

THE ANDROID BRAIN 101

I must first make a statement that I authored some years ago which conveys my deepest thoughts about the “Human Voice”.

“The most beautiful sound I have ever heard is the human voice in song or word. I have studied it and know darn well that it was born in heaven and not groans from hell.”

Before we begin our analysis of the android “speech recognition subsystem” let us do some simple experiments using the “human biological speech generation mechanism”. These experiments will not require any test equipment other than our ears. There are five or if you prefer six primary components which allow us to generate natural speech, two types of carrier generators and three modulation generation components, the tongue, the teeth, and the mouth (i.e. the lips and jaws). We have two ways of generating a carrier for our spoken words. One carrier is created by the air forced over the excited/stretched vocal cords which then vibrate at various audio frequencies and then second is the rush of air over the relaxed vocal cords with out vibrating the vocal cords which creates audio white noise or “a hiss” that we use as the carrier of “whispered” words. These carriers contain little or no information by them selves but are the conveyors (i.e. carriers) of the information i.e. the spoken words. Now as our first experiment exhale/inhale with your vocal cords relaxed. You will hear a rush of air which is known as white noise but you have conveyed NO WORD unless you wish to call this a sigh and let me state that the spoken word a sigh is “SIGH. Now again force air over your tightened/excited vocal cords and you will hear them vibrate generating a specific tone/note but again there is no information on this carrier i.e. a spoken word. Now firmly place and hold your tongue on the bottom of your mouth against your front gums and attempt to count from zero to nine. Some parts of some of these ten spoken words can be distinguished. Now speak these ten words normally using all three modulation components. Note that these ten words are created by using the three modulation components, the tongue, the teeth and mouth. The modulation placed on the carrier by the tongue, teeth and mouth is the spoken word which you wish to convey by modulating a carrier. Some of the letters sent to me by my readers state that speech can be continuous . Given the fact that the nerves controlling the muscles in the three modulation components require several hundred microseconds to transmit the control impulses to the muscles to form the next word after the previous word is completed, and the fact that these muscles require time to expand or contract, and the fact that the muscles and the surrounding flesh also have mass which must be moved into position which takes time, what appears to the listener to be continuous speech in fact is not. Continuous speech is actually continual speech with a few milliseconds of pause/slur between the end of the previous word and the beginning of the next word while the tongue, teeth and mouth are repositioned to modulate a continuously generated carrier i.e. a vibrating vocal cord or a white noise hiss. This slight pause or slur can easily be seen using a “storage oscilloscope” that can freeze the modulated carrier wave form reproduced by a microphone during so called “continuous speech” and the imperceptible pause/slur can be clearly seen. Therefore continuous human speech does not exist due to the “LAWS OF PHYSICS” and the transmission time of the human nervous system as well as the time required by the muscles to reposition the three modulation components for the next word to begin. Only “continual speech” can be accomplished by any human. I believe that this irrefutable science should clear up any confusion in the matter. The android speech recognition hardware circuits strip away the carrier and suppress the slurs leaving only the word modulation envelope with distinct pauses between words which are then digitized.

These pauses simplify the word recognition software and allow us to use faster simpler unsigned integer calculations instead of slower more complex floating point calculations.

The human brain will be our model for the design of the “ANDROID BRAIN” and we will reverse engineer the functions. Before we can begin to understand even the basic principles of the operation of the “ANDROID BRAIN” we must first understand on some level the principles of the neurobiological brain and more specifically its most basic component the most prevalent type of “NEURON” found in the human brain. In later articles we

will examine the human brain at the macro level but for now we must concentrate on the neuron and its component parts. We are not necessarily interested in the biochemical processes but rather the logic functions that are represented by the biochemical processes. Some of my readers may be well versed in the biochemistry of the neuron while the majority most likely are not. In either case study carefully these functions as they are the foundation of the “ANDROID BRAIN” . “SPEECH RECOGNITION” is one of the simpler macro functions in the android brain and is the most important first step for future android training.

A NEURON is made up of three major parts: the cell body, the dendrite inputs and an axon output. These form the synaptic interconnections between nearby neurons. There are multiple dendrites; some enablers and some inhibitors, but only one axon on each neuron. The fact that each neuron has only one axon usually with multiple nodes makes perfect sense. After a neuron fires no further arbitration need take place. Multiple dendrites, each having multiple nodes, also makes sense. The arbitration must take place prior to the triggering of the neuron much like a combination of arbitration logic circuits in a digital computer. However, there is a third parameter that is contained in the neuron cell body and that parameter is the triggering threshold. This is the voltage at which the neuron will fire sending the impulse through the axon to subsequent connected neurons. Each dendrite adds to or subtracts from the threshold which must be reached to trigger the neuron based on its entry point on the neuron cell body. So each dendrite has two factors associated with it. It is either an inhibitor or enabler and it has a weighting factor. Therefore each neuron is equivalent to a complex set of digital logic. It is easy to see why the biological brain containing over 100 billion neurons is far more complex than any single or huge network of the worlds most powerful computer systems and associated software.

When designing an ANDROID BRAIN we must first develop a neuronal element which has the same, similar or super characteristics compared to a biological neuron. while on the surface this seems fairly straight forward in fact it is not that simple and contains many hidden pitfalls. If our model fails even in some seemingly small way the resulting neuronal network will eventually crumble like a house of cards.

One approach, which has been researched for over sixty years, is "neural net technology". This is basically the implementation of a statistical model comprised of two or more layers of neurons that are "N" neurons wide where all possible connections are already made and given a starting weight of zero. Then as the external stimuli are applied via the learning algorithm the weighting is increased on the stimulated synaptic pathways. This approach does not emulate the neurobiological model previously discussed. One important area is that in the neurobiological model the interconnections, synaptic pathways, are not predefined and no practical mathematical model of the predilection for synaptic interconnections has yet been developed. For this and many other reasons it should be obvious why conventional neural net designs have not yielded true intelligence or even simple reasoning capabilities. In this project we will take a major step into the unknown inner space realm of “MACHINE NEUROSCIENCE”. This realm exists not in hardware but rather in software abstract concepts that do not yield to standard “BOOLEAN“ or other mathematical interpretations.

Let us explore some of the reasons that our conventional mathematical principles simply do not work. Albert Einstein said it best; “ Where mathematics refers to reality it is uncertain and where it is certain it does not refer to reality “. Computers are designed using a combination of “Boolean Algebra” and “Conventional Mathematics”. The heart of every computer is the ALU “Arithmetic Logic Unit” this unit executes Boolean Algebra and Conventional Mathematical functions. Let us first look at Boolean Algebra as compared to the “Biological Neuron”. The basic Boolean functions are “AND”, “OR” , and “NOT/INVERT” . All other Boolean functions such as “EXCLUSIVE OR” etc. can be derived from these three basic functions. In the biological neuron there is no “INVERT” function or “OR” function or “EXOR” function there are only prioritized threshold based “AND” functions in each “DENDRITE” attached to the “neuron“ body. Each dendrite can have many nodes and each node has a priority and persistence characteristic. because of the persistence factor all nodes do not need to be active at exactly the same time. However in a Boolean “AND “ function all nodes must be active a the same time to satisfy the function input nodes. There are many more incompatibilities too numerous for this article. We could do as we have always done and “insert band-aid fixes” to modify the Boolean rules to fit the characteristics of the biological neuron dendrites which would be impractical and as brittle as a house of cards. Or we could take the

more reasonable course and develop a completely new set of mathematical principles and rules for calculations in this new domain. I have developed such a math system for this new domain. This math system can be used to describe both the micro and macro functions of single and complex groups biological neurons. In the biological brain there is no identifiable "ALU". Therefore, "Conventional Mathematical" sub systems simply do not exist in the "Biological Brain" so there is no frame of reference to compare it to our "Conventional Mathematics" rules. I have many domestic and foreign patents based on my research of the neurobiological image analysis systems in higher animals. This and other research indicates that the brain uses abstract patterns to represent math and logic functions instead of the rigid absolute approach we use in the scientific community. As an example the "WIRING" in any two brains of even animals of the same genus such as mice who come from the same parents and environment are never wired the same. However both mice are capable of solving the same matrix. I believe we should all stand back as Albert Einstein states in his statement I quoted earlier and take a good hard look at our basic concepts of mathematical principles. It is difficult to defend a math system so full of band-aids to fix problems like "the calculation result of any number divided by zero is equal to one" to prevent infinite calculation loops or fixes to prevent the existence of positive and negative zero and what about the problems with floating point positive and negative zero crossing exponents and characteristics when used in some types of calculations etc, etc.

SUMMARY

The purpose of this design exercise is to introduce my readers to pattern based waveform analysis. In the later articles I will show how pattern based logic and mathematical calculations can be performed rather than the rigid so called "SCIENTIFIC MATHEMATICAL" methods that we hold above human reasoning. Don't misunderstand, our current scientific mathematical methods are wonderful tools but nothing more. They can not replace human reasoning. Science and mathematics by them selves can not reason. They are just a sophisticated tool like the ones you would use to fix or build a car. It is the reasoning humans using those tools and understanding their limitations that make them great. Science and Mathematics are not a substitute security blanket for human reasoning. Another area that will become obvious in later articles and programs is the fact that there can be no learning as in the neurobiological brain with out the inherent ability to make mistakes and correct them. Otherwise the machine brain has no autonomous reasoning abilities and all that is accomplished when a new problem is presented to it is the use of existing pre-programmed solutions. No new synaptic pathway reasoning or solutions can be gained using this approach. There is no true reasoning inherent in that type of design. This is the major root problem with all existing conventional design methods. It should be obvious that completely new methods need to be researched and developed. This is what we shall do together in this project. It is my fervent hope that some of my readers will continue to research and develop even better ways to reach the goal of human like characteristics in a machine brain and pursue this new field that I dubbed decades ago "MACHINE NEUROSCIENCE".

Now we can begin to follow the flow of the information contained in the spoken word and how the machine processes it. Speech recognition and learning occur at the same time in this Android subsystem. Most if not all prior attempts at "Speech Recognition" break this into two separate functions "learning" and "recognition". The human brain does these functions at the same time and so do we.

How It Works

In the previous set of articles we covered the operation of the hardware preprocessing circuits, the digitizer, and COG digitizer driver software which collected the eight bit samples of the spoken word modulation envelope. As the samples are collected they are checked for certain noise spikes and other characteristics to collect the best possible data. The number of valid samples is tracked and counted. At this point things become very different from any other approach and more like the human brain. Instead of using the samples directly we convert the samples to a continuous "DELTA MODULATION" bit string where each eight bit byte sample is represented by a single bit in the string. While this does result in an 8 to 1 data reduction it is not the primary reason for creating the delta modulation bit string. It is done so that some parts of the word could be compared bit by bit to previously

learned information for the closest match i.e. if need be make an educated guess. There is not enough memory in the propeller chip to fully implement this. While the samples are being collected a set of parameters are assembled which describe certain characteristics of the wave form. These parameters are then combined using an algorithm very similar to a tape or disk error checking algorithm to form a unique pointer to a synaptic pathway to the neuronal element that represents the word that is being spoken including any accents etc. If the synaptic pathway is empty then the system prompts the user to spell the word. The system then checks to see if that word is already in its vocabulary. If it does exist it connects the synapse to the neuronal element representing the word otherwise it enters the word into its vocabulary and sets up a neuronal element that represents the word. It then prompts the trainer to enter an action character string that will be sent out on the serial port pin P25 on the propeller when the spoken word is subsequently recognized. The word recognition, actions and learning are performed in “real time” on one Propeller using two COGs plus the interpreter. This makes a total of three COGs. The serial output is on pin P25 at 19.2k baud. This pin is normally used as part of the mouse interface. This can be connected to your robotic project or computer. Since the main memory in the Propeller chip is only 32k bytes we have limited the vocabulary to 256 x 15 byte words and the same for the control string and the number of synaptic pathways is limited to 8192. We would really like to have orders of magnitude more synapses but the limited memory at this point does not have room for more synapses or vocabulary. Later in the project after we install the KISS OS we will then have almost unlimited memory. This part of the project will accomplish two major goals. The first is to build and test the speech recognition and the second is to introduce you to a completely new approach to “MACHINE NEUROSCIENCE”. A Parallax Datalogger module is used to store all of the Vocabulary, Control and Synaptic information files in “windows compatible format. This module uses one additional COG and four control pins. It will connect to any USB storage device and supports FAT16 or FAT32 files. Some words will require more synaptic connections than others. You must take the time to train the machine as you would to teach your pet or a child a new word or command. This will require the same amount of patients to accomplish as it would with your pet or child but can produce a useful front end for most any project in any language that you choose. As we add more real and virtual memory to this speech recognition subsystem similar to the human brain it will continue to become more accurate and powerful. This subsystem will eventually eliminate the need for a keyboard in the android project or any other project or computer for that matter. At this point the limited memory will be the limiting factor in its capabilities. The subsystem at this point does not contain any reasoning capabilities due to the severe memory limitations. These limitations will all be swept away when we have the KISS OS running on the android system. We then will be able to bring higher brain functions into play as I have done in some of my military projects. We must build one subsystem at a time so you must be patient as I document each step in subsequent articles. I will be providing the **source code** on my website so that you can become intimately familiar with this new approach. You may wish to modify it and even write your own waveform pattern extraction algorithms. As you add more waveform description parameters you will find that the speech recognition specificity becomes better. However this will gobble up memory in a hurry and remember the 32K byte limit imposed by the propeller. These limitations will be removed in future articles. This will be a good example of the balance that must be achieved between “SENSITIVITY & SPECIFICITY” in any research project of this type.

What's Next ?

In the next article we will be adding one more Propeller board and a minimum of one expanded memory board. This will be used to implement the KISS Operating system and the vocabulary and all neuronal elements will be in multiterabyte virtual memory. This will support an almost unlimited vocabulary as well as all other higher “Android Brain” functions to be added in later articles. You must understand this relatively simple subsystem if you hope to understand the much more complex higher brain functions which will be covered in future articles. DO NOT BE AFRAID TO EXPERIMENT WITH THE CODE just keep a backup handy. You might wish to exchange ideas with each other on my “FACE BOOK “ site as well. Please understand that my goal is to bring all of you into this exciting new realm of “MACHINE NEUROSCIENCE”. I have expenses to pay as I do this work and therefore must charge a small fee for the [source code](#) (\$20.00) which can be downloaded from my website. If I could I would not charge anything but as with most of you I am not wealthy and must try to at least break even

December 2010

James O. Gouge, Ph.D in Computer Science,
Sally B Gouge M.S in Microbiology

which so far has not happened.