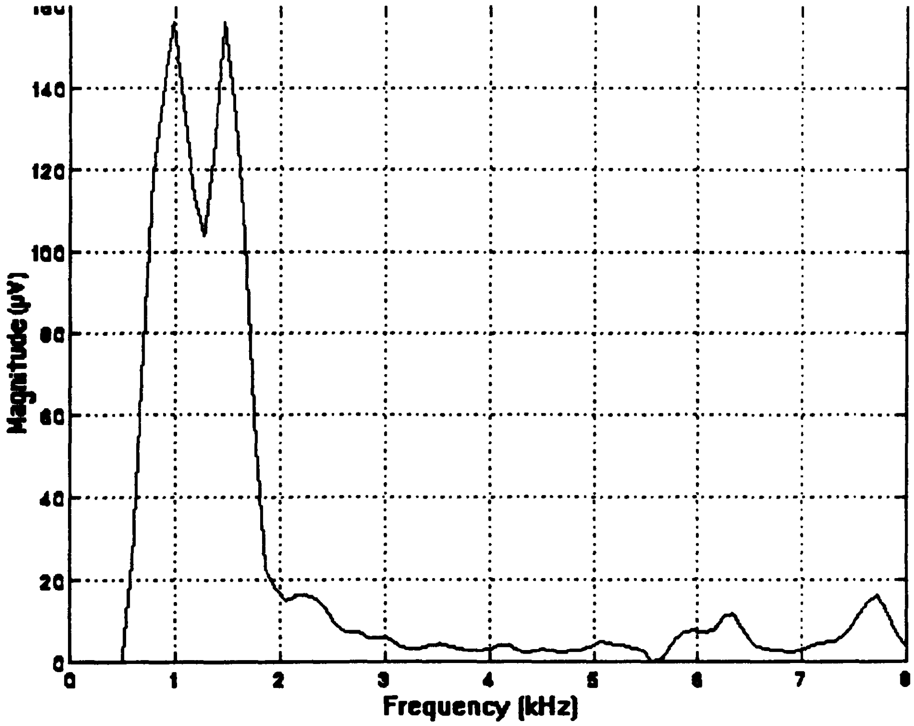


Observations performed several minutes apart give an indication of the difference in the spectrum with changing degree of subcooling. As long as the boiling remained in the onset stage, there was little difference in the peak frequencies over the temperature range expected in the core.

However, the boiling noise appears to have gone through other stages. If the power were increased at a constant temperature or if temperature increased at constant power, at a certain threshold, the boiling frequency spectrum pattern would change. The changes were not simply an increase in the magnitude of the peak or a shift in the peak frequency. The changes suggest different types of bubble being formed.

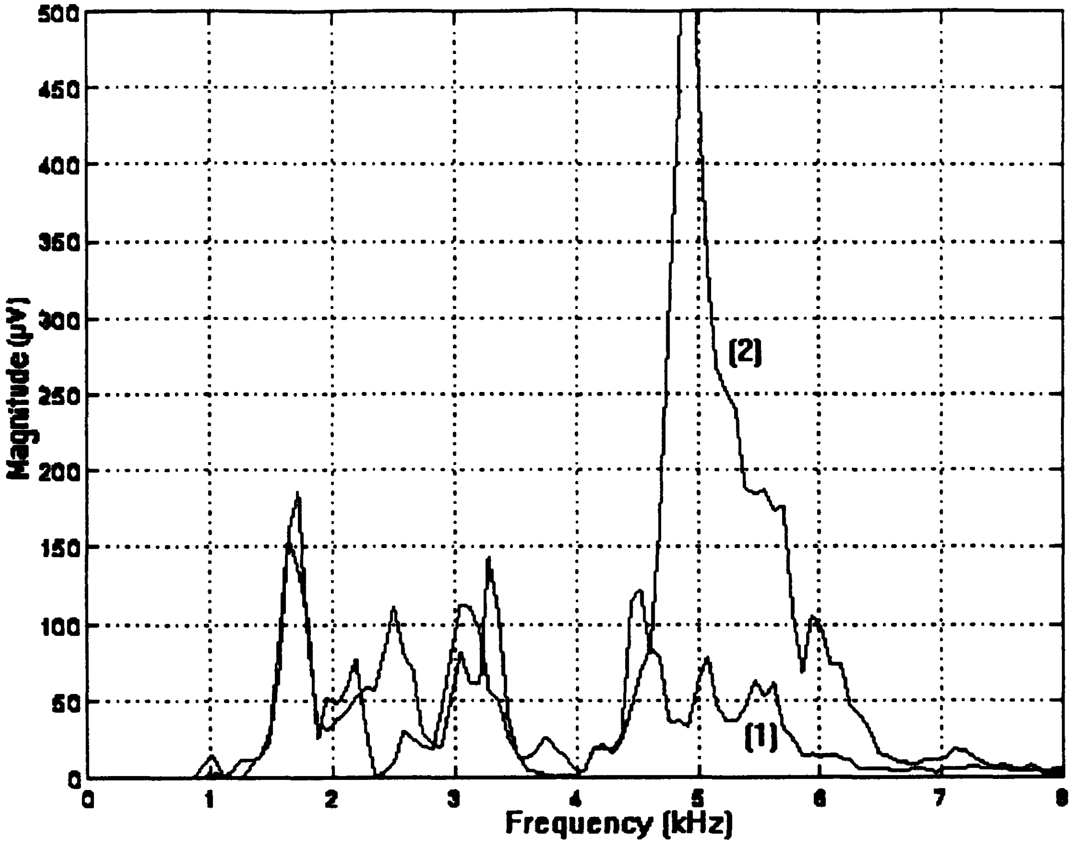
The second stage (or regime) of subcooled boiling initiates when an additional frequency peak begins to grow (**Figure 1.6**). This additional peak is at a lower frequency than the first. As power or temperature continue to rise this second peak grows in magnitude to levels higher than the initial peak. Since the peaks are closely spaced, if power continues to increase, the peaks become one.

Figure 1.6: Boiling at 53 deg C, 500 kW/m^2, no flow



As power is raised even higher a third peak begins to grow (**Figure 1.7**). This broad peak occurs at a much higher frequency than the other two. As power continues to rise, the magnitude of this peak dominates the first two. Observations were not made at power levels higher than this in order to avoid damage to the heater. Explanations for the onset of these peaks is beyond the scope of my research but further analysis may lead to interesting theories about the stages which occur before damage to a fuel element.

Figure 1.7: Boiling at 36 deg C, 700 (1) and 900 (2) kW/m², no flow



Chapter 2: Background Noise

Noise is any disturbance registered by the detector. Noise may affect the detector's ability to determine if boiling is occurring. These disturbances may lead to false alarms or prevent detection. Of particular interest is background noise. Background is any signal which is not the signal of interest. Any measurement taken will always have background. If there is too much background noise, the detector may be incapable of making a reliable estimation of the conditions. Eliminating or reducing noise is paramount to successful boiling detection. Therefore, understanding and predicting the background noise is the topic of this chapter.

Background noise can come from many sources which can be electrical or acoustic. Electric disturbances or noise may result from faulty wiring or poor equipment. This creates a cost tradeoff since equipment which produces the least electrical noise is often the most expensive. But if the electric noise is low relative to the amplitude of the desired signal, the additional clarity afforded by more expensive equipment may not be worth the cost. Once information is digitized, it is not susceptible to more noise.

Acoustic noises are the pressure disturbances received by the hydrophone which are then converted to an electric signal. The acoustic noise has a greater effect on the detection of boiling so it will be the focus of this section. Sources of acoustic noise include the coolant pumps, movement of the control blades, the flow of water, and the creation and destruction of bubbles.

The sources of noise, whether electric or acoustic, can be broken down into several categories. Background noise can be classified according to its duration and bandwidth.

2.1 Duration

Background noise may persist or may be transient. Persistent background noise is a signal which lasts longer than an arbitrarily set period of time. This period of time depends on the frequency of sampling and is relative to the components which make up the signal. For example, if pump noise levels were being measured once a day, changes in the level of noise which took hours to take place would produce fluctuations considered transient. But if spectra were being updated every second, a change in the level which took several hours to take place would appear persistent from one sample to the next.

Conditions in a nuclear reactor consist of transient and persistent components regardless of the time scale chosen. For a boiling detector, the required time scale is determined by the time in which shutdown must be initiated for safe shutdown after boiling has occurred. The detector must have updated itself at least once within this period. This requires updating spectra every fraction of a second. Background noise whose level changes unappreciably over this time interval will be considered persistent. A pump in steady state operation or an electrical bias on an amplifier are examples of persistent noise.

Transients are noise which occurs in a time period on the order of one sample interval. The sound produced when dropping the control blades is an example. It is interesting to note that pump noise may also be a transient during startup. Faulty connections also result in transient noise when the wires are bumped.

Attempts to reduce transient background noise are limited by the need for rapid detection of boiling. Attempts to eliminate persistent background noise are limited by the desire to avoid identifying slowly developing boiling as background noise. Each technique helps focus on boiling but does not eliminate all background noise from being presented to the detector.

2.1.1 Transients

The primary method of reducing the relative magnitude of transients is to increase the length of time of the buffer containing the data to be analyzed (sample time). The Fast Fourier transform (FFT discussed in Chapter 3) produces a frequency spectrum which has the average magnitude of the frequency component during the sample period. If the sample time is longer than the existence of the transient, the average value of the transient will be proportionally less. For example, if a 10V source were introduced for .1 second and then stopped and the sample time were 1 second, the magnitude of the source would appear to be 1V when averaged over the length of the entire signal.

Secondly, the larger the sample time the more computer memory is required to store and analyze the sample. For instance, at a sampling rate of 40 kHz, 320kBytes of memory are required to store double precision data for a one second sample time. Since high sampling rates are required in order to distinguish all frequencies in the boiling range and multiple buffers are required in the analysis of the spectrum, sampling times are on the order of 25 milliseconds. This is not enough time to reduce all but the shortest transients, so more provisions are required to provide adequate reduction of transients.

In addition to increasing the sampling time, sequential samples can be averaged together. By increasing the number of samples averaged, short transients can be eliminated. Averaging forty 25 millisecond samples together will result in an effective sampling time of 1 second.

By requiring the detector to identify boiling on more than one consecutive spectrum, false alarms caused by transients can be limited. This also limits carry-over between spectra averaged which can result in false alarms even for very short transients which are high in amplitude. But the user must be careful not to require too many consecutive boiling signals before the boiling alarm is set off, since the alarm may be unduly delayed.

The time required for initiation of shutdown of the reactor from the onset of boiling limits the time period of each sample. Hypothetically, if a sample is analyzed for boiling every 10 seconds, transients which lasted only 1 seconds would be reduced 2 orders of magnitude. But if boiling began at one second, and if, again hypothetically, in order for safe shutdown the reactor, scram must be initiated 4 seconds after the onset of boiling, the signal would be received 5 seconds too late. Therefore the sampling period must be shorter than the difference between the time from the onset of boiling to the time when scram is mandatory for safe shutdown. In fact it should be significantly shorter to allow for measurement error and additional safety margins.

2.1.2 Persistent Noise

Any source which remains constant is not difficult to remove. If indeed the source is constant, the background spectrum can be measured at one point in time then subtracted from subsequent spectra. Unfortunately, no sound source in the reactor is completely constant. Each source will have small variations with time and must both start and stop at some point. As a result, the background spectrum will change with time. Therefore, the spectrum subtracted from subsequent spectra must be able to adapt to the changing conditions in the core. Appropriate allowance for a moving background accomplishes this.

In order to avoid confusion and simplify the discussion, I will name various spectra used in the detection of boiling. The *source* spectrum is the spectrum entering the computer, the *background* spectrum is the moving average, and the *net* spectrum is the difference between the two.

When a source is received, it is subtracted from the moving background producing a net spectrum. The net spectrum is then analyzed to determine if boiling is occurring. The source is then averaged with the background to produce a new background spectrum and the procedure is repeated.

One problem with this method is the boiling will be quickly integrated into the background. This means that just a few sampling periods after boiling has begun, the net spectrum used to identify boiling will be close to zero even though boiling continues. This could result in missed detection.

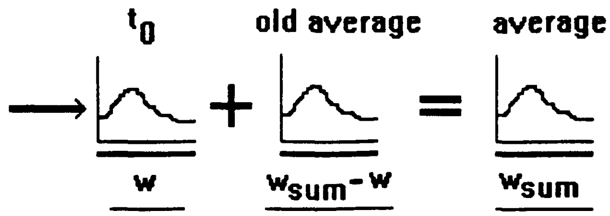


Figure 2.1 Infinite Moving Average for Background

The source and background spectra can be weighted to reduce this effect. Decreasing the weight given to the source spectrum will decrease the rate at which a steady state signal is incorporated into the background spectrum.

Consider, for example, a step signal introduced to a constant background. The weights for this example are assigned such that the ratio (r) of the spectrum weight (w) to the sum of the weights (w_{sum}) is 0.1. The first time step after the step signal has been introduced the source spectrum will be the background plus the step spectrum. The average will consist of 90% (1-r) of the background and 10% (r) of the source spectrum. When this average is subtracted from the source spectrum, the result is the net spectrum. The net spectrum will be 90% of the source spectrum. After the second time step with the same source, the average will again consist of 90% of the old average and 10% of the source, but remember the old average has 10% of the source. Therefore, the net spectrum will be 81% (1-(r+r*(1-r))) of the source. If the source remains constant after this point, the net spectrum, s, after n time steps will be:

$$s = 1 - \sum_{i=0}^{n} r(1 + r^{i}(1 - r)$$

After 3 time steps, the step boiling will still have approximately 73% of its magnitude. If the signal introduced contained boiling, the detector would only see 73 percent of the magnitude of the spectrum. After a period of time, the net spectrum presented for analysis may be too small to be recognized.

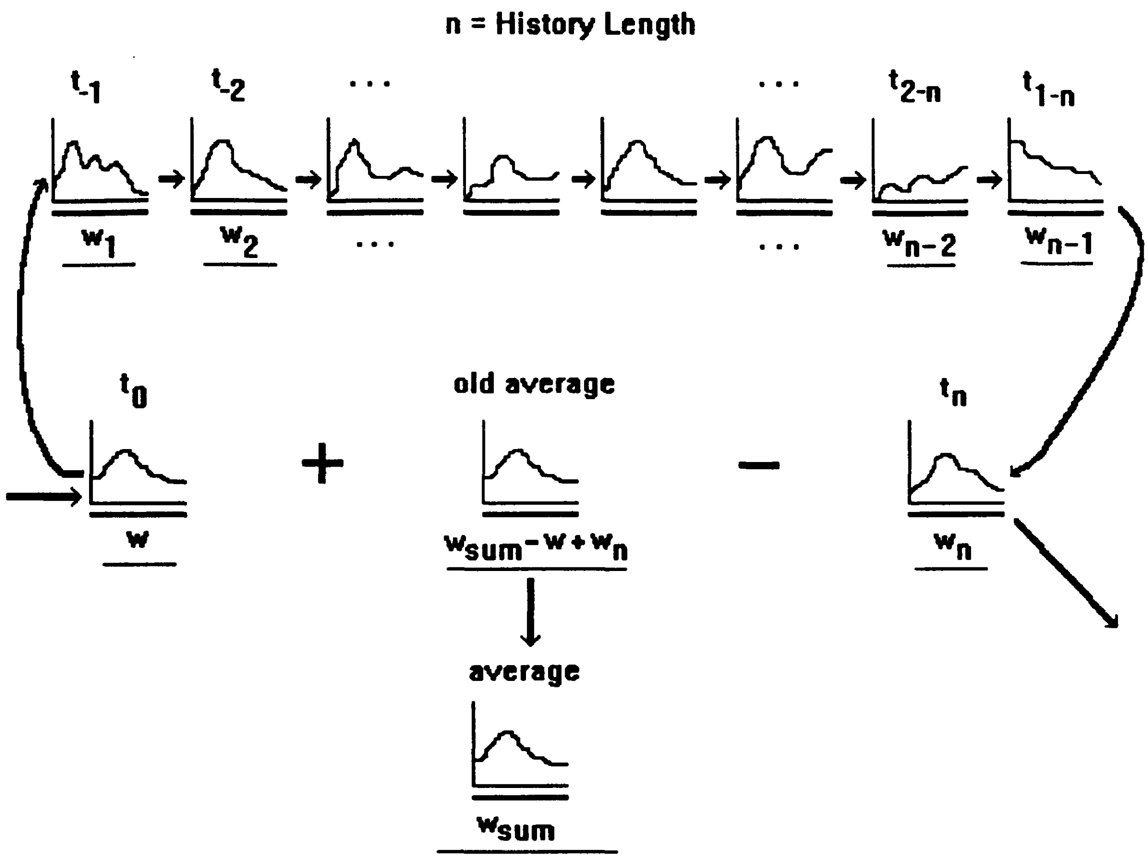


Figure 2.2 Diagram of Weighted Rolling Average

With an infinite moving average (see **Figure 2.1**), a small part of every spectrum will last forever, with its magnitude decreasing as time goes on. This presents a problem since large short lived signals persist for long time after they have stopped. This results in a negative net reading and may cover a spectrum which contains boiling. To alleviate this problem, old spectra should be subtracted from the background, thereby alleviating their influence. This is called a rolling average (see **Figure 2.2**). In order to keep a rolling average, a history of the spectra must be kept (the longer the history the more memory is required). Old samples are stored to the disk along with their weights so they can be removed later. Since the disk will be accessed with each new spectrum introduced, the computer must be able to store and retrieve information from disk quickly.

Additionally the weight to assign a source can be determined by the output of the detector. If the detector indicates boiling is occurring no weight is assigned to the spectrum. This means than the spectrum will not be integrated into the background spectrum and a boiling spectrum on the next sample will not have lost any intensity.

If boiling occurs for an extended period of time, the weights assigned to all the spectra in the history could potentially become zero. This would mean the detector would "forget" what any of the background looked like. To solve this, if the weight is zero (i.e. boiling is occurring), the spectrum is not written to the background history, after all it is not background and if the sum of all the weights goes below some threshold, new spectra are not written to the history unless their weight is higher (i.e. less representative of boiling) than the spectrum they are replacing. This procedure is an excellent way of adapting to changing background signals without missing signals which are boiling.

2.2 Bandwidth

Noise sources in the reactor will have different bandwidths. The energy from narrowband noise is concentrated about a few frequencies while broadband noise energy extends over a wide range of frequencies. (see **Figure 2.3**). The exact cuttoff between broad band and narrowband noise is relative. In this case, spectra with energy extending over a greater energy than a typical boiling spectrum are considered broadband (aproximately an increment of .5 kHz).

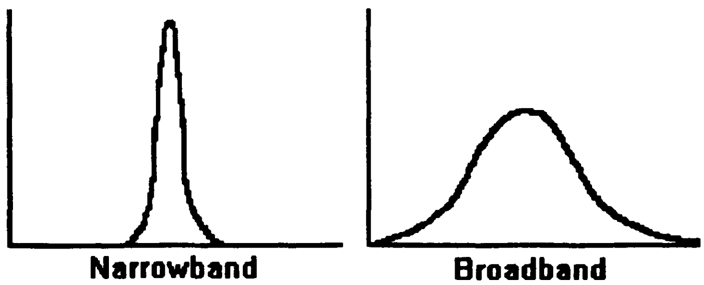


Figure 2.3 Narrowband and Broadband Noise Examples

Pump noise is an example of narrowband noise while turbulence in flowing water produces a broader frequency spectrum. By properly setting the frequency resolution, the influence of broadband or narrowband noise can be reduced. There are several tradeoffs which must be made in choosing a frequency resolution.

First, there is the tradeoff between reduction of broadband and reduction of narrowband noise. Since the energy is spread out over a range of frequencies, increasing resolution makes broadband noise less significant. Conversely, since the energy is focused about one frequency, decreasing resolution makes narrowband noise less significant. Therefore the resolution should be close to the bandwidth of a boiling spectrum.

Reducing the frequency resolution decreases the pattern recognition abilities of the detector. For example, if the boiling spectrum had two peaks which were very close in frequency and the resolution were set low the detector would only see one peak. This limits the detectors ability to distinguish boiling from non-boiling. It is important that the frequencies characteristic of boiling fall into more than one frequency bin. This helps eliminate the possibility of mistaking boiling with a narrowband of another source.

As mentioned in Chapter 1, the frequencies characteristic of boiling change with changing conditions in the reactor. With a low resolution detector, small changes in the associated frequencies with changing conditions will not result in a change in the distribution in the frequency bins.

As the frequency resolution is increased additional computer processing time is required. Depending on the speed of the computer running the detector, this can be a very limiting factor. Additionally increased resolution results in increased demands on the computers memory. This has been perhaps the most limiting factor on this project.

Chapter 3: Signal Analysis

Processing the signal is a very important step in reliable boiling detection. Separating and identifying the signal of interest is challenging due to the changing conditions in the core. This chapter describes the techniques used to remove extraneous data as the information passes through the detector.

3.1 Filters and Windows

The signal from the amplifier can be sent through an analog filter before being received by the computer. A analog low pass filter should be used for anti-aliasing (see Chapter 3.2 FFT). A passive filter can be easily and cheaply implemented without requiring an external power source. This choice reduces the likelihood of system failure. The low pass filter should have a have a corner frequency at approximately 8 kHz which is above the upper frequency expected of boiling (see **Figures 1.4 & 1.5**). The filter is not essential to successful boiling detection since, in general, noise with frequencies above half the sampling rate are uncommon. However, the filter does not require CPU time and it is easy to implement and maintain.

After being filtered, the signal is sent to an analog to digital (A/D) converter where the analog voltage is converted to digital value between of 0 and 2047. This information is put into a buffer where it can be downloaded to computer memory. The total time span of each buffer is the sampling rate, f_{samp}, multiplied by the number of samples, k. After downloading to computer memory, the samples can then be digitally filtered.

Windowing the digital data reduces spectral leakage caused by the finite signal length. Sharpening the window, increases frequency resolution while decreasing the information available. The choice of a window is application specific. A Blackman window was chosen for this application since it provides excellent frequency resolution.

The signal can then be sent through any of a variety of digital filters. One possibility is to send the signal through a digital equalizer. The frequency range of interest can then be emphasized. The most important frequencies are in the range of 1 to 3 kHz but other important boiling frequencies occur in the 3 kHz to 8 kHz range.

Signals below 500 Hz are also characteristic of boiling but are frequently masked by random background noise in this range. Therefore, this information is not useful for boiling detection. This range can be filtered out or simply not analyzed.

3.2 Fourier Transform

The signal is still in the time domain. It would be difficult to extract patterns from time domain data when a number of frequencies are involved. Differences in phase make even a simple single frequency wave look radically different to the neural network making training difficult to impossible. By performing a Fourier transform on the data, the signal can be converted to time averaged amplitudes in the frequency domain. The magnitude of the resulting complex spectral data is called the power spectrum. Since the power spectrum contains no phase dependence, it is easier to analyze.

In order to take a Fast Fourier Transform, the number of samples in the buffer holding the data must be a power of two. The formula for the Fast Fourier transform is[10]:

---

10 Lab Windows Advanced Analysis Library Reference Manual, Vol 2.3. National instruments, 1993.

$$Y[\omega] = \sum_{t=0}^{k-1} X[t] * \exp(-i\omega t/k \text{ , for } \omega = 0, 1*f_{samp}/k, 2*f_{samp}/k, \ldots, (k-1)*f_{samp}/k$$

The result of the Fast Fourier Transform is k discrete frequency bins of complex data. The k-length array of the magnitude of each frequency bin is the power spectrum. As mentioned earlier, the size of the buffer, k, also determines the frequency resolution.

The transformed data is aliased which means that the spectrum is mirrored about half the sampling frequency. Aliasing is a result of discrete data. Therefore, the sampling rate of the A/D converter must be set to at least twice the highest expected frequency of boiling (see **Figure 1.4 & 1.5**). Only the lower half of this data is of interest to boiling detection. Therefore, the size of the buffer containing the spectrum to be analyzed is k/2.

After the background has been subtracted from the source, the resulting spectrum is clipped. Any values which are below zero are unimportant for analysis since boiling does not absorb sound.

3.3 Logarithmic Scale

Some time was spent determining if the logarithmic scale should be used to modify the data before being sent to the network. The advantage of the logarithmic scale is that data the several orders of magnitude difference can be analyzed. The logarithmic scale would be beneficial for use with the detector if the boiling (or important characteristics of boiling occurred several orders of magnitude lower than the magnitude of the peak frequency. For example, if boiling had a peak at a frequency of 2 kHz with no other significant frequencies and severe boiling had a peak at 2 kHz and another peak three orders of magnitude smaller at 4 kHz, the second peak would be lost without the logarithmic scale. This could cause a failure to recognize the type of boiling occurring.

The net spectrum may be zero if, for example, a noise source in the background has been shut off and the background spectrum has not had time to adjust. The logarithm of zero is negative infinity which would throw off detection. Therefore a threshold must be set. Values less than the threshold are set equal to the threshold and then evaluated.

Although the logarithmic scale may be shown to be an improvement with further investigation, it is not essential for boiling detection. Any frequencies characteristic of boiling seem to give peaks of the same order of magnitude. Without the logarithmic scale, the peaks associated with boiling stand out since they are several orders of magnitude above the random background noise. This makes boiling easily detectable.

3.4 Normalization

The spectra are normalized with the maximum magnitude of the power spectrum being set to 1 and everything else being scaled. There are several advantages to normalized spectra. First, the distance and the number of obstructions between the actual boiling event and the hydrophone cannot be known. Therefore, boiling occurring under precisely the same conditions but at different locations would have different magnitudes. It is important that the computer recognize the spectral pattern associated with boiling not characteristic magnitudes. The detected patterns seem to be independent of the type of boiling or even whether or not boiling is occurring. Additionally, this means the network can be trained with the hydrophone one distance from the boiling and implemented a different distance away. Since testing in the core is unlikely until the final stages of development, this will enable the network to be trained in lab conditions.

In addition to different detector locations affecting the magnitude of the spectrum, the conditions in which boiling is occurring affect its magnitude as well as the frequency distribution. The spectra are more uniform and easier to recognize when normalized. This prevents missed signals at low or high extremes of boiling.

Normalizing the spectrum does have its disadvantages. First, if a noise significantly louder than boiling occurs at a different frequency while boiling is occurring, the boiling may be overwhelmed. Therefore, the spectrum should be normalized about the range in which boiling is expected and values above this range should be clipped.

Also, the magnitude of the spectrum contains important information about the boiling. For a binary boiling detector, where the operator is only concerned whether or not boiling is occurring this information is not important. But for a detector which makes an estimation of the power level, flow rate, and pressure, where the boiling is occurring, the magnitude of the spectrum is important information. Therefore, the normalizing value can be included as an input to the detector.

A random fluctuation may also be mistaken for boiling if the spectrum is normalized. To prevent this a threshold should be set. A spectrum whose maximum value is below the threshold should not be entered in the network for consideration as boiling.

3.5 Neural Network

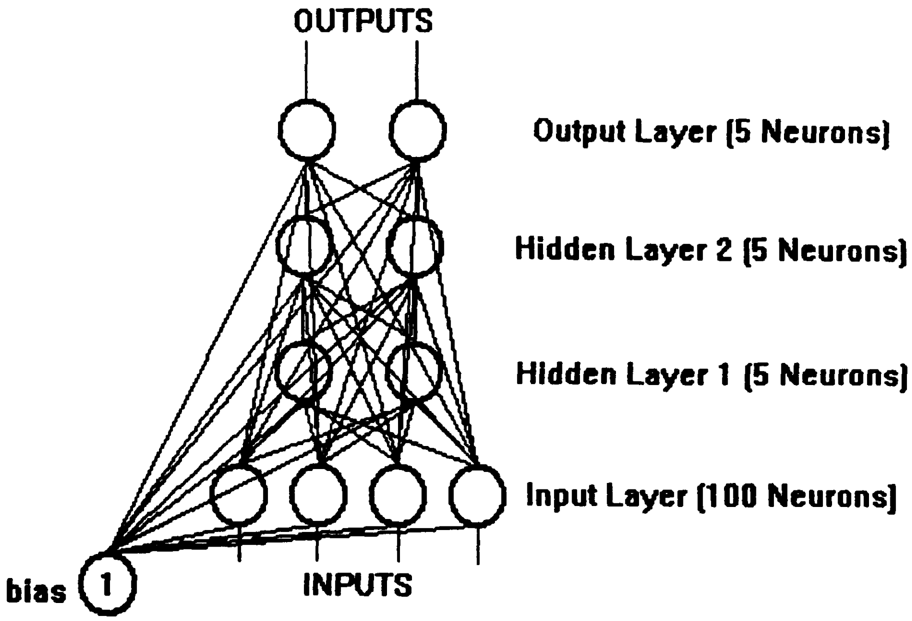


Figure 3.1 Diagram of Neural Network

Neural networks offer a means by which the computer can "learn" to recognize and classify information. They are modeled after the human brain. The advantages of neural networks make them excellent tools for many scientific applications. Further explanation on the theory and implementation of neural networks can be found in Neural Computing[11]. A summary on the theory and application of neural networks was taken from this text.

In an artificial neural network, the unit analogous to the biological neuron is referred to as a "processing" element. A processing element "PE" has many input paths (dendrites) and combines, usually by a simple summation, the values of these input paths. The result is an internal activity level for the PE. The combined is then modified by a transfer function. This transfer function can be a threshold function which only passes information if the combined activity level reaches a certain level, or it can be a continuous function of the of the combined input. The output value of the transfer function is generally passed directly to the output path of the processing element.

---

11 Neural Computing, NeuralWare, Inc. 1991. pp. NC3-NC7.

The neural network used by the boiling detector uses a sigmoid function with a straight line above a certain threshold.

The output path of a processing element can be connected to input paths of other processing elements through connection weights which correspond to the synaptic strength of neural connections. Since each corresponding connection has a corresponding weight, the signals on the input lines to a processing element are modified by these weights prior to being summed. Thus the summation function is a weighted summation. In itself, this simplified model of a neuron is not very interesting; the interesting results come from the way neurons are interconnected. A neural network consists of many processing elements joined together in the above manner. Processing elements are usually organized into groups called layers. A typical network consists of a sequence of layers with full or random connections between layers. There are typically two layers with connections to the outside world: An input buffer where data is presented to the network, and an output buffer which holds the response of the network to a given input.

The network used for the boiling detector has an input layer with 100 neurons, an output layer with 5 neurons, and two hidden layers of neurons with 5 neurons each.(see **Figure 3.2)**

There are two distinct phases in the operation of a network - Learning and Recall. In most networks [including the one used by the detector] these phases are distinct. Learning is the process of adapting or modifying the connection weights in response to stimuli being presented at the input buffer and optionally at the output buffer. A stimulus presented at the output buffer corresponds to a desired response to a given input; this desired response must be provided by a knowledgeable teacher. In such a case the learning is referred to as "supervised learning."

In the case of the boiling detector, the information provided to the input buffer is the normalized frequency spectrum. The output of the detector is the type of boiling (which is represented by four binary numbers) and potentially the conditions of boiling (i.e. temperature, flow, pressure, power, etc. which may be fed back for use as inputs).

Another type of learning is unsupervised learning which may not be applicable to the boiling detector. A third type of learning called reinforcement learning has the network learn by having the user indicate whether its output is "bad" or "good". This type of network should receive further investigation but was not applied to the current detector.

Whatever kind of learning is used an essential characteristic of any network is its learning rule which specifies how weights adapt in response to a learning example. Learning may require showing the network many examples thousands of times. The parameters governing a learning rule may change over time as the network progresses in its learning. The long term control of the learning parameters is referred to as its learning schedule.

The learning schedule was particularly important for the training of the network. If too many boiling spectra or non-boiling spectra were given in a row, the network learned down the wrong path and oscillated, unable to get over a threshold.

The recall stage is done at run time and refers to the way in which the network responds to various stimuli. At this point, the network does not continue to learn. If in the future, the operator decides that the network is not functioning adequately, it will have to be retrained.

3.5.1 Reasons For Use

The use of the neural network has many advantages over other methods of detection for this particular application. Boiling detection is a pattern recognition procedure, in which the incoming pattern may change with a variety of background conditions and changing boiling parameters. One advantage of the neural network is fault tolerance.

Whereas traditional computing systems are rendered useless by even a small amount of damage to memory, neural computing systems are fault tolerant. Fault tolerance refers to the fact that in most neural networks, if some processing elements are destroyed, impaired, or disabled, or their connections altered slightly, then the behavior of the network as a whole is only slightly degraded. As yet more processing elements are destroyed, the behavior of the network is degraded just a bit further. Performance suffers, but the system does not come to an abrupt halt. Neural computing systems are fault tolerant because information is not contained in one place, but is distributed throughout the system. This characteristic of fault tolerance or graceful degradation makes neural computing systems extremely well suited for applications where failure of control equipment means disaster: In nuclear power plant operation, missile guidance, space probe operation, and so on.

The boiling detector is an application which requires robust detection. Since the input to the detector is the power spectrum of the signal from the hydrophone, the spectrum will be complete, but it may be cluttered with background noise and it is possible that part of the boiling spectrum may be drowned out by a loud background noise.

Neural computing systems are adept at many pattern recognition tasks, more so than both traditional statistical and expert systems. The human ability to translate the symbols on this page into meaningful words and ideas is a form of pattern recognition. Pattern recognition tasks require the ability to match large amounts of input information simultaneously and then generate categorically generalized output. They also require a reasonable response to noisy or incomplete input. Neural computing systems possess these capabilities as well as the ability to build unique structures specific to a particular problem, so they are particularly useful in pattern recognition. The ability to select combinations of features pertinent to the problem gives them an edge over statistically based systems. The ability to deduce features on their own is an advantage over expert systems used in pattern classification.

Boiling noise comes in the form of distinct patterns. The overall magnitude is not as important as the particular spectrum shape for an indication of boiling. The shape changes depending on the conditions which eliminates the applicability of many statistical methods.

3.5.2 Training Procedure

To train the network, the spectra are taken and formatted in the same manner that they would be for use with the detector. The spectra are identified manually when taken. The boiling spectra are separated into different *regimes** and the non-boiling spectra are identified.

If the spectra are grouped together according to their type, the network may learn down one path for one type of boiling and then follow a separate path for non-boiling, causing it to oscillate about the right answer. To prevent this, the order of the spectra must be randomized before feeding to the information to the neural network.

*regime refers to a group of spectra with similar frequency spectra which are likely representative of a particular type of boiling

3.5.3 Boiling Training

As many *regimes* of boiling as could be identified had to be rigorously sampled. Four outputs from the detector were used to determine the regime of boiling. Each output is binary. The highest significant bit representing the most severe type of boiling. The lowest significant bit represented whether or not boiling was occurring. Therefore, while training, each spectrum was manually assigned a value of either 0, 1, 3, 7, or 15. Zero means no boiling, one the onset of boiling, three the next most severe boiling, and so on. Therefore, if binary boiling detection is desired, simply look at the lowest significant bit which is one for all cases of boiling and zero for non-boiling.

When training the network, each boiling regime had to be included. The more spectra that are taken, the better the network will be able to recognize boiling. Due to the number of spectra required, and the possible combinations of conditions, the most efficient way to gather the information is to let the computer run the experiments. The computer must be able to control the power to the heater, measure the temperature of the water, and gather the acoustic information (as well as monitor the pressure for a system in which the pressure is allowed to vary). Unfortunately, for this experiment the equipment was not available.

To prevent the network from being erroneous in the event that significant background noise occurs at the same time as boiling, combinations of boiling and background noise spectra had to be taken. This is the most challenging part of boiling training. It is difficult to anticipate all the likely combinations of background noise. Using the program Filechan (appendix C.1), signals could be modified and combined. Acoustic spectra could be taken in the reactor and combined with existing boiling signals to help anticipate background noise in the reactor. Also boiling signals could have been combined with randomly produced signals. But this does not ensure that all feasible combinations of boiling noise are covered. Remember, the value output by the network is no better than the accuracy of the values that were assigned by the trainer to the test spectra.

3.5.4 Non-Boiling Training

To avoid false alarms, the computer must be equally certain of what is not boiling. This seems a simple task, but is perhaps the most difficult part of training the network. The network can only categorize the boiling according to spectra it has seen. For spectra which the network has no experience, it attempts to fit them into a category. Miscategorization can result if the non-boiling spectrum is similar to a boiling spectrum which may lead to a false alarm. Therefore, the more defined the boundary between categories, the less chance of incorrect detection. For a network with two inputs, defining these boundaries is not difficult, but for a network with many inputs, sometimes hundreds, the boundaries are difficult to define (try to imagine a 100 dimensional boundary). Often the specific location of the boundary is uncertain even if its general location is known. For example, at what point do you draw the line between boiling and non-boiling.

To reduce the possibility of mistaken signaling of boiling, a huge number of random signals can be created, checked for similarity to boiling and fed to the network. But even this does not cover all possible combinations of background noise. A mentioned above, the rest is up to the network. Care must be taken to ensure that, as a result of the large number of non-boiling signals used in training, the network does not pick a certain characteristic common to all of the boiling signals, which is coincidental rather than a result of boiling, and mistakenly call everthing else non-boiling. This will lead to an ineffective detector.

Setting a threshold will reduce the number of false alarms is to set a threshold. This prevents a random pattern which may be similar to boiling but of lower magnitude from even being sent to the network where it may be mistaken for boiling. Only larger, more significant signals will be processed.

Chapter 4: The Boiling Detector

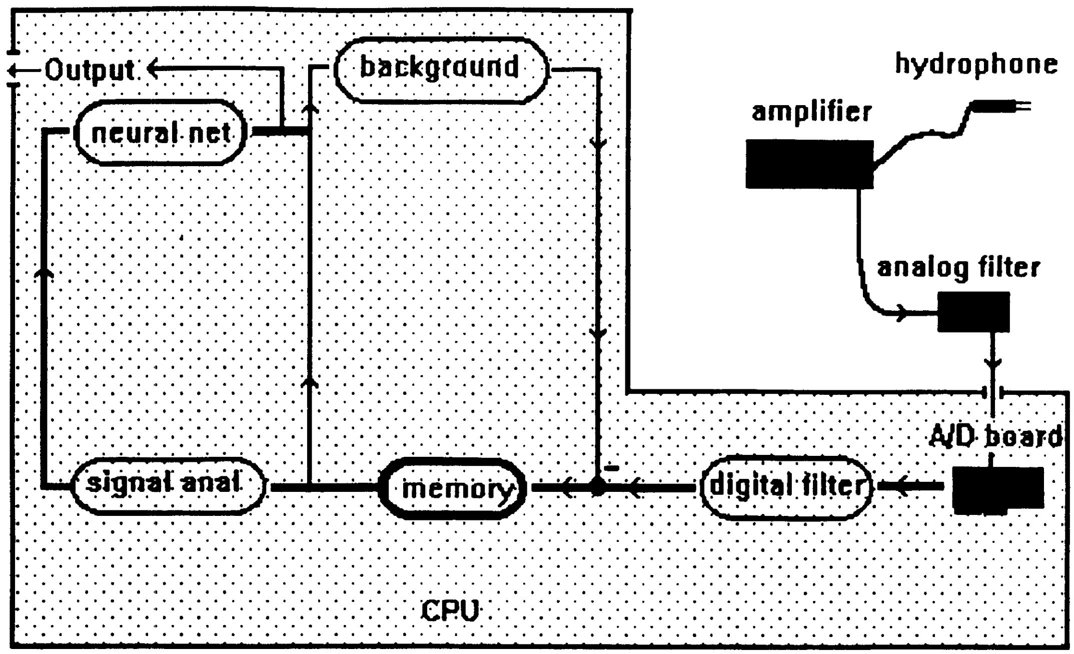


Figure 4.1: Diagram of Boiling Detector

Chapters 1-3 discuss the theory and concepts used in the boiling detector. This chapter discusses how the theory and concepts are implemented rather than the specifics of the programming (see Appendix B.4), but the steps the boiling detector program goes through are discussed in a more general conceptual manner. Also, the default conditions are listed along with explanations as to why they were chosen. Since these defaults can be changed by the user, the advantages and disadvantages of changes will be listed.

4.1 Definitions

In order to simplify the explanation, several terms are defined that will be used repeatedly in the discussion. A *threshold* is set to prevent small random spectra from being normalized and possibly be mistaken for boiling. The peak magnitude is determined. If this magnitude is below the threshold, the network is not consulted and its output is set to zero. The threshold is set to 2×10^{-5} Volts. Increasing the threshold decreases the likelihood of a false alarm but increases the likelihood of a missed alarm. The threshold can also be set by the detector if the background spectrum was integrated.

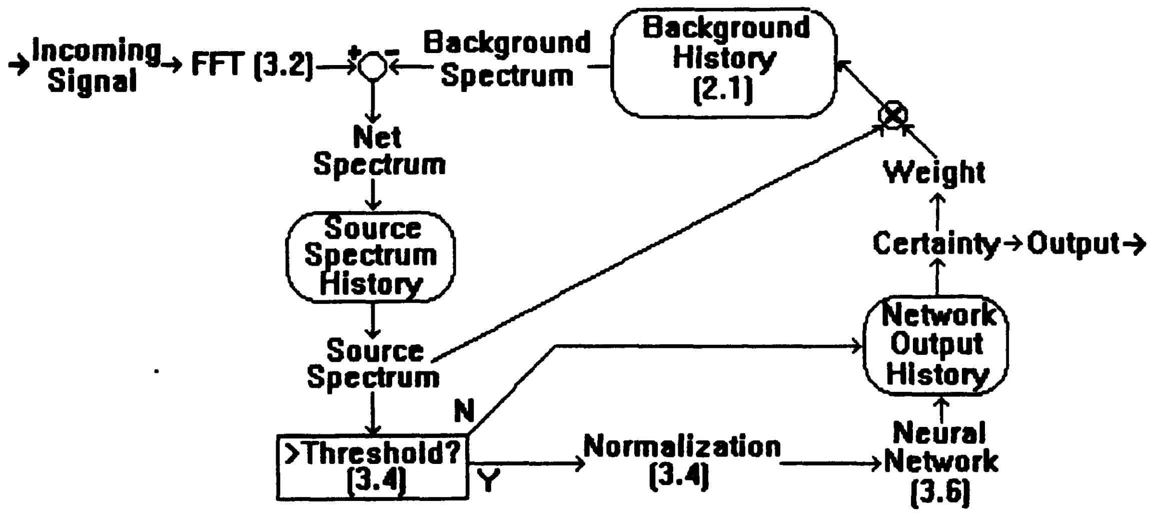


Figure 4.2 Diagram of Path of Data

Parley refers to a consultation of the network to make a boiling decision. Each time the network is parleyed, the determination of whether or not boiling is occurring is stored. After a predefined number of parleys, the determinations are averaged to give a *certainty* of boiling. Increasing the number of parleys, increases the accuracy of the boiling determination but also increases the time required to detect boiling. The default setting for the number of parleys is 50 since this results in a detection update rate of about twice per second with a sampling rate of 40 kHz. Increasing the required certainty of boiling, defined as the *boiling limit*, decreases the possibility of a false alarm but increases the likelihood of a missed alarm. The default setting for the boiling limit is 0.9, meaning that

90 percent of the output of the network must suggest boiling before the alarm is set. It is possible for the boiling limit to be greater than one since the output of the detector may be greater than one for more severe cases of boiling.

The *history* is the recent spectra which are stored in memory (see **Figure 2.2**). Each new spectrum replaces the oldest one in the history. Keeping a history enables a *rolling average* of the spectrum. The rolling average is the average of all the spectra in the history. The rolling average, however, can be updated with each new sample rather than waiting for the entire history to refill. Increasing the size of the history, results in a *smoother* rolling average which means the effects of random noise are reduced. If the spectra are *weighted* this must also be stored in the history. Assigning a weight to each spectrum allows the data to have more or less influence on the average depending on their assigned importance.

The new data has a history of the 5 most recent spectra. Since this setting is one quarter the number of parleys, this means at least four independent spectra are used to make a boiling determination. Reducing the smoothness causes the spectra to fluctuate more with random background noise which reduces the perceptibility of boiling. The advantage of reducing the ratio of the size of the history to the number of parleys is that large random noise does not skew the detection of boiling for long periods of time reducing the possibility of false alarms and missed spectra.

The size of the neural network output history determines the reaction time of the detector. As mentioned earlier, a larger history increases the number of parleys allowing better accuracy but delaying detection time.

The background history consists of 100 smoothed spectra. Increasing the history, makes a smoother spectrum but results in a greater delay in incorporating new background sources into the spectrum. Reducing the history increases the likelihood that boiling which slowly grows in magnitude will be incorporated into the background spectrum without being detected. With a background history of 50 spectra and a new spectrum history of 5 spectra, a new source will be completely incorporated in approximately half a minute.

4.2 Implementation

When the detector is started, the background and new spectrum history are initialized and a new background spectrum is taken. This spectrum is heavily weighted and will dominate the background until it is removed from the history. The initial background spectrum consists of the average of several independently smoothed spectra. If at any time the operator wishes to re-initialize the background spectrum, this process is repeated.

After this initialization takes place, the detector enters the normal operating mode. With each new sample the sample rolling average is updated. The background rolling average is then subtracted from the sample rolling average to get the net spectrum. The net spectrum is then normalized and parleyed if it is greater than the threshold.

Each time a single pass through the new sample history has been made, the sample rolling average is weighted according to the current certainty and incorporated into the background history. The weight given to the sample rolling average is determined by:

$$\text{weight} = 1 - \sqrt{\text{certainty}}$$

The *update limit* is the maximum certainty for which the sample rolling background will be incorporated into background rolling average. If the certainty is above this limit, the spectrum is likely boiling and should not be incorporated into the background. In general, the update limit should be lower than the boiling limit to prevent spectra which represent the onset of boiling from being incorporated.

Each pass through the neural network output history updates the output to the user. If boiling is detected before a pass is completed, the alarm is instantly turned on allowing faster response time. Therefore the minimum time to detection depends on the boiling limit setting. The lower the limit, the faster response time is possible.

4.3 New Ideas

In the process of investigating boiling, a few ideas were developed for methods which could be used to detect boiling. Unfortunately, due to time constraints, these techniques were not developed. Hopefully, whomever takes up this project in the future will investigate their applicability.

4.3.1 Pattern Transform

The first idea is a pattern transform. As mentioned in Chapter 1, the ideal bubble has a very specific pattern (**Figure 1.3**). Under low flow conditions near the onset of boiling, the frequency spectrums sampled seemed to resemble the frequency spectrum predicted frequency spectrum over 1000 Hz. No flow tests were not done in a narrow channel, so this may or may not be a valuable technique for detection in the reactor.

A transform can make a peak out of a pure pattern the same way that the Fourier transform creates a peak for a pure sine wave. An algorithm would have to be developed for the transformation of this spectrum. The result could then be plotted on a three dimensional graph with the x axis the peak frequency, the y axis the bandwidth, and the z axis, the magnitude of the spectrum.

Peaks on this graph would be excellent indicators of boiling. Additionally, the location of the peaks may yield additional information on flow rate, power, and pressure. Impure peaks (those which represent a range on the graph) could be eliminated from consideration.

4.3.2 Fourier Transform of the Power Spectrum

Taking the Fourier transform of the power spectrum enables easier identification of harmonics. Propeller noise is one example of a harmonic sound. The propeller operates at some fundamental frequency. There are harmonics of this fundamental frequency. For each blade there will be a peak evenly spaced between these harmonics. Between each harmonic there will be a number of peaks. By taking the Fourier transform of the sample which has already been transformed once, there will be three peaks, one representing the fundamental frequency, one representing the secondary frequency, and one representing the offset (since all values are greater than zero). From this information the operating speed and number of blades can be determined. It is important to note that high frequencies after the second transform represent low frequencies and low frequencies represent high frequencies. In order to accurately show harmonics, the sampling rate must be many times the fundamental frequency. If elimination of the noise is desired, the second transform can be digitally filtered, then an IFFT performed. This should limit the strength of the harmonic in the frequency spectrum.

This technique could be used for identification of pump noise to prevent false alarms. Also, since the ideal bubble is harmonic, this technique could potentially be used to identify boiling.

4.3.3 Coherence

Boiling detection by neutron noise has existed for years using the coherence function. With coherence, the similarity between signals from two detectors in the same channel in different axial locations can be compared to detect boiling.[12]

In order to get the coherence between two signals several steps must be taken. First obtain the *Cross-Power Spectral Density function (CPSD(f))*:

$$\mathrm{CPSD(f)} = \mathrm{X(f)}^* \times \mathrm{Y(f)}$$

where $\mathrm{X(f)}^*$ = complex conjugate of frequency spectrum of time domain data x(t); and
$\mathrm{Y(f)}$ = frequency spectrum of time domain data y(t).

The *coherence* ($\gamma(f)^2$) can then be defined as:

$$\gamma(\mathrm{f})^2 = \frac{\mathrm{CPSD(f)}^2}{\mathrm{APSD_a(f)xAPSD_b(f)}}$$

[12] Lubbesmeyer, D. "Experimental Reactor Noise - A Review On Noise-Analytic Measurements of Thermohydraulic Parameters in Operating BWR's and Their Interpretations", *Progress in Nuclear Energy*, Vol 14, No 1 pp-41-93, 1984.

where:

$$APSD_a(f) = X(f)^* \times X(f$$
$$APSD_b(f) = Y(f)^* \times Y(f)$$

Coherence can also be used for acoustic boiling detection. The two signals compared are the current spectrum from the core and a prerecorded boiling spectrum. Time dependence is not considered for acoustic boiling detection so phase information is not necessary. Therefore only the power spectrum of the two signals is compared. If the two frequency patterns (independent of magnitude) are identical, the coherence will be one. If the two signals are uncorrelated the coherence will be zero.

There are two problems with this approach. First, as conditions change, the frequency spectrum associated with boiling also changes. With coherence, one spectrum is chosen to be representative of boiling, and all incoming signals are compared to this spectrum. If the representative spectrum is a different regime of boiling than the acoustic signature received by the detector, boiling may be missed.

Also, coherence is a scalar value. One number may not accurately describe a boiling condition particularly if boiling may occur in a range of frequencies. But, if the spectrum is broken into several parts with each part having its own coherence, the result will be an array of coherence values which may be used to determine the type of boiling as well as more accurately predicting the onset of boiling. The parts should be broken up in a way which captures different frequency ranges in which boiling may appear. Each of these ranges should have a reference spectrum which is representative of boiling in that range.

The array of coherence can then be fed to a neural network for boiling output to the user or may be directly evaluated.

Using the coherence function, though, takes away from the power of the neural network since not all of the frequency information is being presented to the network. A correctly trained network will be capable of doing a better job than coherence at distinguishing boiling. But, correctly training the network can be more difficult, so coherence deserves further study as a possible means of detection.

4.3.4 Fluctuations in the Power Spectrum

Instead of a pure tonal, the power spectrum of boiling noise typically fluctuates in magnitude and frequency. By averaging the power spectrum, these fluctuations are limited and a single peak seems to persist. This makes boiling recognizable, but it also increases the likelihood of false alarms. If another sound which has a similar distribution is produced, the only distinguishing factor may be the way the boiling spectrum fluctuates with time.

4.3.5 Power and Temperature Prediction

Perhaps the most important possibility is the detectors ability to determine power, pressure, temperature, and flowrate at the point where boiling is occurring. Since each parameter affects the boiling spectrum, it should be possible to train the network to determine the conditions at the point boiling is occurring. This will involve more extensive experimentation. In order to ensure that the parameters have been tested within reasonable bounds, the computer should be set up to control the power, pressure and flowrate while taking boiling and temperature readings. Each of the parameters can be varied repeatedly to gather the necessary data.

If the detector is successful in not only determining whether or not boiling is occurring, but also finding the conditions where the boiling is occurring, it will become useful in environments where boiling occurs routinely. This would make the detector useful in commercial reactors with designs like the PWR or BWR. It is conceivable that the detector could be used to determine how close the reactor is to critical heat flux.

4.3.6 "Smart" Implementation

There are also a number of improvements which could be made to the current detector with little difficulty. One possible improvement is a "smart" setting system in which the thresholds and limits are automatically set depending on the conditions in the reactor.

These improvements are very specific to the application, however, and would have to be made by the user. For example, if the detector were given access to the reactor power information, these data could be used to set the boiling limit threshold high automatically since the likelihood of boiling is low. By setting the boiling limit high, the detector becomes less sensitive which makes false alarms less likely. As power is increased, the detector could automatically reset the boiling limit.

Access to reactor power information would also enable the computer to determine reactor period. If the reactor period were short, the detection response time could be reduced by making the network output history smaller.

In addition to changing limits and thresholds, the detector could change the neural network that the data is being sent to. Networks trained under different conditions can be more precise in a specific condition.

The computer can also compile the given information to make a better boiling decision. The core outlet temperature scram could be combined with the acoustic boiling detectors signal with an AND or OR gate, or some combination of the two. The outlet temperature and perhaps the inlet temperature as well could be used as an input to the network which would help it to determine the feasibility of a spectrum as well as being a direct input to the scram signal. Pump or other state information could be used to help the detector determine how quickly to update the background, what background to expect, whether or not to increase boiling sensitivity in the event of low flow, and what threshold to set for boiling detection.

An additional improvement would be to include other methods of determining whether or not boiling is occurring such as neutron noise detection, or the other methods mentioned earlier in this chapter. Neutron noise detection could supplement the acoustic boiling detector. In combination with the work of Jeff Hughes on an effective neutron flux measurement system, the neutron noise boiling detector may soon become an inexpensive option. The design of an integrated control system combining reactivity change measurements with acoustic and neutron noise boiling detectors would be highly beneficial and with more reseach may be feasible.

Conclusion

The boiling detector is an improvement on previous boiling detectors in a number of ways. Those unfamiliar or uncomfortable with the use of neural networks will still find the signal separation techniques valuable for other types of acoustic boiling detectors. The neural network adds the versatility and accuracy neccessary for use in a reactor control system.

Future Work

An experiment should be set up which will allow computer control of the parameters affecting boiling noise, such as power, flow rate, and pressure. The computer can then run an experiment varying and storing each parameter individually along with the frequency spectrum to determine the exact effect on the boiling noise. This information could then be used to thoroughly train the network to detect boiling and could eventually be used to train the network to make an effective parameter predictor. Also the effects of varying the postion of the hydrophone relative to the location of the boiling and flow at various velocities should be examined before testing in the core.

In any event, the research and testing is far from complete. The detector has been successfully tested in an environment which is not the same as the conditions that will be seen in the reactor core. The testing was done in a pool type boiling situation and the reactor core has high flow rates past plates (see **Figure A.3**). Additionally, even the testing of the detector outside the core was by no means exhaustive. The results of current studies are promising but perhaps a year or more work is left before the detector will be ready for use in the core.

Appendix A: Experimental Hardware

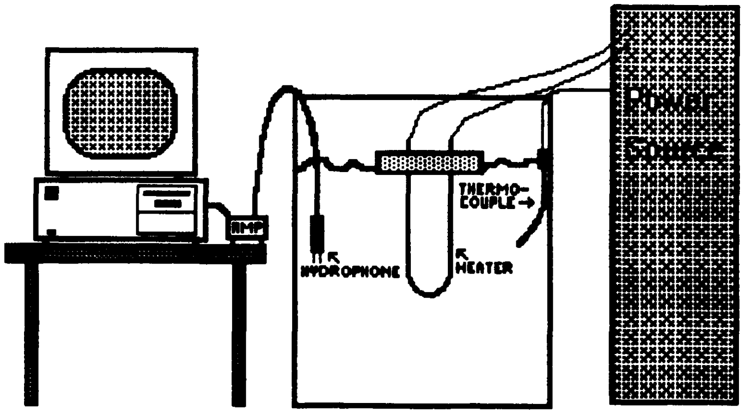


Figure A.1: Diagram of Experimental Setup

A.1 Hydrophone

The hydrophone converts pressure into voltage. The pressure fluctuations resulting from sounds traveling through the water cause voltage fluctuations to be produced. These fluctuations can then be amplified and sent to the computer.

For reliable boiling detection, there are several requirements of the hydrophone. First it must be robust since it will receive high levels of ionizing radiation throughout its life in the core even though it will be located many feet from the core. The hydrophone must not be degraded by the high temperatures in the reactor as well. For the MITR, temperatures in the vicinity of the detector will routinely be approximately 70 degrees C. Since MITR is a pool type reactor, pressures are close to atmospheric and variations are not a major concern. Also, the hydrophone must have good frequency response characteristics in the range boiling is expected to occur (between .5 and 8 kHz).

The signal from the hydrophone is sent through an amplifier. Like the hydrophone, the amplifier must have good frequency response characteristics in the range of boiling. Also, it should have a means of functioning if a power outage occurs.

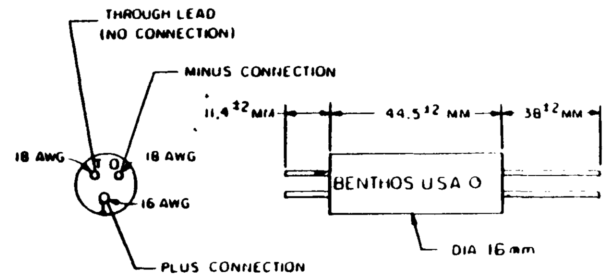
The hydrophone was recieved free of charge from Benthos, Inc.

Benthos, Inc.
Edgerton Drive
North Falmouth, MA 02556 USA
1-800-446-1222
(508)563-1000

Figure A.2: Diagram of Hydrophone

HYDROPHONE CARTRIDGES

	AQ-1	AQ-16
ACOUSTIC SENSITIVITY DBV RE 1 µPA ±1 DB	202.5 DB	197 DB
FREQUENCY RESPONSE ±1.5 DB	1 HZ-10 KHZ	1 HZ-10 KHZ
DIRECTIVITY ±1 DB ON AXIS	10 KHZ	10 KHZ
SENSITIVITY VS. DEPTH LOSS AT 1732 METERS	0.5 DB	0.8 DB
DEPTH-METERS RATED/DESTRUCT	1732/13860	1732/13860
SIZE-CM DIA./LENGTH	1.59/4.45	1.59/4.45
WEIGHT-GRAMS AIR WATER	17.3/8.5	17.3/8.5
CAPACITY-PFD ±25%	14500	3500

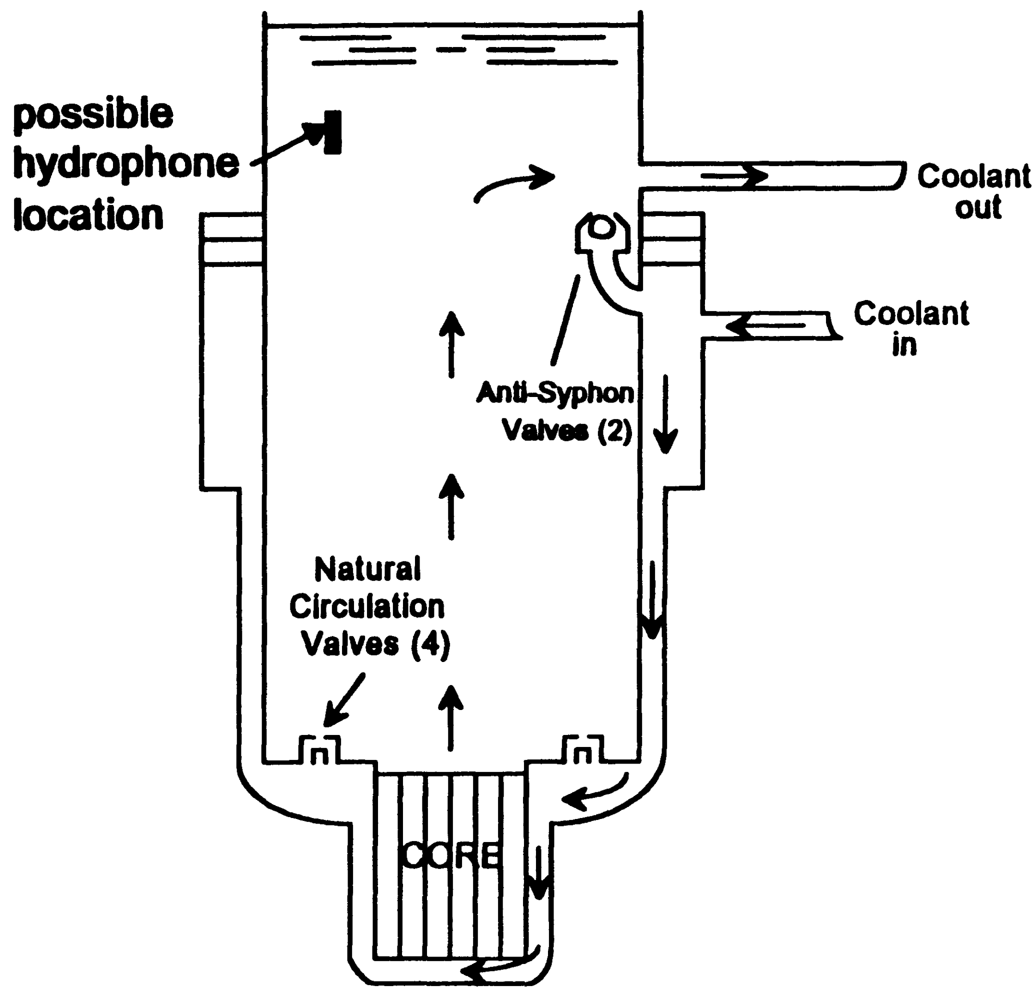


PHONE #(508) 563-1000 / 1-800-446-1222

BENTHOS, Inc.
Edgerton Drive
North Falmouth MA 02556 USA

.010
1/64
1°
HYDROPHONE SPECIFICATIONS
AQUADYNE AQ-1, AQ 16
.010R
27 JULY 87
FULL
B-400-212
A

Figure A.3: Possible Hydrophone Location



A.2 The Heating Element

The type of heating element used to create boiling is important. The surface on which the bubbles are created affects the sound. Further studies will have to be done to research the effects of rough and smooth surfaces since MITR has finned heating elements and narrow channels.

The cost of the heating element was approximately $200.

In order to produce the heat flux required for 40-70 deg C subcooled boiling, a special heater had to be ordered. This heater is capable of heat flux over 1000 kW/m^2 .

$$\text{Area}_{\text{Heated}} = \pi \times \text{Length}_{\text{Heated}} \times \text{Diameter}$$

$\text{Length}_{\text{Heated}}$ = 20.875 in = 530 mm
Diameter = .315 in = 8.00 mm
$\text{Area}_{\text{Heated}}$ = 20.658 sq. in = .01333 m^2

$$\text{Power}_{\text{Max}} = V^2/$$

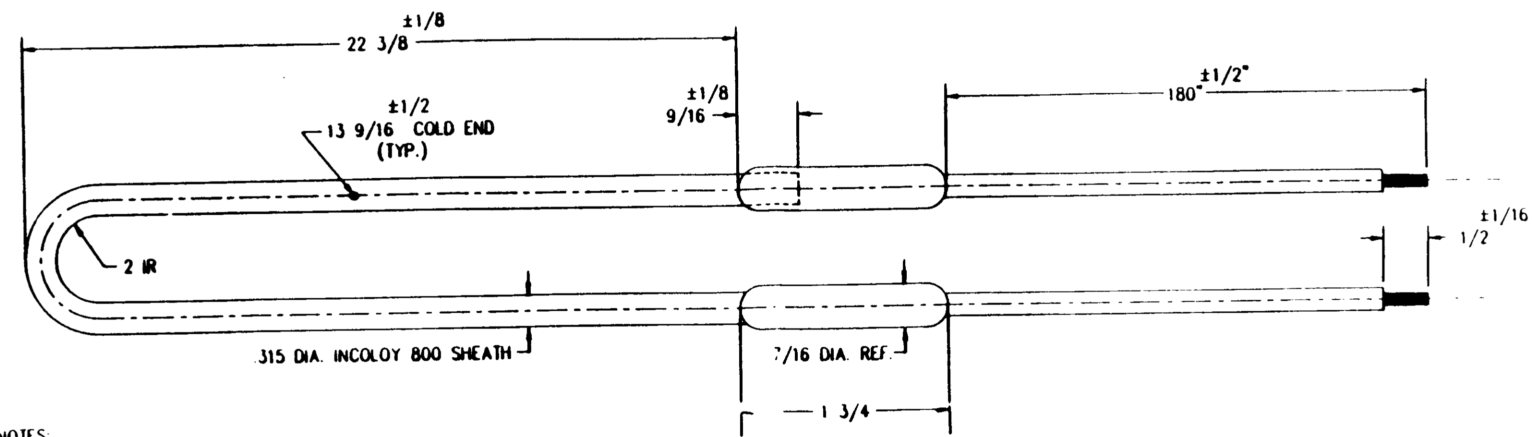
V = 480 Volts
R = 16 Ohms
$\text{Power}_{\text{Max}}$ = 14.4 kW

$$\text{HeatFlux}_{\text{Max}} = \text{Power/Area}$$

Heat Flux_{Max} = 1080 kW/m^2

Figure A.4: Diagram of Heater

REV	DESCRIPTION	DATE	DWN	CHKR	ENGR	APVD
1	RELEASED TO PRODUCTION	4/27/93	CTW	CTW	CTW	CTW



NOTES:

1. ***SAJ*** SPECIAL APPLICATION JOB--HIGH SHEATH AND WIRE WATT DENSITY. CUSTOMER ASSUMES RESPONSIBILITY FOR ELEMENT LIFE IN APPLICATION. NO WARRANTY IMPLIED.
2. STAMP SHEATH WITH INFORMATION SHOWN IN PARENTHESES: (CHROMALOX), PART NUMBER (_), VOLTS (_V), WATTS (_W), DATE CODE (_), AND MANUFACTURING LOCATION CODE(_).
3. LEADWIRES: #12 GA. STRANDED(650 STRANDS OF #30 TINNED COPPER WIRE) SINGLE CONDUCTOR APPLIANCE WIRE, RUBBER INSULATION (NEOPRENE RUBBER BASE) BLACK IN COLOR, COMPOUNDED FOR 90° C MAX. TEMP. (.256 ±.005 O.D.)
4. BLACK MOLDED SYNTHETIC POLYMER TERMINAL CAP, HEAT AND MOISTURE RESISTANT, DUROMETER 65-75, VULCANIZED TO HEATER SHEATH ASSEMBLY AND INSULATION.
5. ALL EXPOSED RUBBER SURFACES TO BE ODOR AND TASTE FREE PER E.L. WIEGAND SPEC. NO. 223-075047-000.
6. MOLDED TERMINALS TO BE IN LINE WITHIN 1/4".

PURCHASED FROM:
LEO PELKUS INC (617) 235 8040
170 WORCESTER ST. WELLESLEY HILLS MASS 02181

PART NUMBER	VOLTS	WATTS	REMARKS
087-876126-001	480	14400	***SAJ*** SEE NOTE 1

UNLESS OTHERWISE SPECIFIED, DIMENSIONS ARE IN INCHES. TOLERANCES ARE FRACTIONS ±1/32 DECIMALS XX ±.03 XXX ±.005 ANGLES ±2°

MATERIALS

FINISH

THIS DRAWING AND THE SUBJECT MATTER SHOWN HEREON IS CONSIDERED CONFIDENTIAL AND MAY BE PROPRIETARY. UNAUTHORIZED DISCLOSURE TO OTHER PARTIES IS EXPRESSLY FORBIDDEN.

CONTRACT NO

USED ON

APPROVALS		DATE
DWN	CTW	04-27-93
CHKR	XXX	XX-XX-XX
ENGR	XXX	XX-XX-XX
APVD	XXX	XX-XX-XX
DWO 740052		PC# XXXXXX
CAD# 87612601		

Chromalox® WIEGAND INDUSTRIAL DIV
600 RIDGELY ROAD
MURFREESBORO, TN 37129
EMERSON

HEATER- RUBBERMOLD

SIZE	CODE IDENT	DRAWING NO	REV
B	07131	087-876126-000	01

SCALE NONE | DO NOT SCALE DRAWING | SHEET 01 OF 01

A.3 Data Acquisition Board

The data aquisition board is the means by which the information can be entered into the computer for analysis. The board samples the analog signal and converts it to digital data. A multitude of A/D boards are on the market, I chose the Data Translation DT2831 because the Naval Academy was able to loan one to me (ie. it was free). The software I developed can be easily linked to other data aquisuition boards with a little programming.

The cost of the Data Translation DT2831 is approximately $1000 dollars. It provides the following featues[13]:

- Software Configurable settings:
  - Single-ended or differential input
  - Input gain (1,2,4,8)
  - Input and output range and coding
  - Interrupt level
  - DMA channel
  - DIO direction
  - Calibration of A/D and D/A converters
- A/D subsystem with up to 16 input channels provides 12 or 16 bit resolution, and a 512-entry channel gain list
- D/A subsystem with 2 independent D/A converters provides 12-bit or 16-bit resolution
- Eight lines of digital I/O
- Am9513A System Timing Controller provides three counter/ timers to automatically initiate A/D, D/A, or simultaneous A/D and D/A conversions, and two counter/timers for external uses such as event counting and frequency measurement
- Interupts to host processor when conversion is complete or error occurs
- Programmed I/O (PIO) or Direct Memory Access (DMA) data transfer modes
- Variable sampling rate (0-50 kHz)

Data Translation, Inc.
100 Locke Drive
Marlboro, MA 01752-1192
(508) 481-3700

---

13 DT2831 Hardware User Manual, Data Translation, 1991.

A.4 CPU

The computer used in the experiment was a Gateway 2000 4DX2-66E. In order to run these programs, the computer must have a math co-processor and/or be 486 or above. Fast computing times are essential due to the sampling rate and the number of calculations that must be performed on the incoming signal. The cost of the computer is approximately $2500.

Gateway
610 Gateway Drive
North Sioux City
South Dakota 57049

(605) 232-2000

Appendix B: User's manuals

In order to allow future use of the detector, User's manuals are provided for the programs produced in this work. In order to train the detector each of these programs should be run. They are designed to be user and programmer friendly. Watch out, however, because there are still bugs.

The first step in training the network is to run **Sampler**. This program is used to store data. it includes and option to take a background signal and subtract it from future inputs. This is useful in creating boiling spectrums for further analysis.

The next step is to run **Filechan**. This program takes the data gathered by Sampler and puts it in a format which can be recognized by the neural network for training. Also the data can be sent to Matlab for printing of further analysis.

After this, **DTRC94** should be executed. This is the neural network training file. Finally, **Boildet** should be run. This is the actual detector.

All of the programs were written using Borland C++ and labWindows libraries. The source code can be obtained through Rachel Morton or the MITR-III design group.

Figure B.1.1 Sampler Main Screen

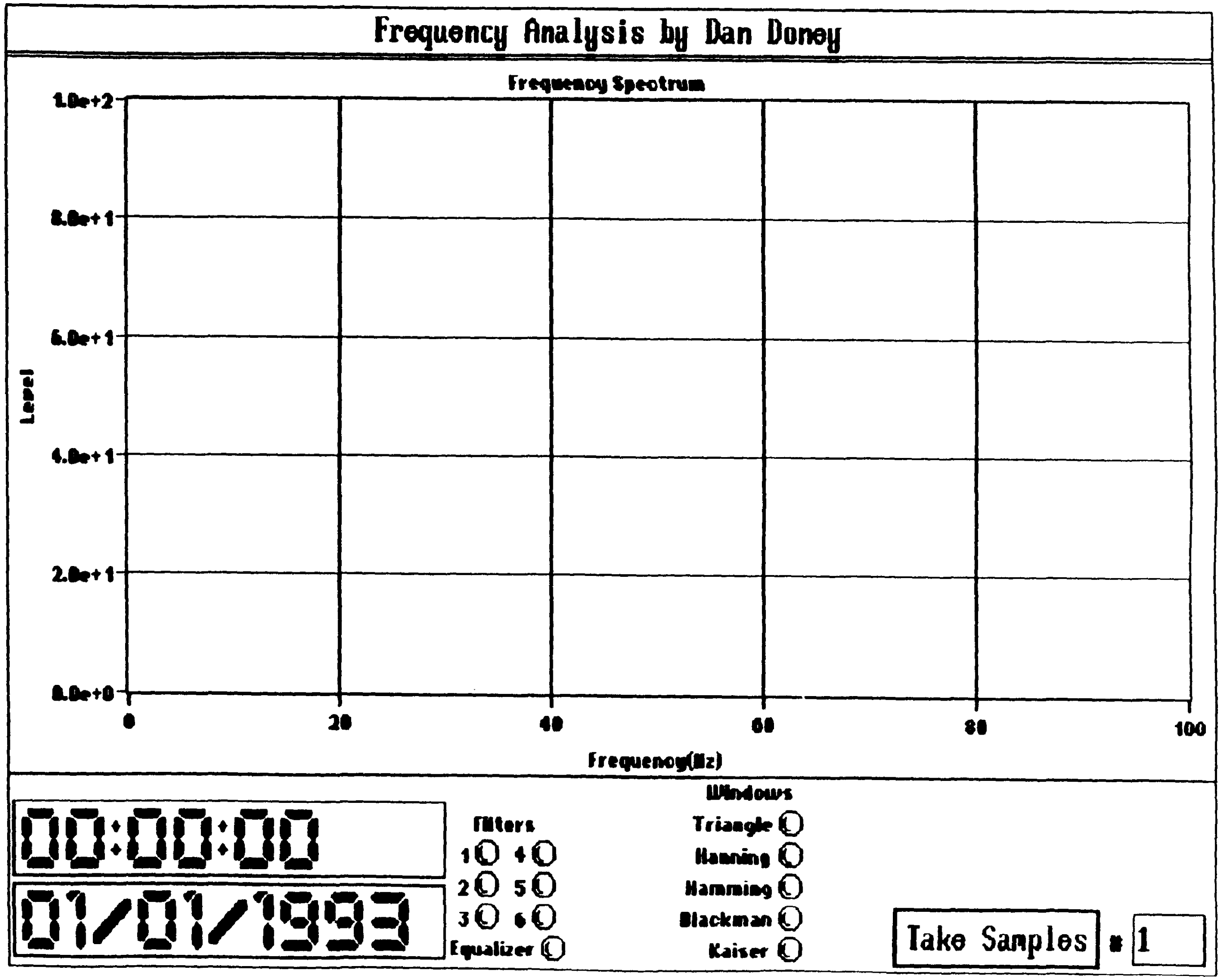
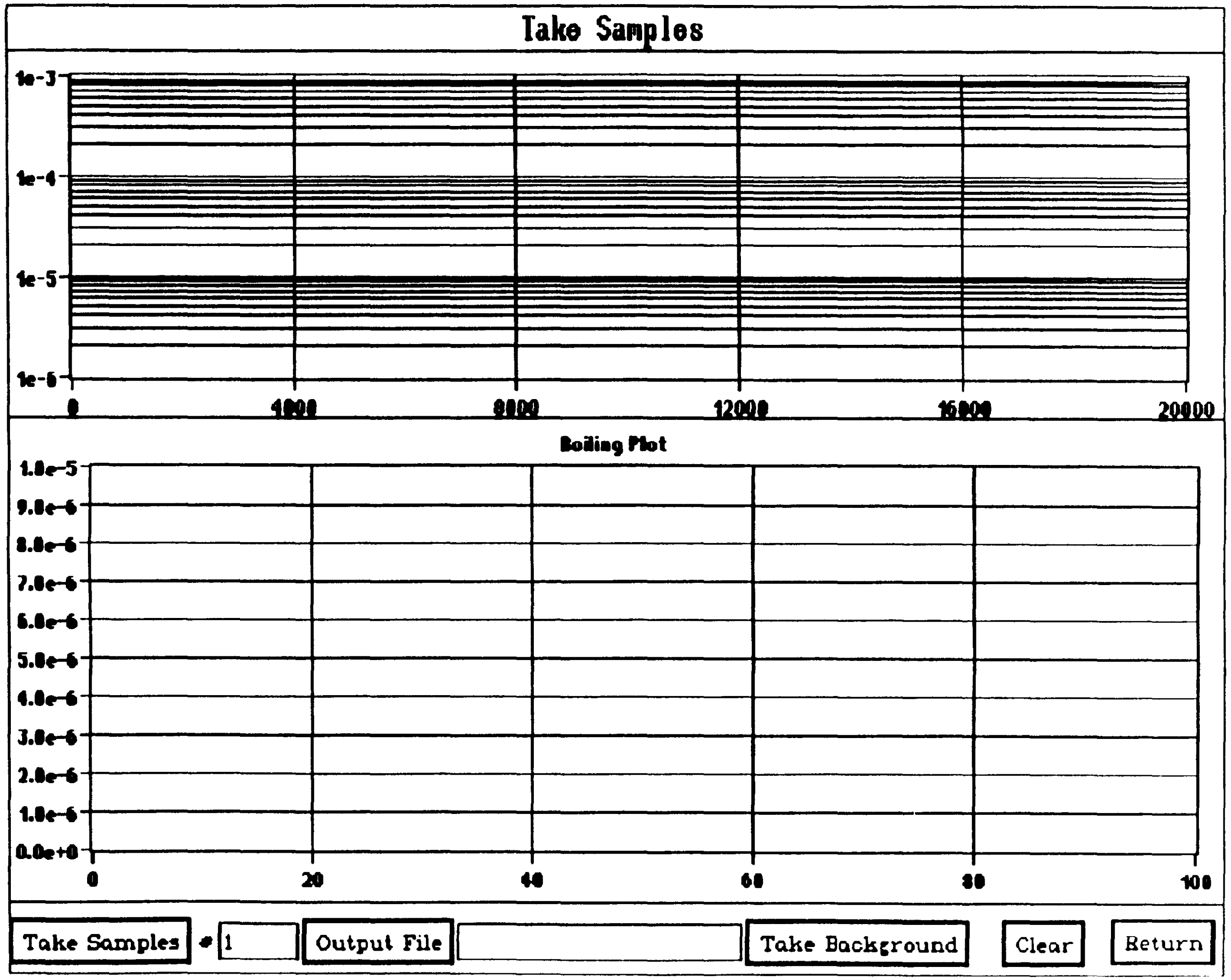


Figure B.1.2 Create Boiling Screen



B.1 Sampler

This program was written to take in samples for analysis. It can be used to create boiling and non-boiling samples for training the neural network.

B.1.1 Startup

To start this program you must have a DT2831 board installed. From the DOS prompt type Sampler. The user is entered into the Main Screen.

B.1.2 Menu Options

Axes Menu: Opens Change Axis screen. See B.4.3.1

Equalizer Menu: Opens Create Equalizer screen. See B.4.3.2.2

Filters Menu: Opens Create Filters screen. See B.4.3.2.1

Windows Menu: Allows the user to toggle the state of the digital windows.

File Menu: Changes output file.

Quit Menu: Exits program.

Create Boiling Menu: Opens create boiling screen.

B.1.3 Create Boiling Screen (see Figure B.1.2)

Take Samples Push Button (hot): Pressing this button initiates a sample taking burst. The number of samples taken in the burst is indicated in the # indicator. The user will be prompted for information about the first sample taken. The rest of the samples in the burst will be assigned the same values. The plots of the spectra taken are plotted on the graphs. Remember, the spectra that are saved to disk are the source minus the background.

Indicator (normal): The user can change the number of samples taken in a burst by changing this number.

Output File Push Button & Indicator (hot): Pressing this button allows the user to change the file the data is sent to. The output file is a binary file in the format:

Int	Length of Spectrum;
Float	Sampling Rate;
Array of Doubles	Spectrum;
Int	Certainty;
Int	Temperature.

Take Background Push Button (hot): Resets the background signal.

Clear Push Button (hot): Erases all frequency plots in both graphs.

<Untitled> graph (indicator): Plots magnitude of frequency spectrum before the background is subtracted.

Boiling Plot graph (indicator): Plots magnitude of frequency spectrum after the background has been subtracted.

Return Push Button (hot): Returns user to main screen.

B.1.4 Main Screen (see Figure B.1.1)

Time Indicator (indicator): Shows current time.

Date Indicator (indicator): Shows current date.

Filters LEDs (indicator): Indicates the state of digital filter 1-6 (red -on; clear -off)

Equalizer LED (indicator): Indicates the state of digital equalizer (red -on; clear -off)

Windows LED (indicator): Indicates the state of digital windows (red -on; clear -off)

Take Samples Push Button (hot): Initiates sample taking burst. The number of samples in the burst can be changed by the # indicator. The samples sent to disk are source samples which means no background has been subtracted. The ouput file is binary and the samples are int the format:

Int	Length of Spectrum
Float	Sampling Rate
Array of Doubles	Spectrum

indicator (normal): Indicates number of samples in a burst.

Frequency Spectrum Graph (indicator): Plots current frequency spectrum.

Figure B.2.1: Filechan Main Screen

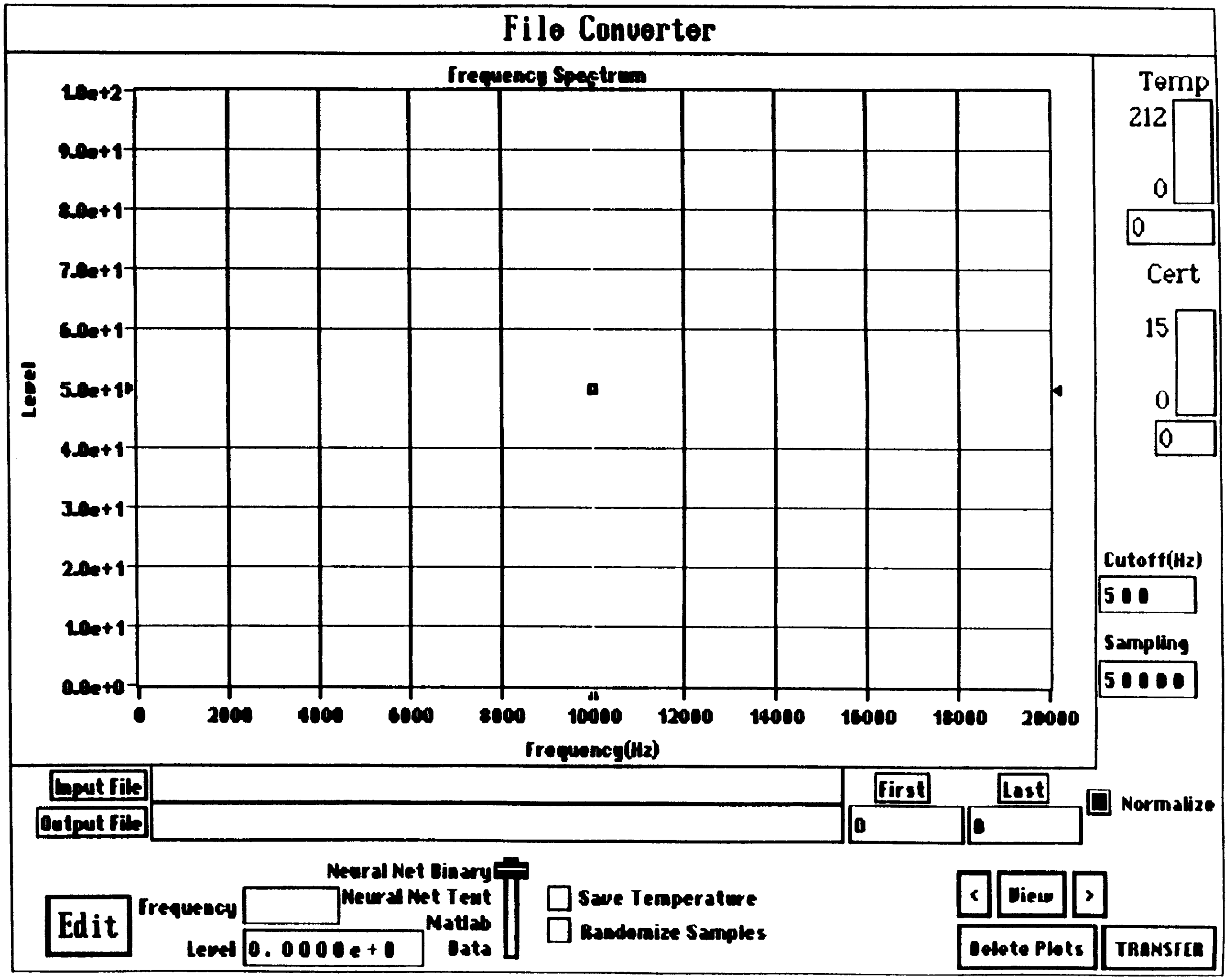
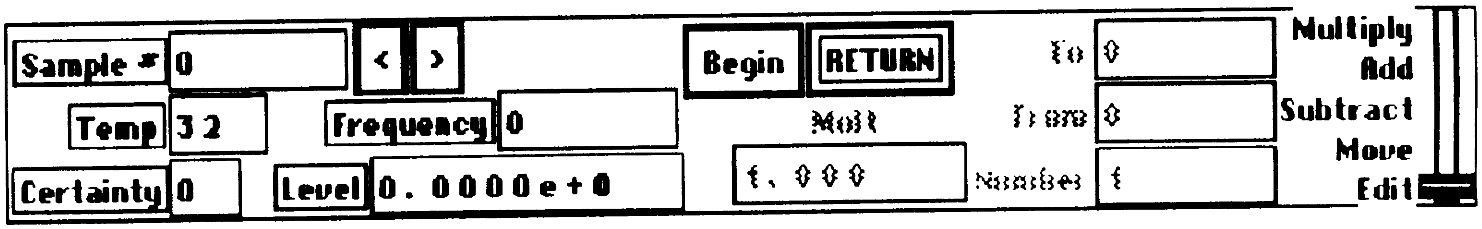


Figure B.2.2: Filechan Edit Screen



B.2 Filechan

Filechan was written to take a file of experimental data and convert it to a file which could be used to print the data or entered into a neural network. The user can modify the data or combine different signals which allows better training of the network.

B.2.1 Startup

On start-up the user is prompted to enter an input file. The cancel option os not allowed. The input file must have a .b9 extension and must already exist. If no *.b9 files exist, the user will not be able to proceed. After entering an input file the user is prompted to enter an output file. The output file is the file data is to be transferred to. These files can have any extension and may be a new file, but the type of the file is a **Nueral Network Binary** by default. If the user wishes to change this type it must be done after startup is completed. After entering an output file, the program runs normally.

B.2.2 Menu Options

Input File Menu (hot): This allows the user to enter a new input file. The input file must always end with a .b9 extension. These files are stored frequency spectra. Choosing cancel keeps the same input file. The chosen file must already exist. After the file is chosen, it will be displayed in the *Input File* indicator.

Output File Menu (hot): This allows the user to enter a new output file. If the output file chosen already exists, the user will be asked if he/she wishes to writer over the file. If yes the old file will be erased. If no, the file will be appended. After selection, the output file will be listed in the *Output File* indicator.

Quit Menu (hot): Terminates the program.

B.2.3 Main Screen (see Figure B.2.1)

Frequency Spectrum Graph (hot): This graph is used to plot the frequency spectrum data stored on disk. Clicking on the graph will cause the cursor to snap to the nearest graph. The frequecy and the level of the cursor will be displayed in the *Frequency* and *Level* boxes located below the graph.

Temp Scale (indicator): This scale gives the temperature in Fahrenheit of the most recent spectrum plotted. This is the temperature which was recorded when the sample was taken.

Cert Scale (indicator): This scale gives the ceratainty of boiling recorded when the sample was taken. The certainty is a 4 bit number (between 0 and 15) representing different levels of boiling. The lowest significant bit is always 1 if boiling is occurring.

First & Last Indicator (indicator): These indicators tell the first and last samples to be plotted if the *View* push button is pressed or tranferred to another file if the *Transfer* push button is pressed. All samples betwwen the first and last will be included as well. Last also indicates the current sample if the user is scrolling through samples with the < or > push buttons.

Normalize Radio Button (hot): This radio button determines whether the samples will be normalized when transferred or plotted. If the normalize button is on the y axis is automatically set with 0 as a minimum and 1 as a maximum. If the radio button is off the axes are allowed to reset automatically. Notice, if the screen is not cleared after the normalizing has been turned off, any new plots which are smaller will not reset the axes. To correct this, simply press *Delete Plots* and the axis will be automatically reset with the next plot.

Logarithm Radio Button (normal): This determines whether or not the log of the frequency spectrum will be plotted/transferred.

View Push Button (hot): When pressed all plots between the first and last inclusive are plotted on the *Frequency Spectrum* graph. If last is higher than the number of spectrum stored to the disk, the value of last is set to one higher than the number of spectrums stored and plotting stops.

> Push Button (hot): This allows the user to step through the plots one at a time. The > push button plots the current plot and increments the counter displayed in the *Last* indicator. If the counter is greater than the number of spectrums stored on the disk, the counter is not incremented and the spectrum is not plotted.

< Push Button (hot): This plots the current plot and decrements the counter displayed in the *Last* indicator. If the counter is less than or equal to zero, the counter is not decremented.

Delete Plots Push Button (hot): This push button deletes all the plots shown in the *Frequency Spectrum* graph.

Transfer Push Button (hot): This transfers the current data to the file determined by *Output File* in the format determined by the *Output File Type* indicator.

Save Temperature Radio Button (normal): This determines whether or not the temperatures will be saved for analysis by the neural network.

Randomize Samples Radio Button (normal): When the *Transfer* push button is pressed, if the *Randomize Samples* radio button is on, the samples to be transferred will be transferred in random order. This aids learning in for the neural network.

Output File Type Slide (indicator): Shows the file type for output. **Neural Network Binary** is for use with <u>DTRC93</u> (Appendix C.3). **Neural Network Text** is for use with <u>NeuralWare.</u> **Matlab** is for entry in <u>Matlab</u>. **Binary** is to create another file in the same format as the input files to this program and the output files of <u>Sampler</u> (Appendix C.1).

Input File Indicator (indicator): Shows the file for input.

Output File Indicator (indicator): Shows the file for output.

Frequency Indicator (indicator): Shows the frequency of the graph cursor.

Level Indicator (indicator): Shows the level of the graph cursor.

Edit Push Button (hot): Opens the edit screen for editing of frequency spectrums.

B.2.4 Edit Screen (see **Figure B.2.2**)

Sample # Indicator (hot): This shows the sample # of the graph currently displayed. If changed a frequency spectrum will be plotted.

Temp Indicator (hot): Changing this number automatically writes a new temperature to the disk.

Certainty Indicator (hot): Changing this number automatically write the new certainty to the disk.

> Push Button (hot): Moves to the next sample.

< Push Button (hot): Moves to the previous sample

Frequency Indicator (indicator): Shows the frequency of the graph cursor.

Level Indicator (indicator): Shows the level of the graph cursor. If the *Normalize* function is on the cursor will still plot the true value of the graph cursor.

Mult Indicator (normal): This gives the number the spectrum is to multiplied by. The magnitude of each point in the spectrum is then multiplied by this value if the Multipy type is chosen on the *Options* scale and the *Begin* push button is pressed.

To Indicator (normal): This is the starting point of the destination for any changes. The destination and the source are always the same for multiply operations.

From Indicator (normal): This is the source point of any changes.

Number Indicator (normal): This gives the number of samples to be operated on.

Option Slide (hot): This allows the user to decide what kind of operation he would like to perform on the frequency spectrum. The *Begin* push button must be pressed to begin any operation. If the **Add** option is chosen, a number of signals (given in the Number indicator) will be added from the source (given by the From indicator) to the destination (given by the To indicator). The certainty and the temeprature will be that of the signals which were in the destination. For example, if the *Begin* push button is pressed and the *To* indicator shows 40, the *From* indicator shows 50, and the *Number* indicator shows 5, sample 50 will be added to sample 40 and deposited in 40's location on the disk. The temperature and certainty will remain what they were for 40. Likewise, samples 41 and 51 will be added, samples 42 and 52, and so on up to and including samples 44 and 54. This allows the user to take background samples taken seperately and add them to the boiling or non-boiling samples to simulate those sample taken at the same time. Reactor core background samples can be taken and added to non-core boiling to simulate in core boiling. In a similar fashion, if the **Subtract** option is chosen, the source will be subtracted from the destination. If **Move** is chosen, the source will write over the desination changing the temperature and certainty as well. If **Multiply** is chosen, the the destination will be multiplied by the number given in the *Mult* indicator. If **Edit** is chosen, this allows the spectrum to be changed using the mouse.

Begin Push Button (hot): The *Begin* push button carries out the operation designated by the *Options* slide. If the **Edit** option was chosen, the spectrum is updated with the modifications made by the mouse.

Frequency Spectrum graph (hot): While in the edit mode, the frequency spectrum can be modified by taking placing the mouse button in the *Frequency Spectrum* graph and clicking the button. Hold the button in and move the mouse to change a number of points at once. The modified frequency spectrum will be displayed when the mouse button is released. To save this spectrum to disk, press the *Begin* push button while the **Edit** option is chosen on the *Option* slide.

Close Push Button (hot): This button closes the Edit window.

Figure B.3.1 DTRC94 Main Screen

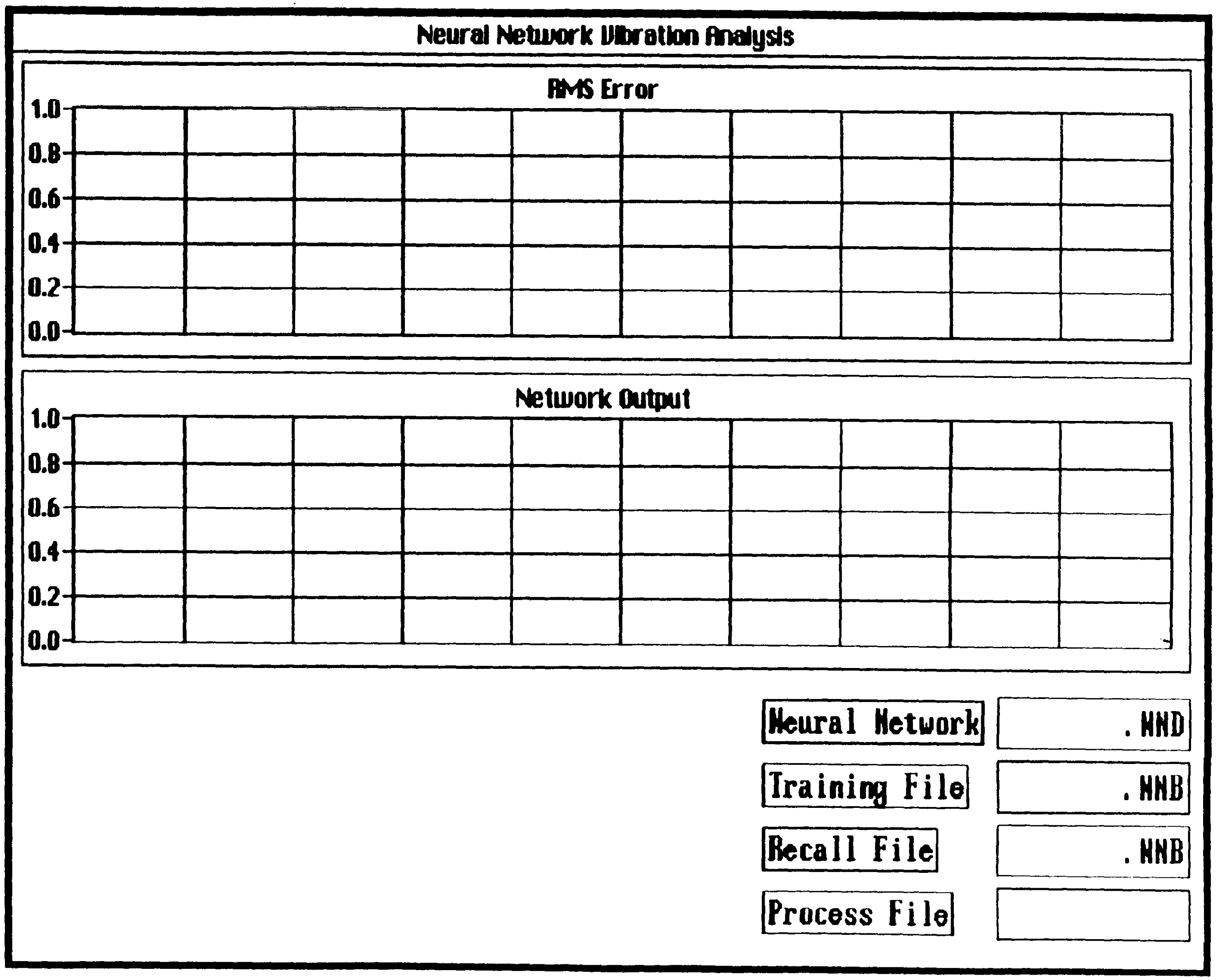
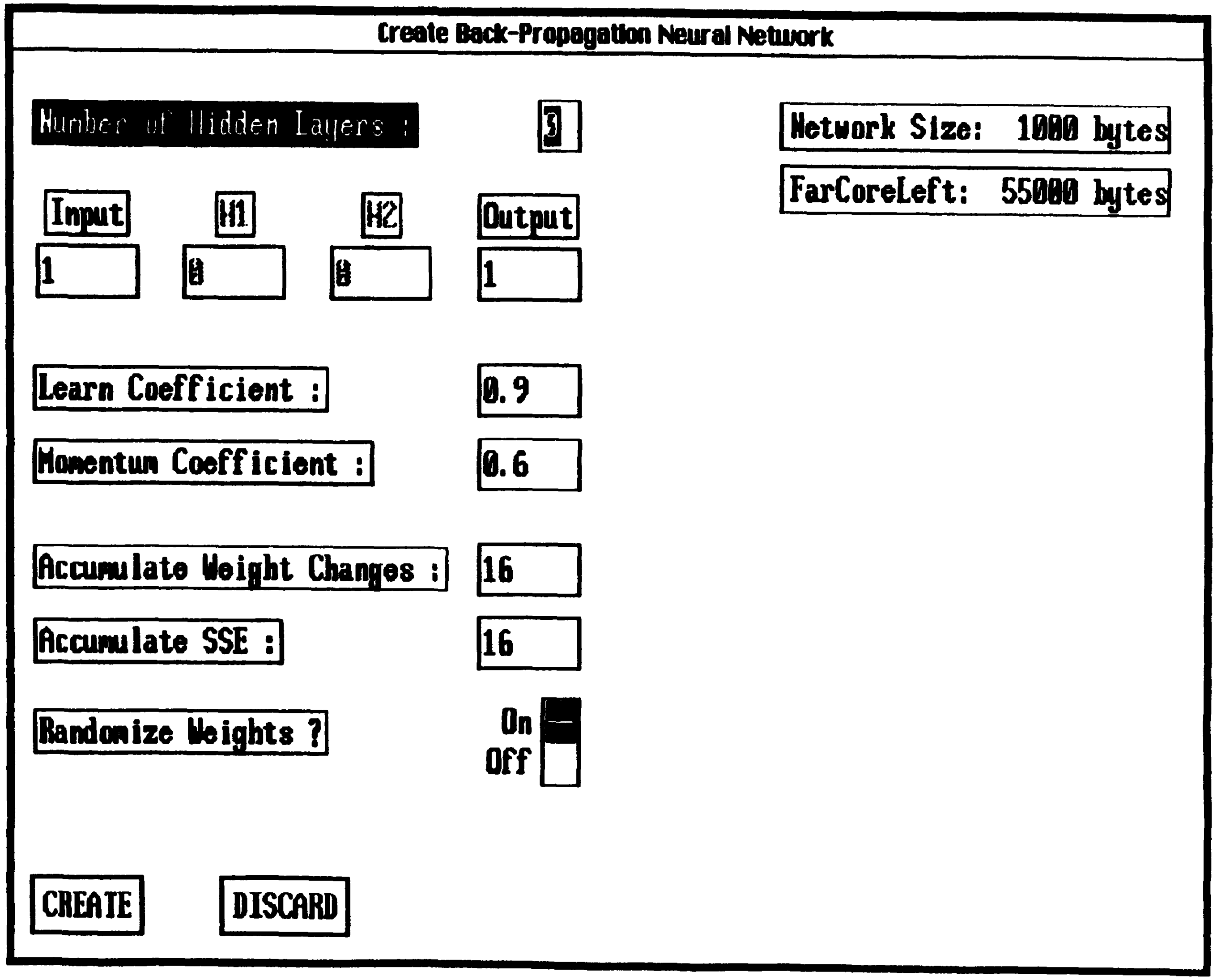


Figure B.3.2 DTRC94 Edit NetworkScreen



B.3 <u>DTRC94</u>

This is the program for training and testing the neural network. It is a slight modification of a program, **<u>DTRC93</u>**, written by William Clement of the U.S. Naval Academy.

B.3.1 Menu Options

Neural Network Menu: Gives the user the following options

New Menu Item: Opens and initializes the Edit Network screen.

Open Menu Item: Loads a previously saved network.

Save Menu Item: Saves the current network to disk.

Edit Menu Item: Opens the Edit Network screen to allow revisions to the current network.

Clear Menu Item: Clears the network in memory.

Randomize Weights Menu Item: Assigns random weights to the links between neurons. This effectively resets the network.

Train Menu Item: Toggles the training condition of the network.

Prune Menu Item: Removes neurons which are not important.

Functional Link Menu Item: Not Applicable

Flash Code Menu Item: Creates a C program that can be linked to which contains the current network. The variables passed to the routine are the inputs and the outputs of the network.

Data Source Menu: Gives the user the following options:

Training File Menu Item: This allows the user to load a binary file containing data to train the network.

Recall File Menu Item: Used to test trained networks.

Process File Menu Item: Not applicable to this application.

DT2831 Menu Item: Not applicable to this application.

Options Menu: Not applicable.

Process Menu: Not applicable.

Quit Menu: Exits the program.

B.3.2 Main Screen (see Figure B.3.1)

RMS Error Strip Chart (indicator): Shows the network error during training.

Network Output Strip chart (indicator): Shows the network output to a recall file.

Neural Network Indicator (indicator): Shows the current neural network file.

Training File Indicator (indicator): Shows the filename of the current training file.

Recall File Indicator (indicator): Shows the filename of the current recall file.

Process File Indicator (indicator): Shows the filename of the current process file.

B.3.3 Edit Network Screen (see Figure B.3.2)

Number of Hidden Layers Indicator (hot): Allows user to change number of hidden layer. the maximum number of hidden layers is 2.

Input Indicator (hot): Displays number of neurons in input layer. This must equal the number of inputs.

H1 Indicator (hot): Displays the number of neurons in the first hidden layer. It is suggested that the user start with more than is neccessary and prune after training. The number allowed depends on the available memory.

H2 Indicator (hot): Displays the number of neurons in the first hidden layer. It is suggested that the user start with more than is neccessary and prune after training. The number allowed depends on the available memory.

Output Indicator (hot): Dipays the number of neurons in the output layer. This must equal the number of outputs of the network.

Learn Coefficient Indicator (normal): The learning coefficient determines how strongly the network will react to errors. It is suggested that this should be three times the momentum coefficient for this application.

Momentum Coefficient Indicator (normal) Determines the momentum of change the network will have while learning.

Accmulate Weight Changes Indicator (normal) Determines how many wieght changes the network will sum before making a revision. One is suggested.

Accumulate SSE Indicator (normal): Determines how many SSE to add before using to make a weight change decision.

Randomize Weights Switch (normal): If on, when the edit screen is exited the weights for the network will be scrambled.

Create Push Button (hot): Makes the changes to the network and returns to the main screen.

Discard Push Button (hot): Returns to the main screen without making any changes to the network.

Network Size Indicator (indicator): Shows the size in bytes of the network.

FarCoreLeft Indicator (indicator): Shows the number of bytes of memory remaining.

Figure B.4.1 Boiling Detector Main Screen

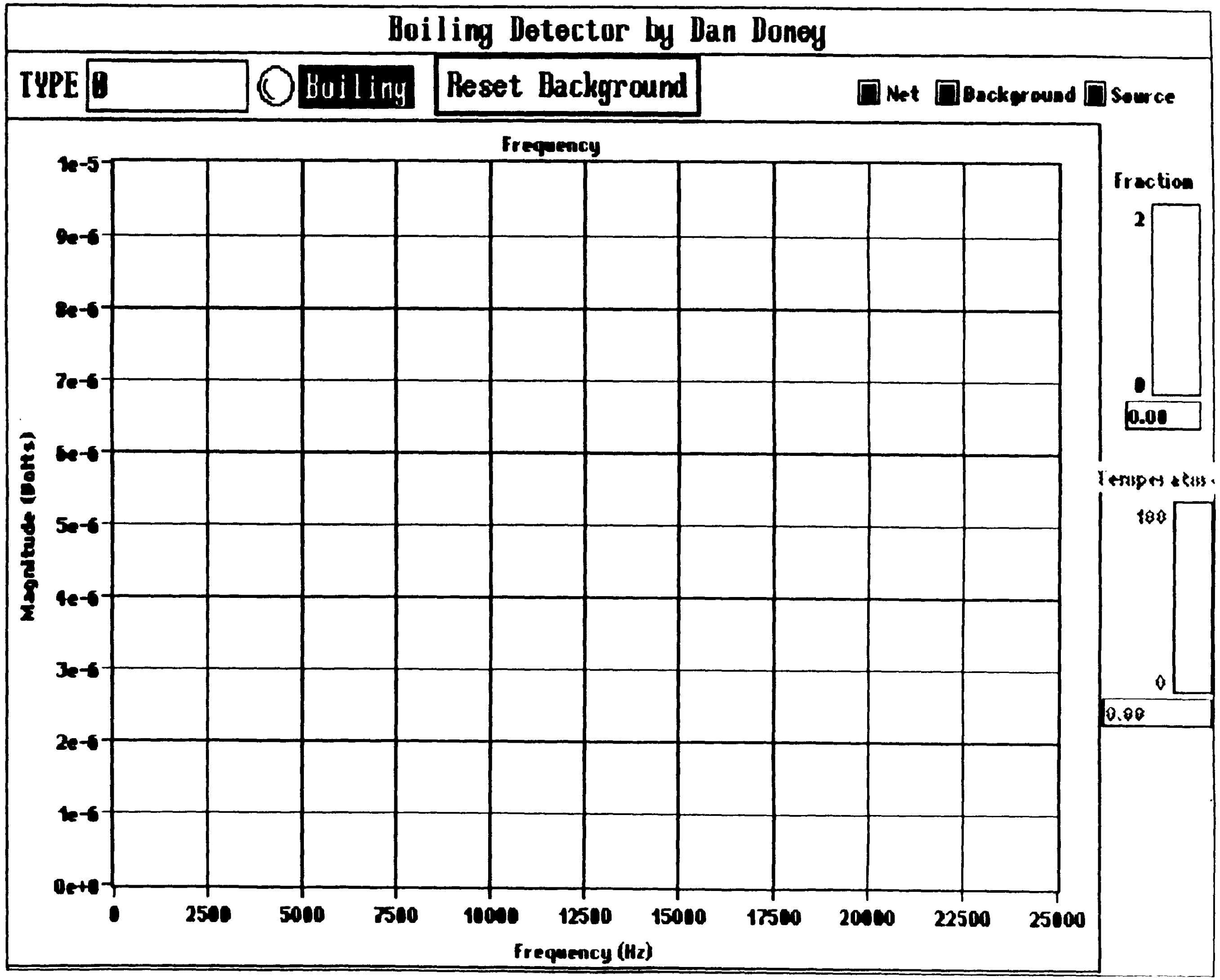


Figure B.4.2 Change Axis Screen

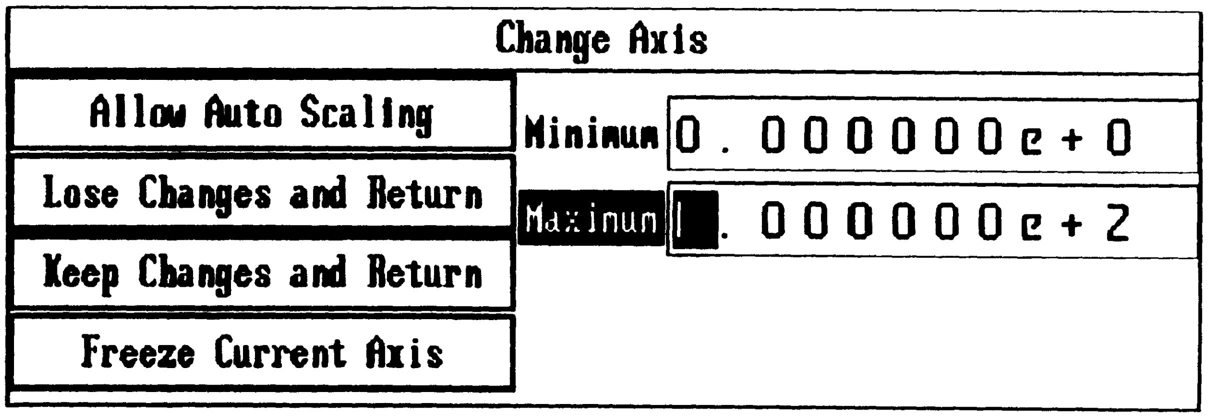


Figure B.4.3 Boiling Detector Options Screen

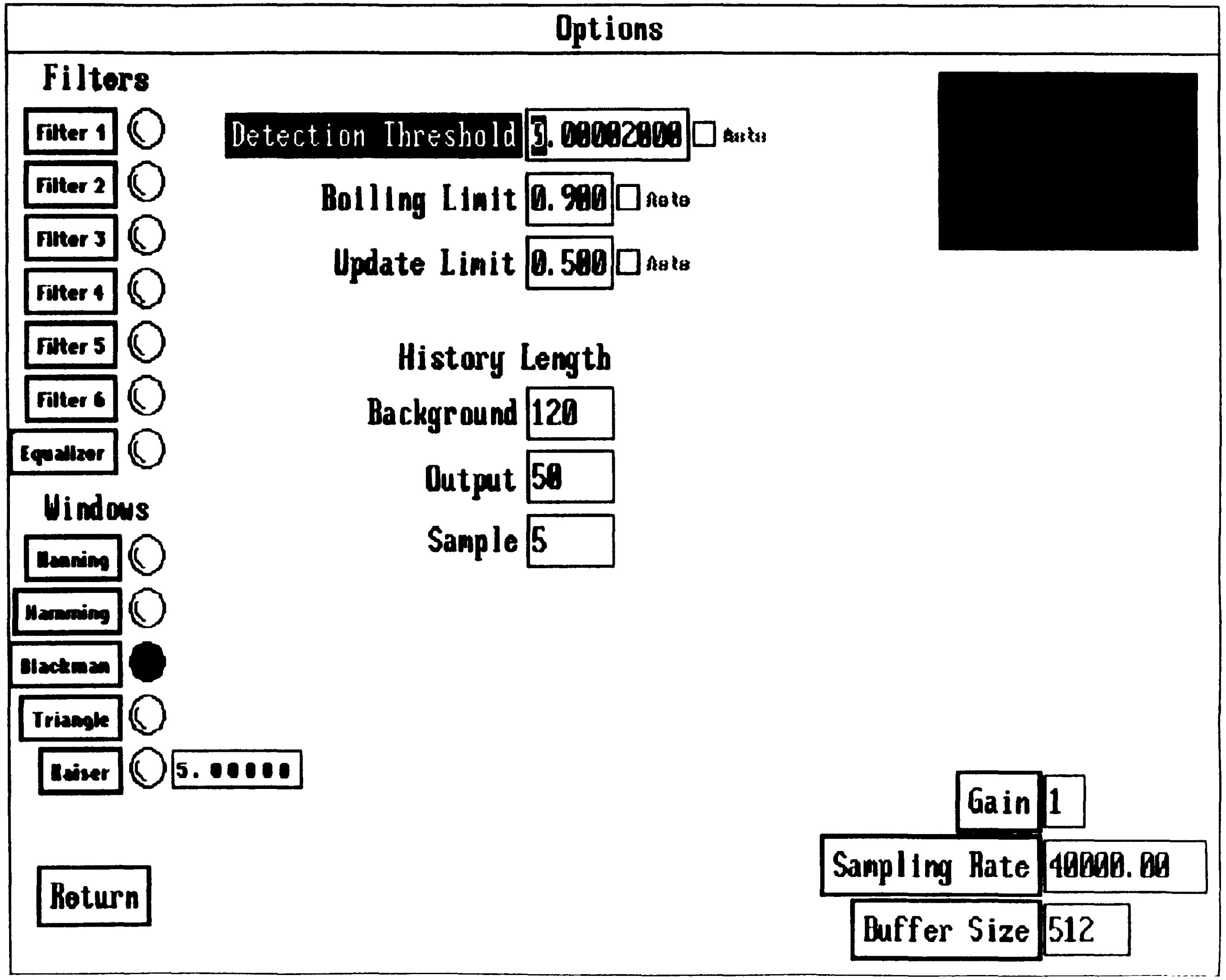


Figure B.4.4 Edit Filters Screen

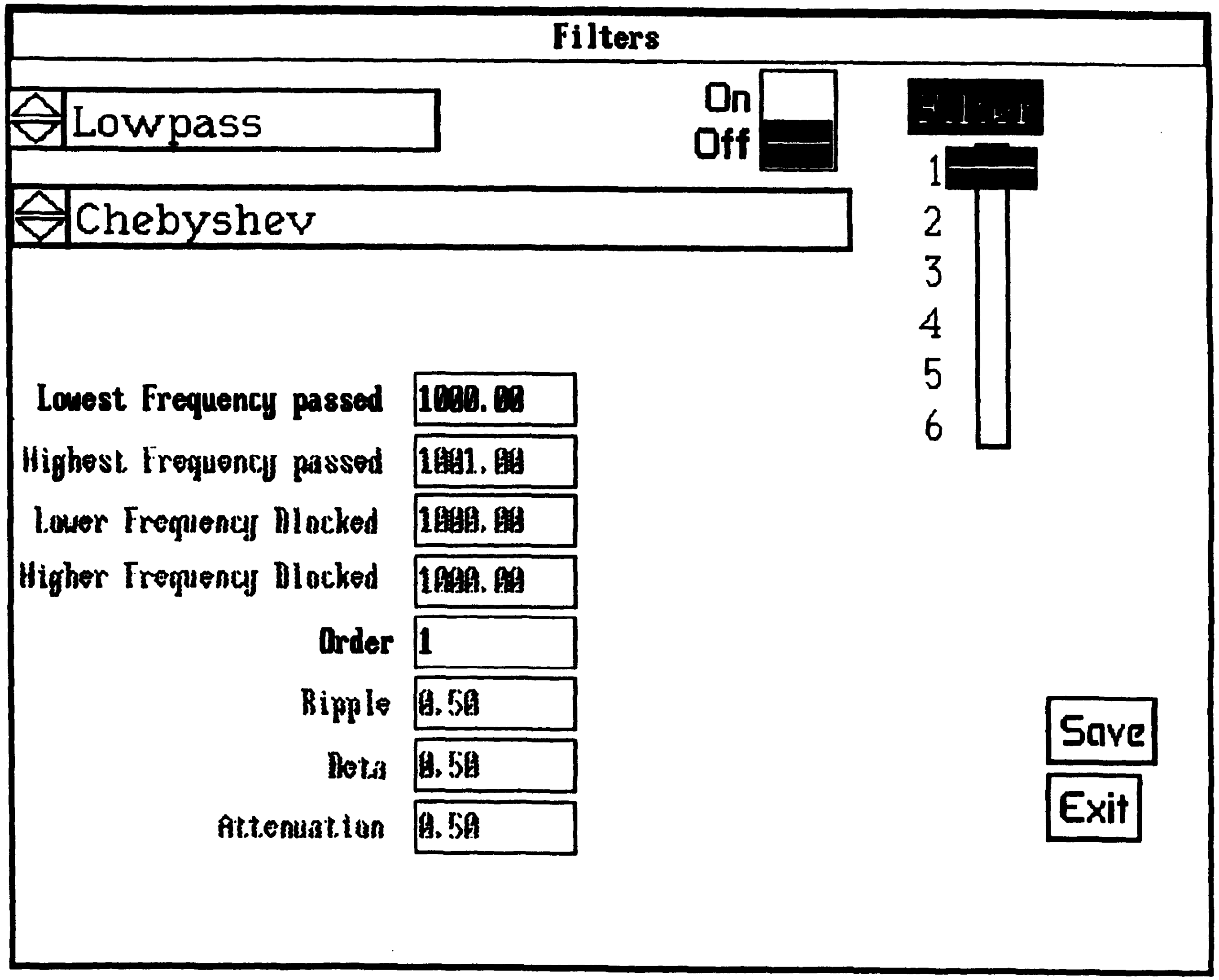


Figure B.4.5 Edit Equalizer Screen

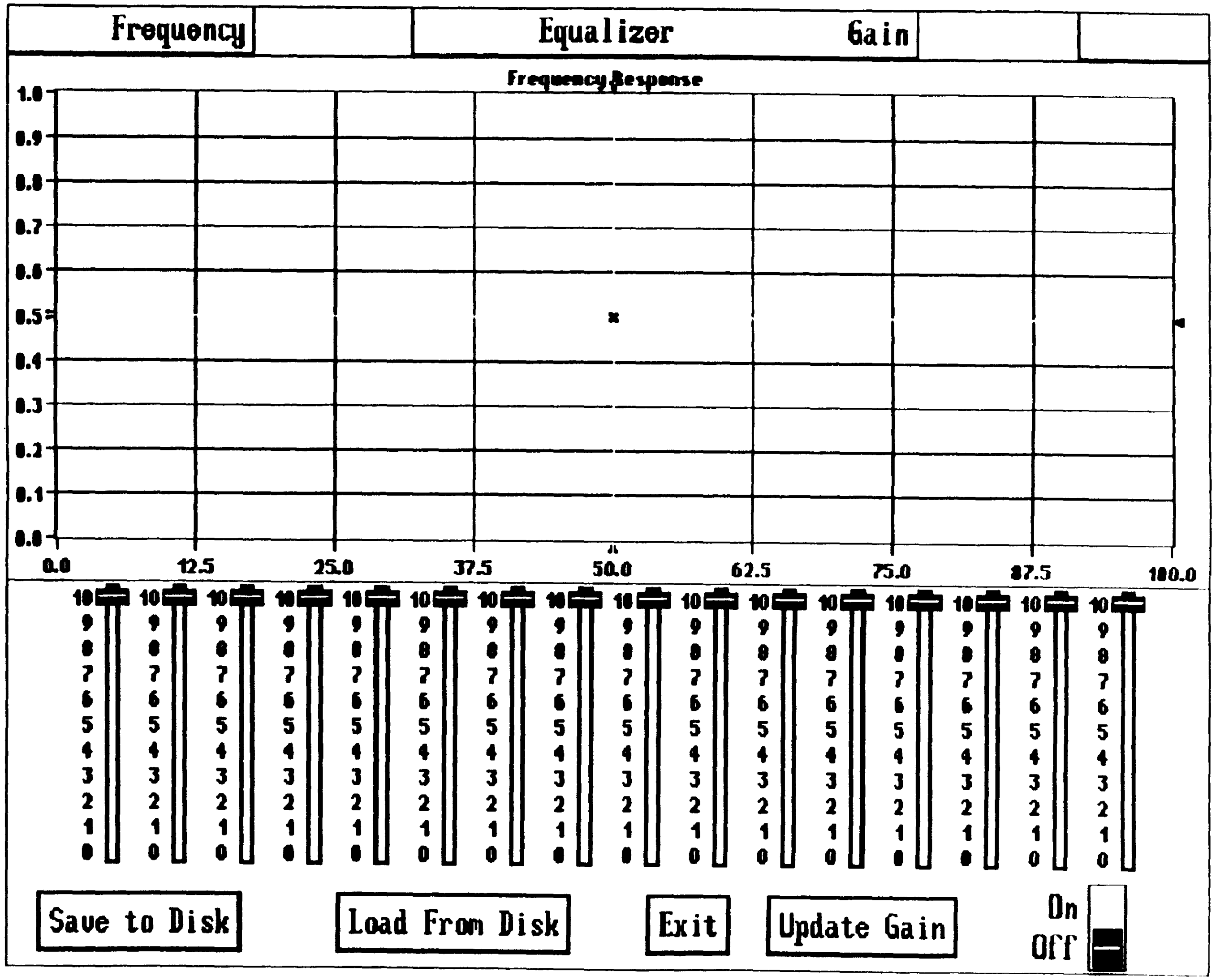


Figure B.4.6 Change Gain Screen

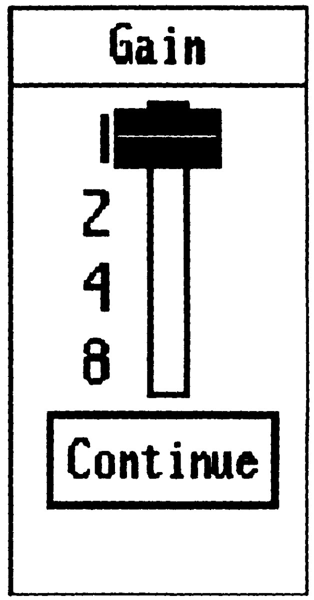
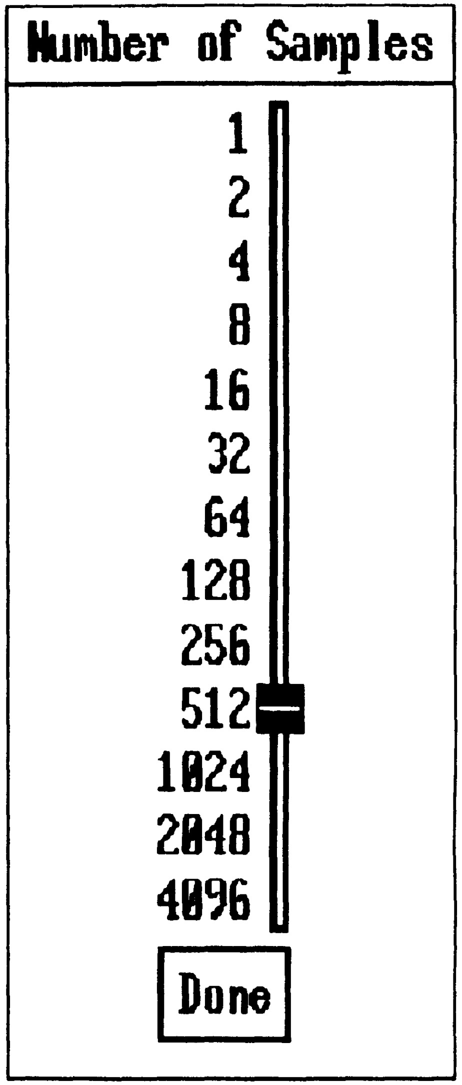


Figure B.4.7 Change Buffer Size Screen



B.4 <u>Boildet.exe</u>

This program is the boiling detector. It gets the data from the data aquisition board, modifies it, enters it into a user specifies neural network, and outputs the data.

B.4.1 Start up

On start up, the user is asked to chhose a neural network. This must be a previously trained neural network. Files with the extension .nnd are binary files which contain, weights and other information about the network. After the network is loaded, the program begins taking data immediately. First the program initializes the backgound signal, then the detector begins functioning normally.

B.4.2 Main Screen (see **Figure B.4.1)**

Type Indicator (indicator): The type indicator is used to tell the user what type of boiling is occuring. A value of 0 means no boiling. A value of 1 represents the onset of boiling. Higher values indicate more severe boiling conditions.

Boiling LED (indicator): The boiling LED turns red in the event that boiling is detected.

Reset Background Push Button (hot): When the reset background button is pressed, the background history is initialized. A new background signal is taken.

Net Radio Button (hot): The net radio button turns on/off plotting of the net signal.

Background Radio Button (hot):The background raio button turns on/off the plotting of the background signal.

Source Radio Button (hot): The source button turns on/off the plotting of the source signal.

Fraction Scale (indicator): The fraction scale shows the current certainty of boiling. A value of zero indicates that the network output history has no values representing boiling. A value of 1 means that the average of the values stored in the network output history are 1 which is a good indication of boiling. This value can be any number in between 0 and the highest possible boiling value.

Temperature Scale (indicator): If the network is trained to determine the temperature, this scale is used to give the temperature predicted by the network.

Frequency Graph (indicator): This is where the frequency spectrums are plotted.

B.4.3 Menu Options

Axes Menu: This allows the user to set the maximum and minimum values for plotting.

X Axis Item (hot): Selecting this item pops up a screen for adjusting the axis.

Y Axis Item (hot): Same as above but for Y-axis.

B.4.3.1 Axis Screen (see Figure B.4.2)

Minimum Indicator (normal): This sets the minimum value for plotting.

Maximum Indicator (normal): This sets the maximum value for plotting.

Cancel Push Button (hot): Pressing cancel returns the user to the previous screen without making any changes to the axes.

Done Push Button (hot): Makes the requested changes to the axes.

B.4.3.2 Options Screen (see Figure B.4.3)

Hanning Push Button & LED (hot): Pressing the push button toggles the state of the window. The LED indicates the current state of the window. The window is used for filtering the data. The default state for this window is off.

Hamming Push Button & LED (hot): Same as above.

Blackman Push Button & LED (hot): Same as above. Default state on.

Triangle Push Button & LED (hot): Same as above.

Kaiser Push Button, LED, & Indicator (hot): Same as above. The indicator tells the beta value which determines the shape of the kaiser window.

Beta	Window
0.0	Rectangular
1.33	Triangle
3.86	Hanning
4.86	Hamming
7.04	Blackman

Remember, a higher the value results in more data lost but better smoothing.

Detection Threshold Indicator (hot): This is used to set the detection threshold. The auto radio button next to the indicator is to turn and the automatic threshold resetter which is not yet enabled. The purpose of each of these values is mentioned in Chapter 5.

Boiling Limit Indicator (hot): Same as above.

Update Limit Indicator (hot): Same as above.

Background Indicator (hot): This sets the length of the history.

Output Indicator (hot): Same as above.

Sample Indicator (hot): Same as above.

Return Push Button (hot): Returns to the main screen.

Filter Push Buttons & LEDs (hot): The LED indicates the current state of the filter. Pressing the push button on any filter will automatically open the filter editor screen and set the values to those of the filter selected.

B.4.3.2.1 Filters Screen (see Figure B.4.4)

Filter # Slide (hot): Once in the filter editor screen the user can change to a different fiter by selecting it on the slide. Selecting a different filter will automatically reset the screen to the values of the filter selected.

Filter Type Roller (hot): The type of filter can be changed. The options are:

<u>Filter Types</u>
High Pass
Low Pass
Band Pass
Band Stop

Selecting a type will automatically toggle the indicators to allow data entry on only the applicable indicators.

Filter Name Roller (hot): The name of the digital filter can also be selected. each name uses a different teqnique for filtering the data. The options are:

<u>Filter Name</u>
Butterworth
Chebyshev
Inverse Chebyshev
Elliptic

Selecting a name will automatically toggle the indicators to allow data entry on only the applicable indicators.

Filter State Switch (normal): This turns on/off the filter.

Lowest Frequency Passed Indicator (normal): For high pass filters, this sets the cutoff frequency. For Bandpass filters, this is the lowest frequency of the passed band.

Highest Frequency Passed Indicator (normal): For low pass filters, this sets the cutoff frequency. For Bandpass filters, this is the highest frequency of the passed band.

Lower Frequency Blocked Indicator (normal): For the band block filter, this gives the lowest frequency of the blocked band.

Higher Frequency Blocked Indicator (normal): For the band block filter, this gives the highest frequency of the blocked band.

Order Indicator (normal): This gives the order of the digital filter.

Ripple Indicator (normal): This gives the number of ripples for the digital filter.

Beta Indicator (normal):

Attenuation Indicator (normal): Gives the attenuation for a band blocking filter.

Save Push Button (hot): Saves the filter settings to memory. Note that is save is not pressed before exitting, the filter state will be left as it was before the editor was entered.

Exit Push Button (hot): Closes the filter editor window and returns the user to the previous screen.

Equalizer Push Button & LEDs (hot): The equalizer LED indicates the state of the equalizer (red on, blank off). Pressing the equalizer button opens the equalizer screen.

B.4.3.2.2 Equalizer Screen (see Figure B.4.5)

///// *Equalizer* Graph
///// *Equalizer* Slides
///// *Update* Push Button
///// *Save* Push Button
///// *Load* Push Button
///// *Return* Push Button

Gain Push Button & Indicator (hot): Pressing the gain push button will open the gain screen. The indicator shows the current gain.

B.4.3.2.3 Gain Screen (see Figure B.4.6)

Gain Slide (normal): For use with the DT2831, there are 4 options for gain, 1,2,4,and 8. A gain of 1 is the default. It is not suggested that a higher gain be used since if their is a strong amplifier bias, data will be lost with a high gain.

Continue Push Button (normal): This closes the gain editor screen, saves the change to memory, and returns to the previous screen.

Buffer Size Push Button & Indicator (hot): The buffer size indicator shows the current buffer size for the information retrieved by the data aquisition board. This value should not be changed since the network has been trained at a specific level. If the buffer size push button is pressed, the number of samples screen will be opened.

B.4.3.2.4 Number of Samples Screen (see Figure B.4.7)

Number Of Samples Slide (normal): the user can choose the number of samples taken.

Done Push Button (hot): This button save the buffer size selected to memory and return to the previous screen.

Sampling Rate Push Button & Indicator (hot): The indicator shows the current board sampling rate. If the push button is pressed the user will be prompted to enter a new sampling rate. This is not suggested since the network has been trained at 40000 Hz.

Figure B.5.1: Fileread Main Screen

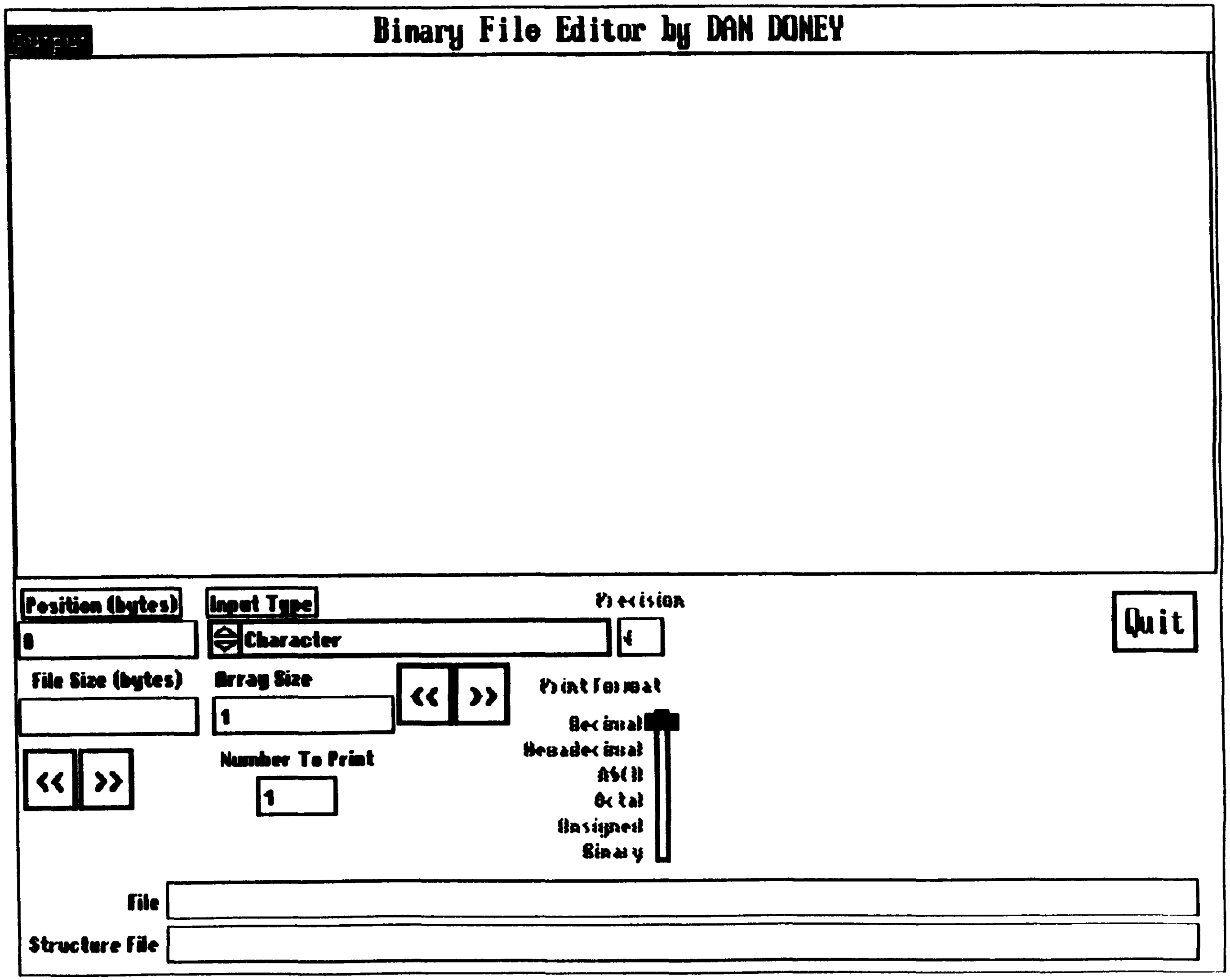
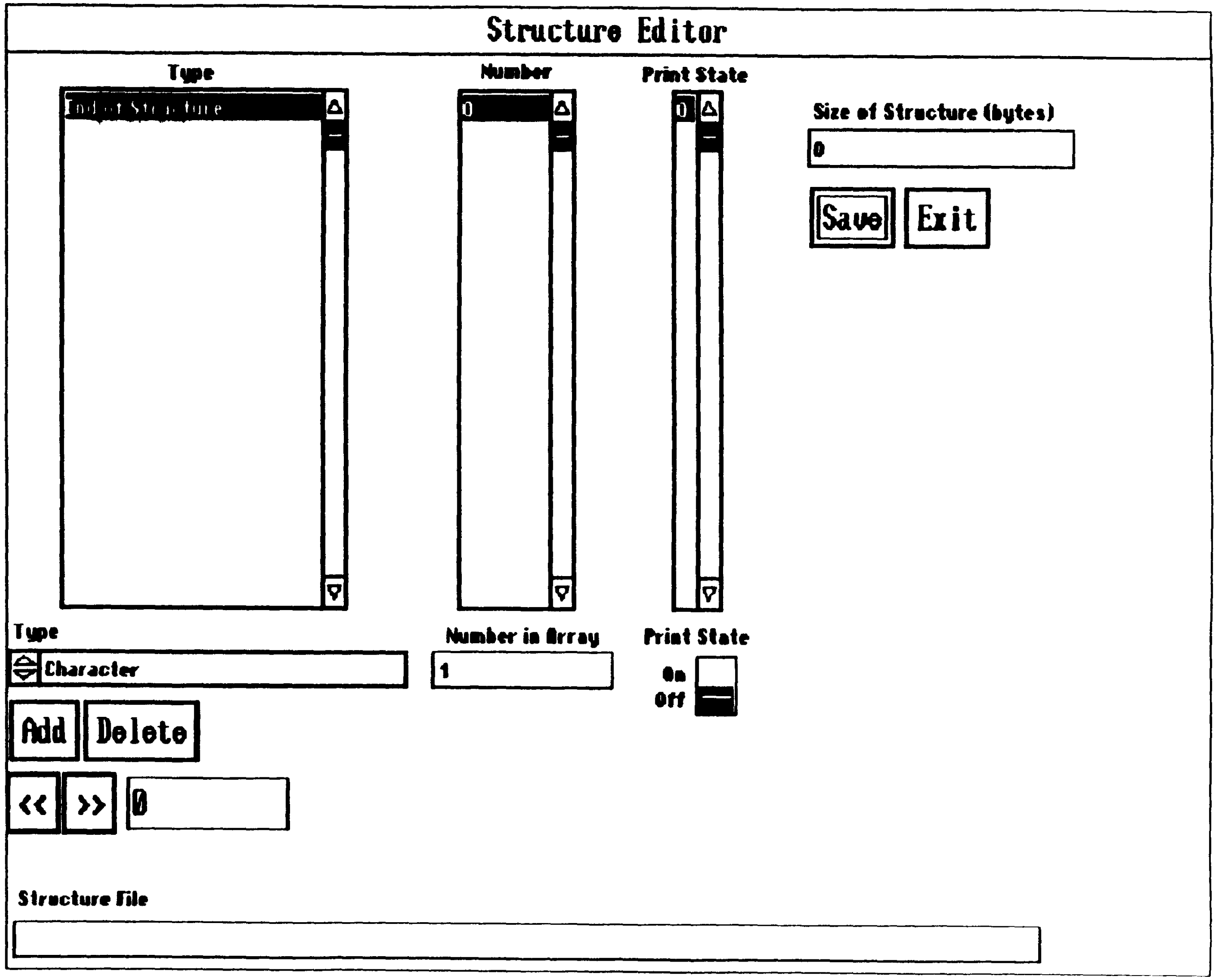


Figure B.5.2: Fileread Edit Structure Screen



B.5 Fileread

Fileread is a utility program to read binary files. It enables the programmer to check the binary files created in the earlier programs for errors.

B.5.1 Startup

To run this program type fileread at the DOS prompt. On start up the user will be prompted to enter the file to be read. The user will then be placed in the Main screen.

B.5.2 Menu Options

File Menu: Selecting file alllows the user to change the file to be edited.

Quit Menu: Exits the program.

Structure Menu: Gives the user the following options:

Load Menu Item: Loads a previously created structure and returns to main screen.

Cancel Menu Item: Returns the user without making any changes.

Create Menu Item: Opens the Create Structure sceen:

B.5.3 Create Stucture Screen (see Figure B.5.2)

Type Selection Box and Ring Control (hot): Shows the data types in the structure. Double clicking on a type automatically brings it up for editing. The ring control allows editing of the type.

Number in Array Selection Box and Indicator (hot): Shows the number in the array of that particular data type in that location of the structure. Double clicking automatically brings up that location for editing.The indicator allows editing of the number in the array.

Print State Selection Box and Switch (hot): Shows whether or not to print that array of data types. Double clicking automatically brings up that location for editing. The switch allows editing of the print state.

Add Push Button (hot): Adds the array to the structure.

Delete Push Button (hot): Deletes the array from the structure.

>> Push Button (hot): Moves to next array in structure.

<< Push Button (hot): Moves to previous array in structure.

Position Indicator (hot): Moves to array in indicated position.

Structure File Indicator (indicator): Shows the filename of the structure.

Size of Structure Indicator (indicator): Shows the size in bytes of the structure.

Save Push Button (hot): Saves structure to disk.

Exit Push Button (hot): Returns to main screen.

B.5.4 Main Screen (see Figure B.5.1)

<< Push Button (hot): Moves one bytes forward in the file.

>> Push Button (hot): Moves one byte backward in the file.

Position Indicator (hot): Shows the position in bytes from the beginning of the file.

File Size Indicator (indicator): Shows the size of the current file.

InputType Ring (hot): Allows user to choose the data type being requested.

Precision Indicator (normal) Allows the user to choose the printing precision for **floating point** and **double precision** data types.

Output Text Box (indicator): Shows the requested data.

Array Size Indicator (normal): Allows the user to choose the number of data of the type indicated by the *Input Type* Ring.

Number to Print Indicator (normal): Allows the user to choose the number of data types from the array to print.

>> Push Button (hot): Reads next array of data from the file starting at the current position.

<< Push Button (hot): Reads the array of data from the file before the current position.

Print Format Slide (normal): Allows the user to decide what format a **byte** of information wll be printed in.

File Indicator (indicator): Shows the file being read.

Structure File Indicator (indicator): Shows the file containing the structure format surrently used.

References

Srinivasan,G.S.and O.P Singh, "New Statistical Features Sensitive to Sodium Boiling Noise", *Ann. Nucl. Energy*, Vol.17, No. 3, pp.135-138, 1990.

Westwater, J.W, A.J. Lowery, Jr., and F.S. Pramuk, "Sound of Boiling", *Science*, Vol 122, 332-333, 1955.

Koski, J.A., A.G. Beattie, J.B. Whitley, and C.D. Croessmann, "Experimental Verification of Subcooled Flow Boiling for Tokamak Pump Limiter Designs", The American Society of Mechanical Engineers papers, 345 E. 47 St., New York, N.Y. 10017, 1987.

Nesis, Y. I., "Acoustic Noise of a Boiling Liquid", Heat Transfer - Soviet Research, Vol. 22, No. 6, 1990. Translated from Two-Phase Flow: Heat Transfer and Unsteady-State Processes in Turbomachinery. USSR Acad. Sci., pp. 68-74.

Todreas, N.E., Kazimi M.S. Nuclear Systems I. Hemisphere Publishing Corporation, 1990, page 534, Eq. 12.16

Lubbesmeyer, D. "Experimental Reactor Noise - A Review On Noise-Analytic Measurements of Thermohydraulic Parameters in Operating BWR's and Their Interpretations", *Progress in Nuclear Energy*, Vol 14, No 1 pp-41-93, 1984.

Lab Windows Advanced Analysis Library Reference Manual, Vol 2.3. National Instruments, 1993.

Neural Computing, NeuralWare, Inc. 1991.

DT2831 Hardware User Manual, Data Translation, 1991.