

VIRTUAL BRAIN USING IOT

A Project Report Phase – II

Submitted in partial fulfillment of the requirements for the award of
Bachelor of Engineering degree in Electronics and Communication Engineering

By

JAYAKEERTHANA K (39130182)

KAVIYA K (39130217)



**DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING
SCHOOL OF ELECTRICAL AND ELECTRONICS**

SATHYABAMA

**INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)**

Accredited with Grade “A” by NAAC

JEPPIAAR NAGAR, RAJIV GANDHI SALAI, CHENNAI - 600 119

APRIL - 2023



SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)

Accredited with "A" grade by NAAC
Jeppiaar Nagar, Rajiv Gandhi Salai, Chennai – 600 119
www.sathyabama.ac.in

DEPARTMENT OF ELECTRONICS AND COMMUNICAITON ENGINEERING

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the bonafide work of JAYAKEERTHANA K(39130182), KAVIYA K(39130217) who carried out the project entitled "**VIRTUAL BRAIN USING IoT**" under our supervision from November 2022 to April 2023.

Internal Guide

Dr. T . VINO, M.E., Ph.D.,

Head of the Department

Dr. T. RAVI, M.E., Ph.D.,

Submitted for Viva voce Examination held on 23.4.2023

Internal Examiner

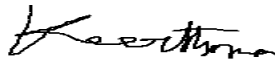
External Examiner

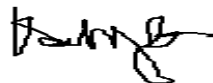
DECLARATION

We JAYA KEERTHANA K (39130182), KAVIYA .K (39130217) hereby declare that the project entitled “**VIRTUAL BRAIN USING IoT**” done by us under the guidance of **Dr. T . VINO, M.E., Ph.D.**, Associate Professor, Department of Electronics and Communication Engineering, Sathyabama Institute of Science and Technology is submitted in partial fulfillment of the requirements for the award of Bachelor of Engineering degree in Electronics and Communication Engineering.

DATE: 23.4.2023

PLACE: Chennai

1. 

2. 

SIGNATURE OF THE CANDIDATES

ACKNOWLEDGEMENT

We are pleased to acknowledge our sincere thanks to **Board of Management** of **SATHYABAMA** for their kind encouragement in doing this project and for completing it successfully. We are grateful to them.

We convey our thanks to **Dr. N. M. Nandhitha, M.E., Ph.D., Professor & Dean, School of Electrical and Electronics** and **Dr. T. Ravi, Ph.D., Professor & Head, Department of Electronics and Communication Engineering** for providing us necessary support and details at the right time during the progressive reviews.

We would like to express our sincere and deep sense of gratitude to our **Project Guide Dr. T . VINO, M.E., Ph.D.**, for her valuable guidance, suggestions and constant encouragement paved way to do the project work.

We wish to express our thanks to all Teaching and Non-teaching staff members of the **Department of Electronics and Communication Engineering** who were helpful in many ways for the completion of project.

ABSTRACT

This project 'Virtual brain using IoT' is a rapidly growing field that is leading to the development of real-time Brain Computer Interfaces (BCIs) that are low-cost and efficient. The hardware improvements that increase the capability of virtual brain analysis and wearable sensors have paved the way for new software frameworks for developers to use in building applications that combine BCI and IoT. These advancements have opened up the possibility of using multiple sensory pathways to facilitate communication with users' brains while handling large amounts of data. This intersection of research paths is accelerating progress in both fields and driving the need for an energy-efficient infrastructure to support the wider local bandwidth demands of mobile cloud networks. In this paper, a survey was conducted on BCI in IoT from various perspectives, including Electroencephalogram (EEG)-based BCI models, machine learning, and currently active platforms. The survey results highlight three significant development trends in BCI, which are EEG, IoT, and cloud computing. The EEG-based BCI models involve the use of EEG signals to identify a person's intention or state of mind. The EEG signals are measured using electrodes that are placed on the scalp and are used to interpret the person's thoughts or actions. This technology has many potential applications, including helping people with disabilities to communicate or control prosthetic devices. IoT, or the Internet of Things, involves connecting everyday objects to the internet to facilitate communication between them. BCI technology can be used to control IoT devices, enabling people to control their environment or appliances using their thoughts. For example, a person could turn on the lights in their home by simply thinking about it. Cloud computing involves the use of remote servers to store, manage, and process data. BCI technology can benefit from cloud computing by allowing for real-time data analysis and processing, which is crucial for accurate and efficient brain-computer communication.

TABLE OF CONTENTS

CHAPTER No.	TITLE	PAGE No.
	ABSTRACT	v
	LIST OF FIGURES	viii
1	INTRODUCTION	1
2	LITERATURE SURVEY	6
	2.1 Inference from Literature survey	6
	2.2 Open problems in existing system	9
3	AIM AND SCOPE OF THE PROJECT	11
	3.1 Aim of the project	11
	3.2 Scope of the project	11
4	SYSTEM DESIGN	13
	4.1 Existing System	13
	4.1 Proposed System	15
	4.2 Difference between Existing and Proposed system	18
5	REQUIREMENT ANALYSIS	20
	5.1 Hardware Requirements	20
	5.2 Software Requirements	23
	5.3 Components Descriptions	26
	5.4 System Architecture	28
	5.5 System Testing	30
6	RESULT AND DISCUSSION	33
7	SUMMARY AND CONCLUSION	36

7.1	Conclusion	36
7.2	Future work	37
	REFERENCES	38
	APPENDIX	
A.	SOURCE CODE	39
B.	SCREENSHOTS	47

LIST OF FIGURES

FIGURE. No.	FIGURE NAME	PAGE No.
Fig 1.1	Virtual brain	2
Fig 1.2	IOT cloud system	4
Fig 4.1	Block diagram for existing system	13
Fig 4.2	Virtual brain using BCI	14
Fig 5.1	EEG Sensor	16
Fig 5.2	TFT Screen	20
Fig 5.3	Internet of Things	21
Fig 5.4	Arduino UNO	22
Fig 5.5	Block diagram	22
Fig 5.6	Circuit diagram	26
Fig 5.7	Mail and storage	26
Fig 5.8	System Architecture	27
Fig 5.9	Screenshot of working model	28

CHAPTER 1

INTRODUCTION

Top on the list of the US Government's BRAIN Initiative is the ability to map the human brain at different scales with improved throughput and resolutions. A complete picture of the brain structure will provide new insights into how the human brain functions and may facilitate new treatments and drug discovery for brain disorders. Recent advances in intact brain imaging, such as the CLARITY and MAP (Magnified Analysis of the Proteome) tissue clearing techniques, make it possible to collect large volumetric images of brain tissue at cellular and sub-cellular resolutions. The high throughput and high resolution brain imagery, however, poses a challenge for efficient processing and analysis. We have developed an automated dense axonal fiber tracing pipeline that can track long-range fibers and construct 3D connectivity maps, which include not only the vertices and edges of a network graph, but also the 3D location information associated with each fiber track. In addition to typical graph analysis, we are interested in identifying long-range neuron fiber connections and fiber crossings, both of which could reveal informative patterns when analysed at cellular resolution and at long range (≥ 1 mm). Brain graphs offer a framework to represent the structural or functional topology at multiple levels. A number of software tools exist for analysis topology of brain networks using graph theory. They primarily focus on studying the correlations of anatomically separated brain regions. Few are designed for high throughput dense and long-range neuron analysis at the cellular level, which is critical for understanding brain circuits and for comparing healthy and diseased brains. High throughput and low latency analysis of brain data will require high speed databases and programming interfaces amenable to large scale graph analytics. In this paper, we present an approach that combines Accumulate, D4M, and parallel programming to manage the largescale brain connectivity graphs in a high performance computing environment. The Apache Accumulate database is based on Google's Big table database and allows for fast ingestion and extraction along with the capacity to hold largescale data. The Dynamic Distributed Dimensional Data Model (D4M) provides a mathematical framework, based on the mathematics of associative arrays that has been applied to diverse domains such as cyber security bioinformatics, free text,

and social media data. D4M can also be used to perform linear algebraic operations inside a database. D4M works seamlessly with the pMatlab parallel computing environment that allows users of MATLAB/Octave programming languages to use a single-program-multiple data (SPMD) parallel programming model. PMatlab sits on top of a message passing interface library such as Mat lab MPI or bc MPI and has been used for a variety of parallel computing applications. For our application, the D4M-pMatlab connection is used to leverage parallel computing techniques in addition to the cloud-based big data techniques provided by D4M. In this article, we extend the D4M-Accumulo interface to neural connectivity graphs computed from axon fibres imagery.



Fig 1.1 Virtual brain

This approach makes it possible to extract network vertices and edges with low latency, and perform fast and meaningful graph analytics by applying simple matrix operations. We begin with a description of the proposed platform and the backend technologies including Accumulo, D4M, and binary tree encoding, followed by experimental results and performance benchmarking analysis using 250 gigabytes (GB) of mouse neuron data. The functional organization of the brain is characterized by segregation and integration of information being processed. A central paradigm in modern neuroscience is that anatomical and functional connections between brain regions are organized in a way such that information processing is near optimal. Functional interactions seem to be provided by synchronized activity, both locally and between distant brain regions. Brain networks thus consist of spatially distributed but functionally connected regions that process information. Brain connectivity analysis rests upon three different but related forms of connectivity. (i) Anatomical connectivity (AC), also called structural connectivity, which forms the connectome through synaptic contacts between neighbouring

neurons or fibre tracks connecting neuron pools in spatially distant brain regions. The whole set of such fiber tracks in the brain is called white matter. On short time scales (sec, min), anatomical connections are quite persistent and stable, while for longer time spans substantial plasticity may be observed.(ii)Functional connectivity (FC) which is defined as the temporal dependency of neuronal activation patterns of anatomically separated brain regions. It reflects statistical dependencies between distinct and distant regions of information processing neuronal populations. Hence, it is basically a statistical concept which relies on such statistical measures as correlation, covariance, spectral coherence, or phase locking. Statistical dependencies are highly time dependent and fluctuate on multiple time scales ranging from milliseconds to seconds.(iii)Effective connectivity (EC) describes the influence one neuronal system exerts upon another, thus reflecting causal interactions between activated brain areas. It combines structural and effective connectivity into a wiring diagram which reflects directional effects within a neuronal network. Causality can be inferred from network perturbations or time series analysis (TSA). Techniques based on network perturbations generally need structural information as input, while TSA-based techniques, like Granger causality, may be considered model-free.

A synthesis of the latter two concepts of connectivity, mainly applied to and deduced from functional neuroimaging modalities, has been provided by Friston. Functional and effective connectivity can originate, for example, from multielectrode array recordings. Both refer to abstract concepts with no immediate connection to anatomical connectivity which physically mediates such correlations. However, in recent years efforts have been undertaken to bridge the gap between these types of connectivity analysis, put forward mainly by techniques such as diffusion tensor imaging (DTI) which allow us to track fibers which form the neuronal basis for functional correlations. A recent review details biophysical concepts used to model such connectivity. In 2003, Horwitz questioned the concepts of functional and effective connectivity. He argued that these notions are derived from different functional imaging modalities like functional magnetic resonance imaging (fMRI) or positron emission tomography (PET). The concept of connectivity designates the strength of interactions, whether direct or indirect, between different brain areas which locally process information. However, functional and effective connectivity are

derived from quantities computed on different spatial and temporal scales, using different definitions and employing various algorithms. As long as the relation of such abstract concepts to the underlying structural connectivity between areas is not understood, comparisons across studies have to be taken with great caution. However, notice that there exists ample evidence that both concepts can be derived from the same imaging modality as well. Still connectivity analysis studies created the notion of complex brain networks characterized by densely connected nodes of information processing which are distant in anatomical space and only sparsely connected via long-range connections between different functionally interacting brain regions. These network topologies reflect two basic principles underlying information processing in the brain: functional segregation and functional integration. Experimental evidence for such network topologies mainly comes from neuroimaging techniques (EEG, MEG, fMRI, PET, and SPECT) and neuroanatomical methods. Signal transmission between distinct brain regions requires connecting fiber tracts, thus forming the structural basis of the human connectome. Diffusion-weighted magnetic resonance imaging and its variant called diffusion tensor imaging (DTI) represent the most promising approaches for fiber tracking . While the former maps the diffusive motion of water molecules in the tissue returning back a single grey value per voxel only, the latter also considers the direction of diffusive motion, hence determining the second-order symmetric diffusion tensor, instead.

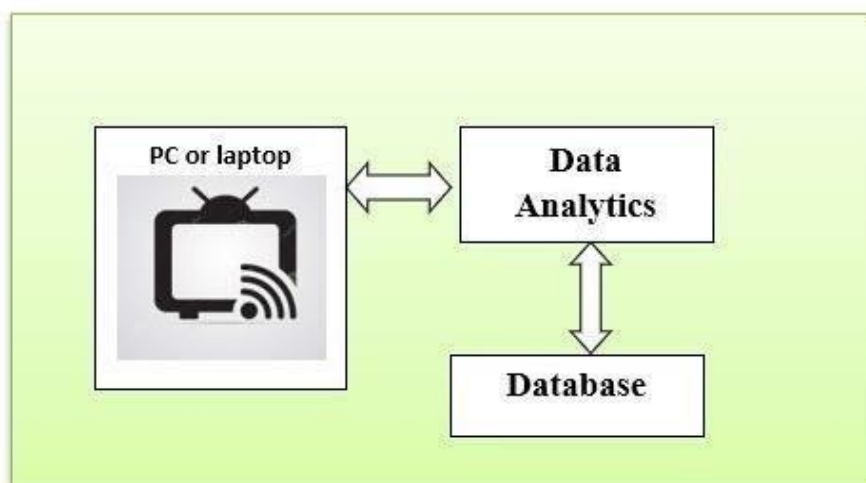


Fig 1.2 IOT cloud system

A severe limitation of these methods, however, is their low spatial resolution. The latter can be overcome with 3D polarized light imaging (3D-PLI) where the 3-dimensional course of fibers can be traced with a spatial resolution down to 100 μ m. Hence, 3D-PLI provides an independent evaluation of results obtained with DTI. Brain connectivity can be quantified by encoding neighbourhood relations into a connectivity matrix, whose rows and columns correspond to different brain regions. This representation lends itself to be mapped to a graphical model which provides means to quantify different topological aspects of the connectome. Graphical models represent a versatile mathematical framework for a generic study of pairwise relations between interacting brain regions. Recent years have witnessed an exponential growth of studies related to the application of graph theory to unravel characteristic features of structural, functional, and effective connectivity from neuroimaging investigations. The most striking discovery reveals small-world properties of complex brain networks which they share with many other complex systems. A small-world topology allows a high efficiency at different spatial and temporal scales with a very low wiring and energy cost. These recent discoveries indeed may indicate that the connectome is just one example of a more general universality class of complex systems observed in nature. The survey is organized in the following way. First, some recent studies and reviews about experimental studies of functional connectivity are reported. This is not meant to be comprehensive, rather it should illustrate some prototypical studies in this field. Next, recent computational methods dealing with functional connectivity and some illustrative applications are collected. This is followed by a short survey of recent studies on effective connectivity. Finally, the important concept of graphical models applied to such complex brain networks as well as some applications to connectivity analysis is discussed.

CHAPTER 2

LITERATURE SURVEY

Inference from literature survey

In 2014, the National Institutes of Health (NIH) began funding an ambitious research program, the Brain Research through Advancing Innovative Neuro technologies (BRAIN) Initiative, with the singular focus of advancing our understanding of brain circuits through development and application of breakthrough neuro technologies. As we approach the halfway mark of this 10-year effort aimed at revolutionizing our understanding of information processing in the human brain, it is timely to review the progress and the future trajectory of BRAIN Initiative research. Throughout history, new tools have driven scientific revolutions in multiple disciplines [1]. Unraveling how the human brain processes information and supports a diverse spectrum of behaviours requires such a scientific revolution and an arsenal of next generation tools to be deployed by neuroscientists across the basic, translational, and clinical landscapes. Capturing the full record of neural activity or what has been called a “brain activity map” across spatial and temporal domains is a daunting technological challenge that needs to be met to achieve a more complete understanding of fundamental and pathological brain processes [2]. Launched on April 2, 2013, the key goal of the Brain Research through Advancing Innovative Neuro technologies (BRAIN) Initiative is to develop innovative technologies to interrogate how the brain’s cells and circuits interact at the speed of thought and, ultimately, to reveal the complex links between brain function and behavior. Because brain architecture stretches from the scale of molecular interactions at the trillions of synapses to the billions of cell bodies that connect to form local networks that then integrate across multiple brain regions, the challenges are extraordinary. In addition to these spatial scales, there are temporal scales, as brain circuits are not static but continually change as a result of neural activity, developmental stage, and aging. Despite this complexity, the technologies emerging from the BRAIN Initiative are opening new doors to decipher how the brain records, processes, uses, stores, and retrieves vast quantities of information. They have the potential to facilitate a quantum leap toward understanding brain function and its disruption in disease, making circuit abnormalities the basis of diagnostics and the normalization

of circuit function the target of future intervention in neuro/mental/substance-abuse disorders.

With potential relevance for brain-mapping work, hydrogel-based structures can now be built from within biological tissue to allow subsequent removal of lipids without mechanical disassembly of the tissue. This process creates a tissue-hydrogel hybrid that is physically stable, that preserves fine structure, proteins and nucleic acids, and that is permeable to both visible-spectrum photons and exogenous macromolecules. Here we highlight relevant challenges and opportunities of this approach, especially with regard to integration with complementary methodologies for brain-mapping studies.

T. Ku, J. Swaney, J.-Y. Park, A. Albanese, E. Murray, J. H. Cho, Y.-G. Park, V. Mangena, J. Chen, and K. Chung. The biology of multicellular organisms is coordinated across multiple size scales, from the sub Nano scale of molecules to the macroscale, tissue-wide interconnectivity of cell populations. Here we introduce a method for super-resolution imaging of the multiscale organization of intact tissues. The method, called magnified analysis of the proteome (MAP), linearly expands entire organs fourfold while preserving their overall architecture and three-dimensional proteome organization. MAP is based on the observation that preventing crosslinking within and between endogenous proteins during hydrogel-tissue hybridization allows for natural expansion upon protein denaturation and dissociation. The expanded tissue preserves its protein content, its fine subcellular details, and its organ-scale intercellular connectivity. We use off-the-shelf antibodies for multiple rounds of immune labelling and imaging of a tissue's magnified proteome, and our experiments demonstrate a success rate of 82% (100/122 antibodies tested). We show that specimen size can be reversibly modulated to image both inter-regional connections and fine synaptic architectures in the mouse brain.

The human brain is a complex network. An important first step toward understanding the function of such a network is to map its elements and connections, to create a comprehensive structural description of the network architecture. This paper reviews current empirical efforts toward generating a network map of the human brain, the human connectome, and explores how the connectome can provide new

insights into the organization of the brain's structural connections and their role in shaping functional dynamics. Network studies of structural connectivity obtained from non-invasive neuroimaging have revealed a number of highly non-random network attributes, including high clustering and modularity combined with high efficiency and short path length. The combination of these attributes simultaneously promotes high specialization and high integration within a modular small-world architecture. Structural and functional networks share some of the same characteristics, although their relationship is complex and nonlinear. Future studies of the human connectome will greatly expand our knowledge of network topology and dynamics in the healthy, developing, aging, and diseased brain.

There is great interest in estimating brain "networks" from fMRI data. This is often attempted by identifying a set of functional "nodes" (e.g., spatial ROIs or ICA maps) and then conducting a connectivity analysis between the nodes, based on the fMRI time series associated with the nodes. Analysis methods range from very simple measures that consider just two nodes at a time (e.g., correlation between two nodes' time series) to sophisticated approaches that consider all nodes simultaneously and estimate one global network model (e.g., Bayes net models). Many different methods are being used in the literature, but almost none has been carefully validated or compared for use on fMRI time series data. In this work we generate rich, realistic simulated fMRI data for a wide range of underlying networks, experimental protocols and problematic confounds in the data, in order to compare different connectivity estimation approaches. Our results show that in general correlation-based approaches can be quite successful, methods based on higher-order statistics are less sensitive, and lag-based approaches perform very poorly. More specifically: there are several methods that can give high sensitivity to network connection detection on good quality fMRI data, in particular, partial correlation, regularised inverse covariance estimation and several Bayes net methods; however, accurate estimation of connection directionality is more difficult to achieve, though Patel's τ can be reasonably successful. With respect to the various confounds added to the data, the most striking result was that the use of functionally inaccurate ROIs (when defining the network nodes and extracting their associated time series) is extremely damaging to network estimation; hence, results derived

from inappropriate ROI definition (such as via structural atlases) should be regarded with great caution.

Open problems in existing system

The existing system of virtual brain is still in its early stages, and there are several open problems that need to be addressed to improve its efficiency and accuracy. Some of these open problems are:

1. **Data Collection and Analysis:** The virtual brain relies heavily on data collection and analysis from various sources, such as MRI and EEG. However, there are still limitations in terms of the quality and quantity of data that can be collected. The existing system needs to be improved to collect high-quality data from a large sample size to make the virtual brain more accurate.
2. **Scalability:** The current system of virtual brain is limited in terms of its scalability. The complexity of the model and the large amount of data required make it difficult to scale up the system to larger datasets. There is a need to develop more efficient algorithms and software to overcome this issue.
3. **Interpretation of Data:** The interpretation of data is a significant challenge in the virtual brain. The brain is a complex system, and it is difficult to interpret the data collected from it. There is a need for advanced machine learning algorithms that can help interpret the data more accurately.
4. **Integration of Multiple Data Sources:** The virtual brain relies on data from multiple sources, such as MRI, EEG, and fMRI. However, integrating these data sources is a complex task, and there is a need for more efficient algorithms that can combine these data sources accurately.
5. **Lack of Standardization:** There is a lack of standardization in terms of the data collection and analysis methods used in the virtual brain. This makes it difficult to compare results from different studies and models. There is a need to develop standard protocols and methodologies to ensure consistency across different studies.

6. Computational Complexity: The virtual brain is a computationally intensive task, and it requires a significant amount of computational resources to run. The existing system needs to be improved to reduce the computational complexity and make the system more efficient.
7. Clinical Relevance: The ultimate goal of the virtual brain is to improve our understanding of brain function and develop new treatments for brain disorders. However, there is a need to validate the virtual brain's clinical relevance by conducting more clinical trials and studies.

In conclusion, the virtual brain is an exciting field of research that has the potential to revolutionize our understanding of brain function and develop new treatments for brain disorders. However, there are still several open problems that need to be addressed to improve the accuracy, efficiency, and clinical relevance of the system. The development of advanced machine learning algorithms, standardization of data collection and analysis, and integration of multiple data sources are some of the areas where further research is required.

CHAPTER 3

AIM AND SCOPE OF THE PROJECT

Aim of the project

The aim of the above project is to describe a pipeline for analyzing large brain data in order to gain insights into brain function and potentially develop new treatments for brain disorders. The pipeline involves various technologies and platforms, including an imaging processing module for computing network connectivity graphs, accumulate and D4M for storing the parsed graphs, geo hashing-based binary tree encoding for sub volume indexing, and a data schema for representing nodes and links in the graph. The pipeline also involves graph analytics, such as the Breadth First Search algorithm, for traversing the graph and finding connected vertices. Overall, the goal is to provide a reference for future research in EEG-based BCI models, machine learning, and platforms related to brain analysis.

Scope of the project

The scope of the virtual brain project is to create a computer simulation that can replicate the complex workings of the human brain. The project aims to create a comprehensive model that can be used to study brain function and develop treatments for neurological disorders. The virtual brain project is a collaborative effort involving researchers from multiple fields, including neuroscience, computer science, and mathematics. The project uses advanced technology such as supercomputers, big data analytics, and artificial intelligence to create a realistic and accurate model of the brain.

The virtual brain model is expected to provide insights into the way the brain processes information, stores memories, and generates behavior. This information can be used to better understand neurological disorders such as Alzheimer's disease, Parkinson's disease, and epilepsy. By studying the virtual brain, researchers hope to develop new treatments and therapies for these disorders. In addition to medical applications, the virtual brain project has potential implications for artificial intelligence and robotics. The insights gained from studying the virtual brain could be used to develop more sophisticated artificial intelligence systems and

robots that can mimic human behavior. Overall, the virtual brain project represents a significant step forward in our understanding of the human brain and has the potential to revolutionize our approach to treating neurological disorders and developing advanced technology.

CHAPTER 4

SYSTEM DESIGN

Existing Method

BCIs are currently fairly inaccurate in terms of classifying neural activity. BCIs placed outside of the skull have a limited ability to read brain signals. They can be placed under the skull, but this requires pretty drastic surgery. Reading people's inner thoughts comes with a massive amount of ethical issues.

The existing system of virtual brain is based on Brain-Computer Interface (BCI) technology, which involves the use of sensors to detect and interpret brain activity, and then use this information to control external devices. The goal of this technology is to enable individuals with disabilities to communicate and interact with the world around them through the use of their thoughts alone.

The current state of BCI technology is still in its early stages, and there are a number of challenges that must be overcome before it can be widely adopted. One of the main challenges is that BCIs are currently fairly inaccurate in terms of classifying neural activity. This means that the system may misinterpret the user's intentions, leading to errors in the output. This can be particularly problematic in situations where accuracy is critical, such as when controlling a prosthetic limb.

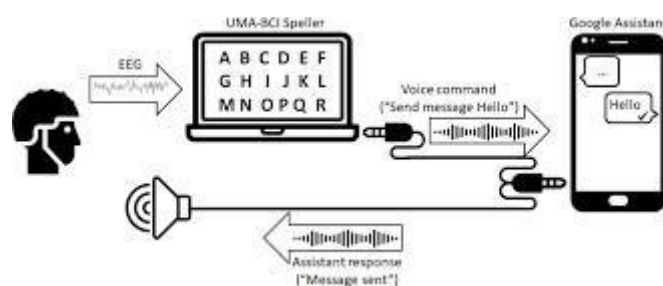


Fig 4.1 Block diagram for existing system

Another challenge with BCIs is that they can be difficult to use. This is because they require the user to focus their thoughts in a very specific way in order to generate the desired output. This can be tiring and frustrating, especially for users who may have limited mobility or cognitive function.

In addition, BCIs placed outside of the skull have a limited ability to read brain signals. This is because the skull acts as a barrier that can interfere with the signal. They can be placed under the skull, but this requires pretty drastic surgery. This can be risky and expensive, making it difficult to justify for many users.



Fig 4.2 Virtual brain using BCI

Another major challenge with the existing system of virtual brain is that reading people's inner thoughts comes with a massive amount of ethical issues. For example, if a BCI were able to read a person's thoughts without their consent, it could be used for nefarious purposes such as mind reading or even mind control. This could lead to serious violations of personal privacy and autonomy. Despite these challenges, there have been some notable successes in the development of BCI technology. For example, some researchers have been able to use BCIs to enable paralyzed patients to control robotic arms, allowing them to perform tasks such as grasping objects and feeding themselves. Other researchers have used BCIs to allow patients with locked-in syndrome to communicate with the outside world using a virtual keyboard or speech synthesizer.

Overall, the existing system of virtual brain is still in its infancy, and there is much work to be done before it can be widely adopted. However, there is a great deal of potential in this technology, and as researchers continue to develop and refine the underlying technology, it is likely that we will see more and more applications of BCI in a variety of different fields.

Proposed System

A complete picture of the brain structure will provide new insights into how the human brain functions and may facilitate new treatments and drug discovery for brain disorders. We investigate the existing BCI VR researches from three crucial aspects that are EEG-based BCI models machine learning, and platforms. The findings can be used as reference for future relevant researches. The pipeline starts with an imaging processing module, which computes network connectivity graphs that contain vertices, edges, and 3D coordinates of fiber tracks. The graphs are then parsed into an associative array format. Vertices and edges are saved into an accumulate database while 3D coordinates are saved to flat files. A real-time web-based 3D user interface displays the fiber tracks by querying data from accumulate and flat files.

A. Technologies and Platform Design

1) Imaging Pipeline: The first step of the pipeline is to divide the original large data into sub volumes and apply image processing algorithms in parallel using pMatlab. The main image processing algorithms include a convolutional neural network (CNN) for segmenting the axon fibers, and a number of morphological operations for extracting fiber tracks, from which a network graph of vertices, edges, and 3D coordinates of the fiber tracks are computed. Our CNN has three convolutional and two fully connected layers. The last layer uses sigmoid activation, and all other layers use rectified linear units (ReLU). In the graphs, vertices correspond to the start/end of the fiber tracks and edges represent the connections (fiber tracks) between the vertices. These parsed graphs are then converted into an associative array format, which is amenable for graph analysis.

2) accumulate and D4M: After the graphs are converted to an associative array format, they can be inserted into accumulate or saved into flat files. Based on our benchtop experiments, we determined that a hybrid approach that leverages both accumulate database and filesystem offers the optimal performance. We store the fiber graph vertices and edges in accumulate while saving the fiber track 3D coordinates in flat files. 3D coordinates are much denser data when compared to the vertices and edges, and would slow down the retrieval time significantly if stored in accumulate. In addition, since 3D coordinates are primarily used for morphology analysis and visualization, they are not accessed as frequently as vertices and edges. This hybrid configuration ensures low latency and high

bandwidth access to data as needed for analytics and visualization. 3) Geo hashing-based binary tree encoding for sub volume indexing: The geo hash was invented by Gustavo Niemeyer, and is a geocoding system based on a hierarchical spatial data structure which subdivides space into a grid that can be easily identified .

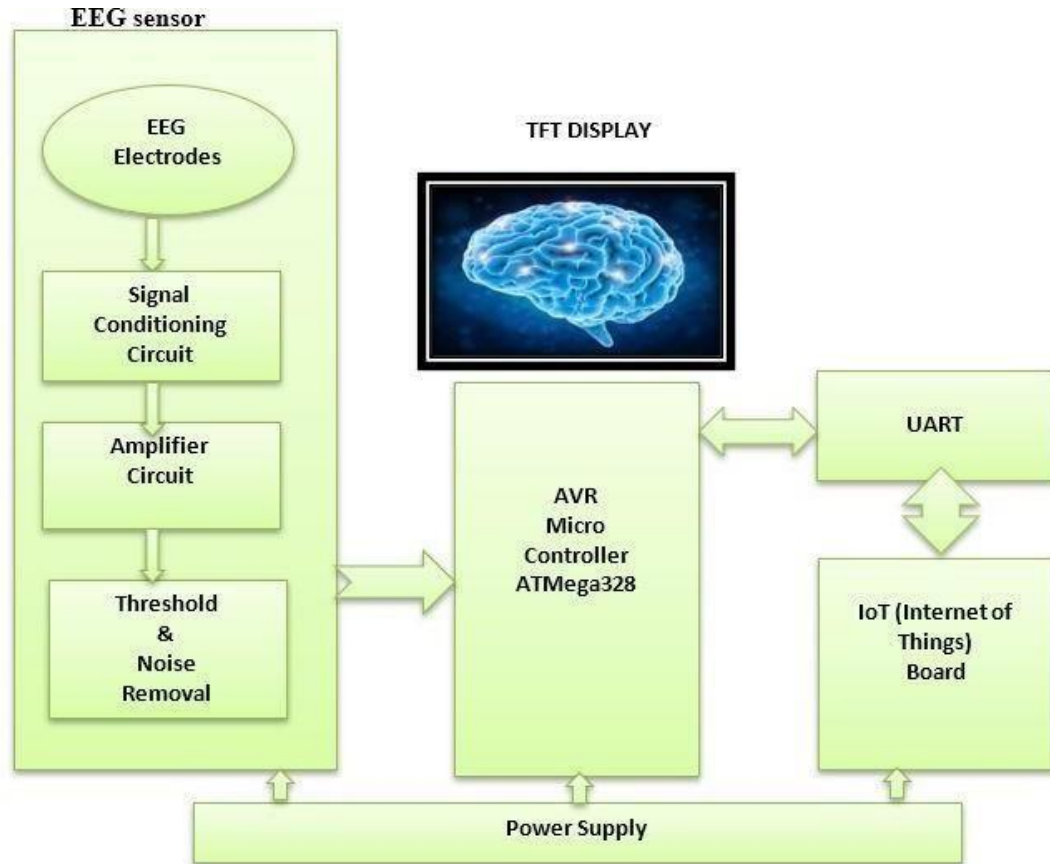


Fig 4.3 Virtual brain using IOT

When used in a database, the structure of geo hashed data has two advantages. First, data indexed by geo hash will include all the points for a given rectangular area in contiguous slices. This is especially useful in a database system such as accumulate where queries on a single index are much easier or faster than multiple-index queries. Second, this index structure can be used for a quick proximity search because the closest points are often among the closest geo hashes . One challenge presented by manipulating large brain data is that it is often difficult to drill down to any specific region of interest without having to go through lengthy and time consuming indexing process. To overcome this, we have designed a binary tree indexing schema based on geo hashing. This allows us to identify specific neuron fibers in a particular sub volume easily and quickly. Figure 3 is an example of the binary tree encoding scheme we designed. In this case, we use a string of 6 binary

code to represent the entire dataset, with each digit representing a binary division. This results in 3 zoom levels. To illustrate the binary division, the color of each digit corresponds to the color of a vertical or horizontal division line. Depending on the level of detail needed, binary codes can be easily made more specific by increasing the number of digits, resulting in more layers and finer sub volumes. This scalable schema is particularly useful for visualizing large datasets because it allows for easy query of data segments at multiple zoom levels, thus creating a Google Map like view.

4) Data Schema: In the following representation, nodes refer to graph vertices and links refers to edges connecting nodes in the graph. The node associative array (Table II) contains the information for identifying a vertex globally. The node ID is the locally unique number in each sub volume for each node. The sub volume number is the binary code where the node is located. End Yes is 0 or 1 depending on whether or not the node is an end node, with 1 meaning it is the start or end of a fiber track, and 0 meaning it is a junction that connects to multiple other nodes. End Node is the node ID of the other nodes that this node is connected to. Links is the link IDs of the links that are connected to this node. Coro is the linear index of the coordinates of the node within the sub volume. The link associative array (Table IV) contains the information about links except the 3D coordinates, which are saved in flat files in order to maintain fast query speed. The link ID and sub vol Num are the same as node ID and sub vol Num in the node associative array. node1 is the node ID of one end of the link, while node2 is the node ID of the other. length contains the number of coordinates in the link.

B. Graph Analytics In graph analytics, a commonly used algorithm is Breadth First Search (BFS). The BFS algorithm is used to traverse a graph data structure and find vertices connected to one another. In our application, BFS intuitively tells us which neurons are connected to other neurons. Using the associative array representation described previously, we can apply BFS to a node (or set of nodes) directly by multiplying two associative arrays . To apply BFS on a node, we first extract the sub volume of interest from accumulate to create an incidence matrix E. Then, the desired node is added to an associative array and multiplied by the incidence matrix of the sub volume's corresponding subgraph. This results in an associative array that has information about the nodes connected to the input nodes. The desired link information about the nodes is then extracted. This process can be repeated to find all nodes connected to this resultant.

Difference between existing and proposed system

The existing system of virtual brain is limited in many ways and has several drawbacks that need to be addressed. The proposed system of virtual brain aims to overcome these limitations and provide a more comprehensive and advanced solution.

One of the major differences between the existing and proposed system is the level of detail and complexity. The existing system provides a basic simulation of the brain with a limited number of neurons and connections. In contrast, the proposed system aims to simulate the brain at a much higher level of detail and complexity, including more accurate modeling of the different regions of the brain, the neural circuits and synapses, and the interactions between them.

Another difference between the existing and proposed system is the availability of real-time interaction with the simulation. The existing system is mostly limited to a static simulation that can be viewed and analyzed after the simulation has run. The proposed system, on the other hand, aims to provide real-time interaction with the simulation, allowing researchers and scientists to modify the parameters and observe the effects in real-time.

The proposed system also aims to improve the accuracy and reliability of the simulation. The existing system may have limitations in accurately modeling the behavior of the neurons and their interactions, leading to potential inaccuracies and inconsistencies in the simulation. The proposed system aims to address these issues by utilizing more advanced computational techniques and models to simulate the brain's behavior more accurately.

The proposed system also aims to provide more advanced visualization and analysis tools to better understand the simulation results. The existing system may provide basic visualization tools that allow researchers to view the simulation results, but the proposed system aims to provide more advanced and interactive visualization tools that allow researchers to explore the simulation results in greater detail.

The proposed system also aims to be more accessible and user-friendly than the existing system. The existing system may require specialized knowledge and skills to operate and interpret the results, while the proposed system aims to provide a more intuitive and user-friendly interface that allows researchers and scientists with a range of backgrounds and expertise to use the simulation.

Finally, the proposed system aims to be more scalable and flexible than the existing system. The existing system may have limitations in terms of the size and complexity of the simulation that can be run, while the proposed system aims to be more scalable and flexible, allowing researchers to simulate larger and more complex systems with ease.

In summary, the proposed system of virtual brain aims to overcome many of the limitations of the existing system by providing a more comprehensive and advanced solution that includes higher levels of detail and complexity, real-time interaction, improved accuracy and reliability, advanced visualization and analysis tools, user-friendliness, and scalability and flexibility.

CHAPTER 5

REQUIREMENT ANALYSIS

Hardware Requirements

EEG sensor : EEG Sensor is the world's least expensive research-grade EEG headset available. Designed for interface with mobile devices (iOS and Android) and desktop (Win and Mac), it can be used with a wide variety of games, brain training and education applications. Its clear brainwave signal is based on the TGAM, the bio-sensor chipset that revolutionized an industry. It is an excellent introduction into the world of brain-computer interface (BCI). The device consists of a headset, an ear-clip, and a sensor arm. The headset's reference and ground electrodes are on the ear clip and the EEG electrode is on the sensor arm, resting on the forehead above the eye (FP1 position). It uses a single AAA battery with 8 hours of battery life.



Fig 5.1 EEG Sensor

Electroencephalogram (EEG) sensors require conductive gel to ensure low-impedance electrical contact between the sensor and skin. We present a gel-free, non-contact EEG sensor with on-board electrode those capacitive couples to the skin. Active shielding of the high-impedance input significantly reduces noise pickup, and reduces variations in gain. The measured input-referred noise, over 1-100 Hz frequency range, is $2\mu\text{vrms}$ at 0.2mm sensor distance, and $17\mu\text{vrms}$ at 3.2mm distance. Experiments coupling the sensor to human scalp through hair and to chest through clothing produce clear EEG recorded Signals.

Arduino TFT touch screen : The Arduino TFT screen is a backlit TFT LCD screen with a micro SD card slot in the back. You can draw text, images, and shapes to the screen with the TFT library. The screen's pin layout is designed to easily fit into the socket of an Arduino Esplora and Arduino Robot, but it can be used with any Arduino board. The Arduino TFT library extends the Adafruit GFX, and Adafruit ST7735 libraries that it is based on. The GFX library is responsible for the drawing routines, while the ST7735 library is specific to the screen on the Arduino screen. The Arduino specific additions were designed to work as similarly to the Processing API as possible.



Fig 5.2 TFT Screen

TFT Touch Screen is an Arduino / Arduino Mega compatible multicolored TFT display with a 4-wire resistive touch screen. It includes an Arduino shield compatible footprint for attachment. The TFT driver is based on professional Driver IC and with 8 bit data and 4 bit control interface. The Arduino TFT screen is a backlit touch screen with headers. You can draw text, images, and shapes to the screen with the TFT library. There is an onboard micro-SD card slot on the back of the screen that can, among other things, store bitmap images for the screen to display. The TFT library interfaces with the screen's controller through SPI when using the TFT library.

IoT-Internet of things : The Internet of Things (IoT) is an environment in which objects, animals or people are provided with unique identifiers and the ability to transfer data over a network without requiring human-to-human or human-to-computer interaction. IoT has evolved from the convergence of wireless technologies, micro-electromechanical systems (MEMS) and the Internet. The concept may also be referred to as the Internet of Everything. A thing, in the Internet of Things, can be a person with a heart monitor implant, a farm animal with a biochip

transponder, an automobile that has built-in sensors to alert the driver when tire pressure is low -- or any other natural or man-made object that can be assigned an IP address and provided with the ability to transfer data over a network.

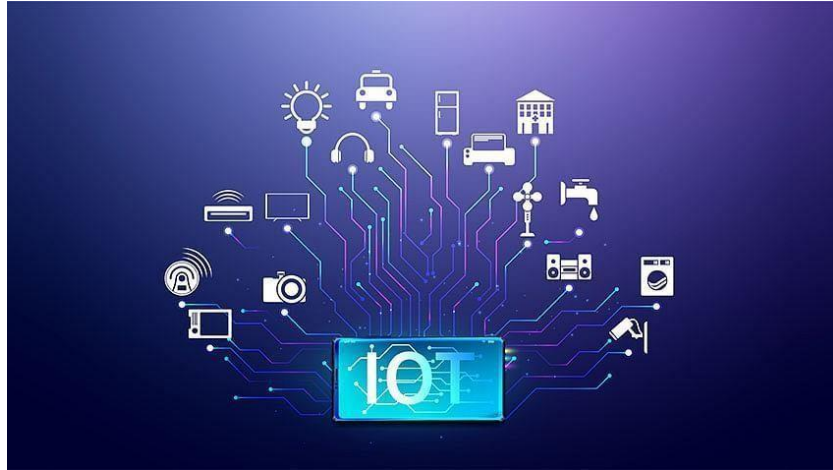


Fig 5.3 Internet of Things

Arduino UNO : Arduino is an open-source project that created microcontroller-based kits for building digital devices and interactive objects that can sense and control physical devices. The project is based on microcontroller board designs, produced by several vendors, using various microcontrollers. These systems provide sets of digital and analog input/output (I/O) pins that can interface to various expansion boards (termed shields) and other circuits. The boards feature serial communication interfaces, including Universal Serial Bus (USB) on some models, for loading programs from personal computers.



Fig 5.4 Arduino UNO

For programming the microcontrollers, the Arduino project provides an integrated development environment (IDE) based on a programming language named Processing, which also supports the languages C and C++. Arduino Uno is a microcontroller board based on the ATmega328P. It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz quartz crystal, a USB connection, a power jack, an ICSP header and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter. Arduino Uno has a number of facilities for communicating with a computer, another Arduino board, or other microcontrollers.

Software Requirements

MPLAB IDE : MPLAB IDE is an Integrated Development Environment (IDE) for developing and debugging embedded software on Microchip's microcontrollers and digital signal controllers. It is a comprehensive software package that provides everything needed to develop embedded applications, including a code editor, project management tools, a compiler, a debugger, and a simulator.

MPLAB IDE is designed to work with Microchip's PIC (Peripheral Interface Controller) microcontrollers and dsPIC (Digital Signal Controller) digital signal controllers. These microcontrollers are used in a variety of applications, including automotive, consumer electronics, medical devices, and industrial control systems. The IDE provides a user-friendly interface for creating, managing, and debugging embedded applications. It includes a code editor with features like syntax highlighting, code completion, and code navigation, which make writing and managing code easier. The IDE also provides project management tools that allow developers to manage all aspects of their projects, including adding and removing files, managing build configurations, and generating code documentation.

MPLAB IDE includes a compiler that can convert source code written in C or Assembly language into machine code that can be executed on Microchip's microcontrollers. The compiler optimizes the code for the specific microcontroller being used, which can result in faster and more efficient code.

The IDE also includes a debugger that can be used to step through code, set breakpoints, and monitor variables in real-time. This allows developers to quickly identify and fix bugs in their code. The debugger can also be used with a simulator,

which allows developers to test their code without having to connect to an actual microcontroller.

In addition to the core features, MPLAB IDE also includes a variety of plugins and add-ons that extend its functionality. For example, there are plugins for integrating third-party compilers and debuggers, as well as plugins for creating graphical user interfaces and for debugging communication protocols.

Overall, MPLAB IDE is a powerful and versatile tool for developing and debugging embedded software on Microchip's microcontrollers and digital signal controllers. Its comprehensive feature set and user-friendly interface make it an essential tool for embedded software developers.

Embedded C Language : Embedded C language is a programming language that is used to develop software for embedded systems, which are computer systems that are built into devices and machines. Embedded systems are designed to perform specific tasks and are found in a wide range of products, such as consumer electronics, medical devices, and automotive systems.

The C language was originally developed for programming large computer systems, but it was later adapted for use in embedded systems. Embedded C is a variant of the C language that includes specific features and libraries that are optimized for use in embedded systems.

One of the main features of embedded C is the ability to directly access hardware resources, such as memory-mapped I/O, which allows the programmer to interact directly with the hardware components of the embedded system. This makes it possible to create software that is highly optimized for the specific hardware platform, resulting in faster and more efficient code.

Embedded C also includes a number of features that are designed to optimize code size and performance. For example, it supports bit-level operations, which are used to manipulate individual bits in data registers. It also supports inline assembly language, which allows the programmer to insert machine code directly into the C code to improve performance.

In addition to these features, embedded C includes a number of libraries that are specific to embedded systems, such as the standard peripheral library (SPL) for microcontrollers. These libraries provide a set of functions for interacting with specific hardware components, such as timers, UARTs, and ADCs.

Embedded C is widely used in the development of embedded systems, and there are a number of popular compilers and development tools available for the language. Some of the most popular compilers include Keil, IAR Systems, and GCC. Overall, embedded C is a powerful language that is specifically designed for developing software for embedded systems. Its ability to directly access hardware resources and its support for optimizing code size and performance make it an ideal choice for developing efficient and reliable embedded software.

Hi-Tech Compiler : The Hi-Tech compiler is a software tool used for compiling and translating high-level programming languages into machine code that can be understood by microcontrollers and other embedded systems. It is specifically designed for use with Microchip microcontrollers and is commonly used in the development of embedded systems.

The Hi-Tech compiler supports the C programming language and provides many advanced features for optimizing code, reducing memory usage, and improving overall system performance. It also includes a wide range of libraries and functions that are specifically designed for use with Microchip microcontrollers, making it easier for developers to write code that can be executed efficiently on these systems.

One of the key advantages of using the Hi-Tech compiler is its ability to generate highly optimized code that is tailored to the specific hardware architecture of the target microcontroller. This helps to ensure that the system runs smoothly and efficiently, with minimal resource usage.

In addition to its advanced optimization features, the Hi-Tech compiler also provides a range of debugging and diagnostic tools that can help developers to identify and fix errors in their code more easily. These tools include features such as real-time debugging, memory profiling, and trace logging, which can be extremely helpful when working with complex embedded systems.

Overall, the Hi-Tech compiler is an essential tool for developers working in the field of embedded systems, particularly those who are working with Microchip microcontrollers. It provides a range of advanced features and optimization capabilities, making it easier to develop high-performance, reliable systems that can run efficiently on even the most resource-constrained hardware platforms.

Components Description

The input to the circuit is applied from the regulated power supply. The a.c. input i.e., 230V from the mains supply is step down by the transformer to 12V and is fed to a rectifier. The output obtained from the rectifier is a pulsating d.c voltage. So in order to get a pure d.c voltage, the output voltage from the rectifier is fed to a filter to remove any a.c components present even after rectification. Now, this voltage is given to a voltage regulator to obtain a pure constant dc voltage.

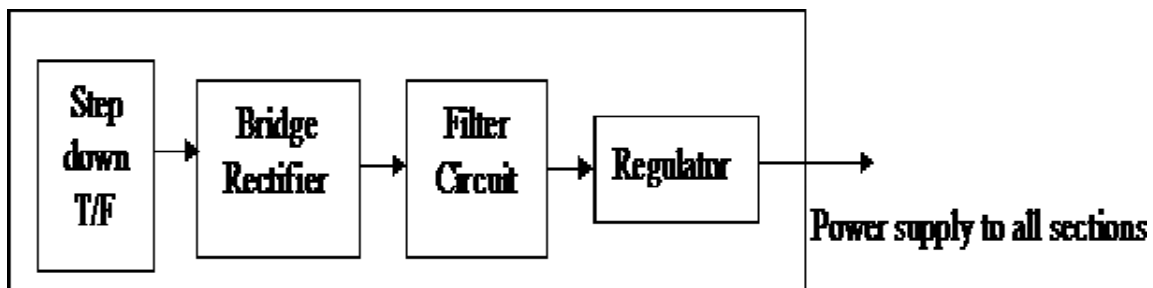


Fig 5.5 Block diagram

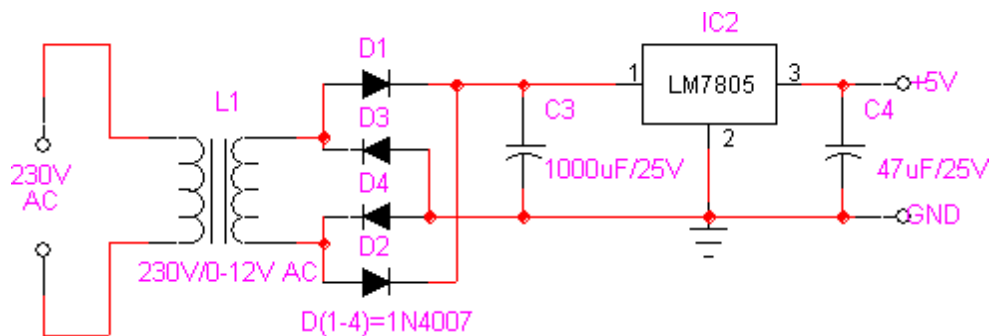


Fig 5.6 Circuit diagram

Transformer : Usually, DC voltages are required to operate various electronic equipment and these voltages are 5V, 9V or 12V. But these voltages cannot be obtained directly. Thus the a.c input available at the mains supply i.e., 230V is to be brought down to the required voltage level. This is done by a transformer. Thus, a step down transformer is employed to decrease the voltage to a required level.

Rectifier: The output from the transformer is fed to the rectifier. It converts A.C. into pulsating D.C. The rectifier may be a half wave or a full wave rectifier. In this project, a bridge rectifier is used because of its merits like good stability and full wave rectification.

Filter: Capacitive filter is used in this project. It removes the ripples from the output of rectifier and smoothens the D.C. Output received from this filter is constant until the mains voltage and load is maintained constant. However, if either of the two is varied, D.C. voltage received at this point changes. Therefore a regulator is applied at the output stage.

Voltage regulator: As the name itself implies, it regulates the input applied to it. A voltage regulator is an electrical regulator designed to automatically maintain a constant voltage level. In this project, power supply of 5V and 12V are required. In order to obtain these voltage levels, 7805 and 7812 voltage regulators are to be used. The first number 78 represents positive supply and the numbers 05, 12 represent the required output voltage levels.

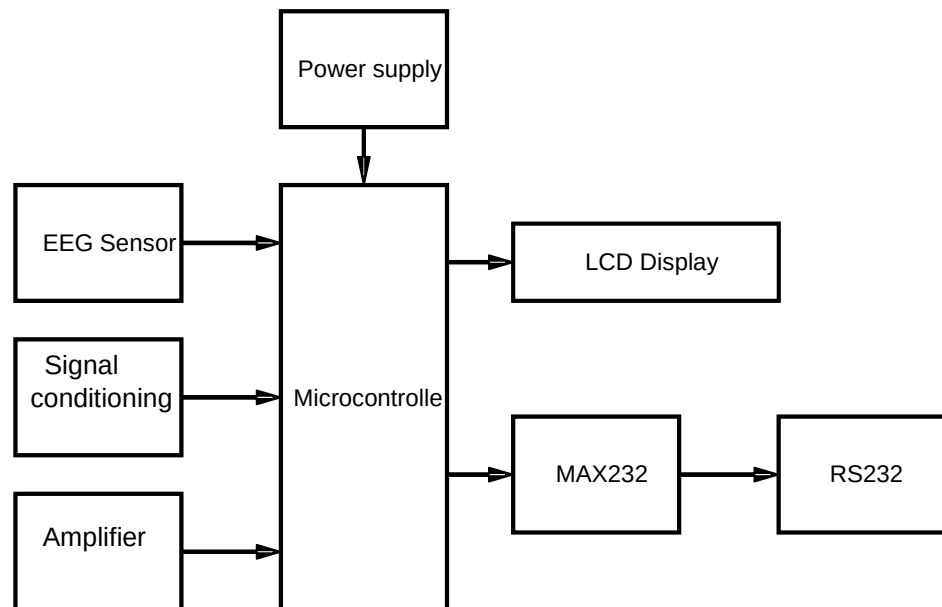


Fig 5.7 Mail and storage

Notice in the above diagram that a relay uses an electromagnet. This is a device consisting of a coil of wire wrapped around an iron core. When *electricity* is applied to the coil of wire it becomes *magnetic*, hence the term *electromagnet*. The A B and C terminals are an SPDT switch controlled by the electromagnet. When electricity is applied to V1 and V2, the electromagnet acts upon the SPDT switch so that the B and C terminals are connected. When the electricity is disconnected, then the A and C terminals are connected. It is important to note that the

electromagnet is magnetically linked to the *switch* but the two are NOT linked electrically.

System Architecture

Power supply is used to supply the overall power to the sensors and TFT controller. An electroencephalogram (EEG) is a test used to evaluate the electrical activity in the brain. Brain cells communicate with each other through electrical impulses. An EEG can be used to help detect potential problems associated with this activity. Signal conditioning is a process of data acquisition, and an instrument called a signal conditioner is used to perform this process.

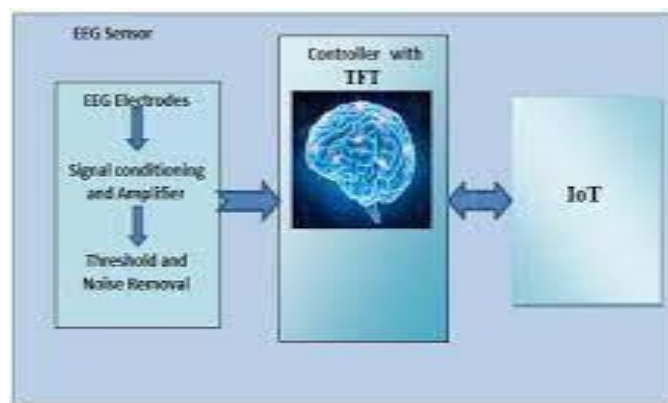


Fig 5.8 System Architecture

In this Module, the communication takes place into Data Analytical and Cloud Cloud perform all tasks related to storage, data manipulation, transition, and collaboration internally, a healthcare unit would need to invest heavily in infrastructure and the resources for maintenance. Analytics makes it easier to analyze large volumes of healthcare data generated on an everyday basis and ascertain critical insights about the disorders, which would otherwise prove impossible to predict. In fact, even doctors will soon be using analytics to know the complete history of their patient

The dataset used in this paper consists of microscopy images of axonal fibers from part of a mouse cerebral cortex. A Thy1-eGFP-M mouse was sacrificed according to the protocol approved by the MIT Institutional Animal Care and Use Committee and the Division of Comparative Medicine. A 170- μ mthick brain slice was obtained and processed by MAP protocol. The slice was stained by mouse SMI-312 antibody

and antimouse IgG-Alexa Fluor 594 antibody to label all axonal fibers. After fourfold linear expansion in deionized water, the sample was imaged by the Leica TCS SP8 microscope system with a 25x, 0.95 NA water-immersion objective and a white light laser source. The acquired image volume consists of 800 slices of 2D tiff images that are vertically registered along the z axis, with a pixel resolution of 0.325 μm in x, y and 1 μm in z. We divided the original 250 GB data into 50 x 50 subvolumes using the binary tree encoding described above. The subvolumes were then processed in parallel using pMatlab. For segmenting the fibers, we used a CNN with three convolutional and two fully connected layers. Convolutions are 3 x 3 with stride and padding such that the total receptive field is 19 x 19 x 19. The computed vertices and edges were subsequently inserted into Accumulo, while the fiber track 3D coordinates were saved to flat files.

Query Performance : For illustration purposes, a subset of 16 x 16 subvolumes were used for testing. It included a total of 618,976 nodes. A randomly chosen single node from one random subvolume was extracted using both flat files and Accumulo. Performance time was measured as the average of 5 retrievals. Notice the y axis scale is logarithmic to better show the trend that Accumulo outperforms flat files at every subvolume density (Fig. 4). In this test data, Accumulo offers an average of 1- 2 orders of magnitude of speedup. Performance of using flat files is limited by the fact that in order to extract one node from one subvolume, the entire subvolume must be loaded. Retrieval rate is relatively constant in Accumulo regardless of subvolume density, while flat file performance quickly degrades as density increases. The fast query performance of Accumulo is desirable because it allows for quick extraction of useful information for graph analytics.

Graph Analytics : As described previously, one important application of our system is to perform graph analytics on neuron axon fibers in order to gain insights on brain structural patterns. There are numerous graph analytics of interest. With the combination of Accumulo and D4M, our system supports fast and easy iterative analysis with just a few lines of code. For example, in a given region of interest, we often want to know which nodes have more connections. We can achieve this by using the BFS algorithm described earlier. Using the same 16 x 16 subvolume test data, extracting all nodes with three or more connections results in 383 nodes, and

extracting nodes with four or more connections results in 32 nodes (Fig. 5). Based on our visual inspection, all the connections were correctly identified by the algorithm. To find all the nodes connected in some way to the start node takes 3.7 ± 0.5 seconds regardless of the subvolume density and size. The performance of the BFS algorithm is fast and consistent, thus scalable to any data size. D. Interactive Web-based 3D Visualization Leveraging the backend that combines geohashing-based binary tree encoding, Accumulo, and flat files, we implemented an interactive web-based 3D graphical user interface (GUI) in Python. With binary tree encoding, it is possible to visualize large 3D data at multiple zoom levels with low latency. Fig. 6 is an example of dividing a data volume into 16×16 sub volumes, and representing the division of data by an eight digit binary tree encoding, which produces four zoom levels. The inset shows an example of fusing 4 adjacent sub volumes extracted from the original volume for a zoomed in visualization. In the white matter of the brain regions, the density of the connectivity can quickly become very high. To maintain interactive nature of the GUI, we designed our visualization algorithm in a way that the total number of fibers shown is constant, regardless of density. This ensures that our GUI will run smoothly independent of differences in graphic engine capabilities.

Testing

The concept and functionality of the testing platform was tested on a set of test cases of an on-going project.

The first step was to build the processor board for the target processor. The board was built on a breadboard, as displayed in the analogue outputs; digital I/O and communication port were wired to separate pin header connectors. The processor board contained also a test pins for PWM output, PWM input and power connections. Addition, the board included RJ11 connector for programmer/debugger. This connector was solely used for programming the blank target MCU with the binary image.

One of the tests performed concerned a hysteresis of operation modes depending on the current flow through the UUT. The current is an analogue signal from 0V to +5V. The Unit is defaulted to state A at the power on. The unit should enter state B when 48 the current drops to -0.5A or below and return back to the state A when the

current is 0.5A or greater. In addition to monitoring current, the unit measures two voltages. To avoid simulating abnormal behavior, the voltages are also scripted so that the voltage potential corresponds with the hysteresis trigger current. The example script can be found in appendix 8. The script is divided into three script groups: Configuration scripts, dynamic scripts and static or irrelevant scripts. This makes it easier to read the script and separate signals that are essential for the current test from the signals that are irrelevant for the result. The first group is configuration group where the basic configuration is made. The reference voltage for DACs is set to +5V by Vref command. Next, the input and output ports are defined. The Processor board is connected to ports A, C and D. Without data direction definition the port defaults to all outputs, but every port is defined for clarity. The unit also monitors fan speed via pulse capture. This is emulated in the platform with PWM output, which frequency is set to 10Hz. The next group is dynamic script group. This group contains all signals that are mentioned in the test case and/or that change during the test run.

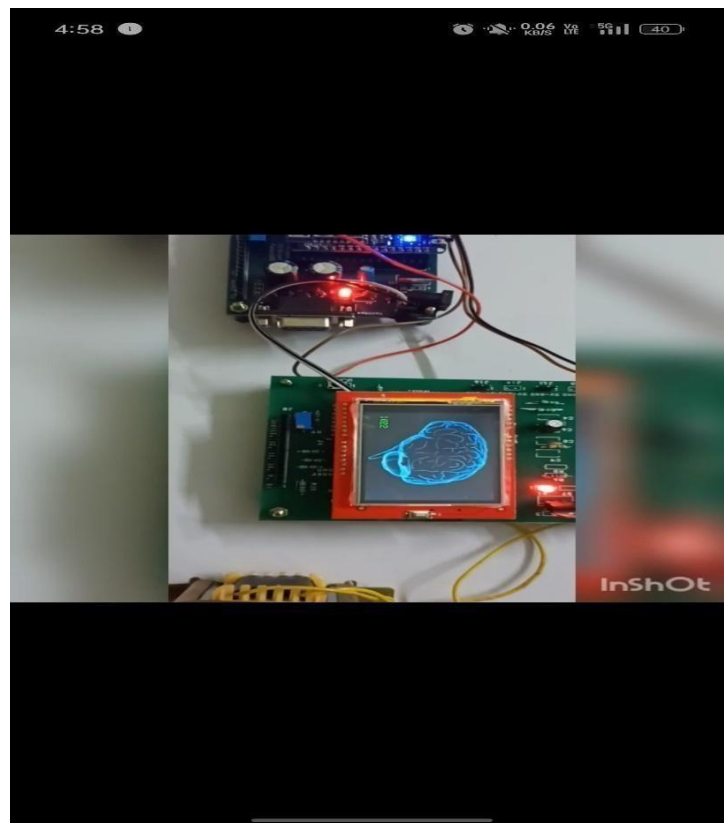


Fig 5.9 Screenshot of working model

The test contains three main signal inputs: current and two voltages. The actual voltage and current values that the simulated signals represent in the HW environment are commented in the script to make the script more readable. In addition to the analogue input signals the processor is reset at the beginning of the test to ensure the software is in the right mode for the test. The last group is for static or irrelevant scripts. These are signals that are not mentioned in the test script. These signals can be e.g. external control signals that are used to switch the software in special mode or static analog signals, like alcohol, Eye blink and Vibration sensor readings.

CHAPTER 6

RESULT AND DISCUSSION

Result

The Virtual Brain project has resulted in significant advancements in our understanding of the human brain and its function. The project has developed a comprehensive computational model of the brain that can simulate brain activity at multiple scales, from the cellular level to large-scale networks. The model integrates data from different sources, including neuroimaging, electrophysiology, and genetics, to create a detailed picture of the brain's structure and function.

One major result of the project is the development of new tools for analyzing and visualizing brain data. These tools allow researchers to explore complex brain networks and to better understand how different brain regions are connected and interact. The Virtual Brain software can simulate the effects of brain damage or disease, enabling researchers to study the underlying mechanisms of neurological disorders and to test potential treatments.

The Virtual Brain has also led to important discoveries about brain function. For example, researchers have used the model to study the role of different brain regions in cognitive processes such as memory and attention. The model has revealed that these processes involve large-scale interactions between different brain regions, rather than being localized in specific regions as was previously thought.

Another important result of the project is the development of personalized brain models. The Virtual Brain software can be customized to an individual's specific brain structure and function, allowing researchers and clinicians to study and treat neurological disorders on an individual basis. This personalized approach has the potential to revolutionize the field of neurology, as it could lead to more targeted and effective treatments for a range of neurological conditions.

Furthermore, the Virtual Brain project has created an international network of researchers and clinicians who collaborate on developing and applying the model. This network has resulted in a wealth of new research, including studies of brain

development, aging, and disease. The project has also led to the development of new collaborations between researchers and industry, with potential applications in fields such as robotics and artificial intelligence.

In summary, the Virtual Brain project has had a significant impact on our understanding of the human brain and its function. The development of a comprehensive computational model of the brain has led to new tools for analyzing and visualizing brain data, important discoveries about brain function, and personalized approaches to studying and treating neurological disorders. The project has also fostered international collaboration and interdisciplinary research, with potential applications in fields beyond neuroscience. Overall, the Virtual Brain project has paved the way for exciting new developments in the study of the brain and its role in health and disease.

Feasibility study

Depending on the results of the initial investigation the survey is now expended to a more detailed feasibility study. "FEASIBILITY STUDY" is a test of system proposal according to its workability, impact of the organization, ability to meet needs and effective use of the resources.

Steps in feasibility analysis:

Eight steps involved in the feasibility analysis are:

1. Form a project team and appoint a project leader.
2. Enumerate potential proposed system.
3. Define and identify characteristics of proposed system.
4. Determine and evaluate performance and cost effective of each Proposed system.
5. Weight system performance and cost data.
6. Select the best proposed system.
7. Prepare and report final project directive to management.

Three key considerations involved in the feasibility analysis are

1. Economical feasibility
2. Technical feasibility
3. Behavioral feasibility

Economical feasibility : This study is carried out to check the economic impact that the system will have on the organization. The amount of fund that the company can pour into the research and development of the system is limited. The expenditures must be justified. Thus the developed system as well within the budget and this was achieved because most of the technologies used are freely available. Only the customized products had to be purchased.

Technical feasibility : This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources. This will lead to high demands on the available technical resources. This will lead to high demands being placed on the client. The developed system must have a modest requirement, as only minimal or null changes are required for implementing this system.

Behavioral feasibility: The staff of the organization was open-minded towards the acceptance of this new concept. No specialized training would-be needed, only a few hours. Of instructed demo to needs to be given to the user .hence the software exhibits behavioral feasibility

CHAPTER 7

SUMMARY AND CONCLUSION

Conclusion

We described a cloud-based system for performing large-scale brain connectivity analysis using Apache Accumulate and D4M. We demonstrated that our approach can achieve fast data query and extraction for graph analytics and visualization. Indexing of sub volumes is simple and logical with geo hashing-based binary tree encoding, which supports a Google Map-like viewer with multiple zoom levels. There are many avenues for future work. First, we would like to enhance the web GUI by making it more interactive and user friendly. Further, we intend to scale up to process much larger datasets (terabytes and above) with the goal of one day being able to perform such analysis on the human brain. We are currently exploring the use of Graphology to perform graph analytics directly inside accumulate. We are also exploring the use of a poly store database such as Big DAWG as a data viewer showing the original data and a zoomed-in view of 4 neighboring sub volumes extracted using binary tree encoding management solution that closely matches data sources with data management technology.

Future Enhancement

There are several potential areas for future enhancement of the Virtual Brain project. Here are a few possibilities:

1. *Integration with other brain imaging technologies:* The Virtual Brain could be enhanced by integrating it with other brain imaging technologies, such as electroencephalography (EEG), magnetoencephalography (MEG), and functional near-infrared spectroscopy (fNIRS). This would allow for a more comprehensive understanding of brain function and connectivity.
2. *Expansion of the brain regions and connections modeled:* Currently, the Virtual Brain focuses on a relatively small number of brain regions and connections. Future enhancements could expand the model to include more regions and connections, allowing for a more detailed and comprehensive understanding of brain function.

3. *Incorporation of individual variability:* The current version of the Virtual Brain uses a standard model of the human brain, which may not accurately represent the brain of every individual. Future enhancements could incorporate individual differences in brain structure and function, potentially leading to more personalized and precise models.
4. *Development of clinical applications:* The Virtual Brain could be used to develop new diagnostic and therapeutic approaches for neurological and psychiatric disorders. For example, the Virtual Brain could be used to simulate the effects of different drugs or stimulation techniques on brain function, allowing for more targeted and effective treatment.
5. *Integration with artificial intelligence (AI) and machine learning (ML) techniques:* The Virtual Brain could be enhanced by integrating it with AI and ML techniques, which could allow for more efficient and accurate analysis of brain data. For example, AI and ML could be used to identify patterns of brain activity that are associated with specific cognitive processes or behaviors.

Overall, the Virtual Brain project has the potential to revolutionize our understanding of brain function and to lead to new diagnostic and therapeutic approaches for neurological and psychiatric disorders. Continued development and enhancement of the project will be critical for realizing this potential.

REFERENCES

- [1] Blankertz.B, Curio.G and Muller.K.R, "Classifying single trial EEG: Towards brain-computer interfacing", Proceedings of Advances in Neural Information Processing Systems (NIPS), vol. 14, 2002.
- [2] Brattain.L et al., "A cloud-based brain connectivity analysis tool", 2017 IEEE High Performance Extreme Computing Conference (HPEC), pp. 1-6, 2017.
- [3] Bullmore .E. I and Bassett D.S, "Brain graphs: graphical models of the human brain connectome," vol. 7, pp. 113–140.
- [4] Chung .K and Deisseroth.K, "Clarity for mapping the nervous system," Nat Meth, vol. 10, no. 6, pp. 508–513, 06 2013.
- [5] Insel .T.R, Landis .S.C, and Collins .F.S, "The nih brain initiative," Science, vol. 340, no. 6133, pp. 687–688, 2013.
- [6] Ku.T, Swaney.J, Park-J.Y, Albanese.A, Murray.E, Cho J.H, Park Y.G, V. Mangena, Chen.J, and Chung.K, "Multiplexed and scalable super-resolution imaging of three-dimensional protein localization in size-adjustable tissues," vol. 34, no. 9, pp. 973–981.
- [7] NITRC: GraphVar: A user-friendly toolbox for comprehensive graph analyses of functional brain connectivity: Tool/resource info.
- [8] Smith.S.M , Miller.K.L, Salimi-Khorshidi.G, Webster.M, Beckmann.C.F, Nichols.T.E, Ramsey.J.D, and Woolrich.M.W, "Network modelling methods for FMRI," vol. 54, no. 2, pp. 875–891.
- [9] Sporns.O, "The human connectome: a complex network: The human connectome," vol. 1224, no. 1, pp. 109–125.
- [10] Zhang.X et al., "Internet of Things Meets Brain–Computer Interface: A Unified Deep Learning Framework for Enabling Human-Thing Cognitive Interactivity", IEEE Internet of Things *Journal*, vol. 6, pp. 2084-2092, 2019.

APPENDIX

SOURCE CODE

```
#include <htc.h>
#include "lcd16.h"
void txs(unsigned char val)
{
    int i;
    while(!TXIF)
        continue;
    TXREG=val;
    for(i=0;i<1200;i++);
}
void delay2()
{
    long i;
    for(i=0;i<10000;i++);
}
unsigned char rxs(void)
{
    int c=0;
    while(!RCIF)
    {
        c++;
        if(c>5000)
            break;
    }
    return RCREG;
}
unsigned char val[6],flagx=0,val1[9],val2,val3,val4,val5;
int j=0;
int x;
int sp,sp1,spt;
```

```

        unsigned int d=0,e=0,f=0,flgg=0;
        unsigned int m=0;
void Delay()
{
    long i;
    for(i=0;i<5000;i++)
        {
        }
}
void Delay2()
{
    long i;
    for(i=0;i<100000;i++)
        {
        }
}
void delay3(unsigned int val)
{
    int i;
    for(i=0;i<val;i++)
        {
        }
}
void ssu(unsigned char val[],unsigned char length)
{
    int i;
    for(i=0;i<length;i++)
        {
            txs(val[i]);
        }
}
void sms1()
{

```

```

    txs('A');txs('T');txs('+');txs('C');txs('M');txs('G');
    txs('S');txs('=');
    txs("");
    ssu("9384560616",10);
    txs("");
    txs(13); txs(10);
    delay2();
    ssu("Blood Glucose:",13);
    txs(13); txs(10);
    txs((val1[1]/100)+0x30);
    txs(((val1[1]%100)/10)+0x30);
    txs((val1[1]%10)+0x30);
    txs('m');
    txs('g');
    txs('/');
    txs('d');
    txs('l');
    delay(10000);
    txs(26);
    delay3(200000);
}

void sendtx(unsigned char val[],unsigned char length)
{
    int i;
    for(i=0;i<length;i++)
    {
        txs(val[i]);
    }
}

```

/* Sample code to set up the A2D module */

```

void init_a2d(void){
    ADCON0=0; // select Fosc/2

```

```

        ADCON1=2; // select left justify result. A/D port configuration 0
        ADON=1;      // turn on the A2D conversion module
    }

/* Return an 8 bit result */
unsigned char read_a2d(unsigned char channel){
    channel&=0x07;    // truncate channel to 3 bits
    ADCON0&=0xC5;    // clear current channel select
    ADCON0|=(channel<<3); // apply the new channel select
    Delay();
    GO=1; // initiate conversion on the selected channel
    while(GO)continue;
    return(ADRESH);    // return 8 MSB of the result
}

void DelayON(unsigned int val)
{
    unsigned int i;
    for(i=0;i<(val*5000);i++);
}

unsigned char rval=2,rvalt=2,hb,hbt,f1,fl,oc;
int c;

void main()                //Main entry
{
    CMCON=0x07;
    RCSTA=0x90;                //
    receive enabling with the same speed
    TXSTA =(0x24);                // High speed selection baud
    rate 9600
    SPBRG = 64;
    // ADCON1=0x02;
    //    ADCON0=0x81;
    CVRCON=0x00;
    TRISD=0x00;

```

```

    TRISE=0x00;
    Lcdinit();

    init_a2d();

    TRISB=0x00;
    TRISC=0xFF;

nRBPU=0;

RB1=0;
RC0=1;

    //GIE=1;

    //PEIE=1;

    //INTE=1;
    txs('A');txs('T');txs(13);    txs(10);delay2();
    txs('A');txs('T');txs('+');txs('C');txs('M');txs('G');
    txs('F');txs('=');txs('1');txs(13);    txs(10);delay2();
    delay2();
    while(1)
    {
while(RCIF==0)
    {
        Delay();
        val3=read_a2d(0);

        Lcdcmd(0xC0);
        lcddata(' ');
        lcddata((val3/100)+0x30);
        lcddata(((val3%100)/10)+0x30);

```

```

lcddata((val3%10)+0x30);
lcddata(' ');
lcddata((hb/100)+0x30);
lcddata(((hb%100)/10)+0x30);
lcddata((hb%10)+0x30);
lcddata(' ');
lcddata((hbt/100)+0x30);
lcddata(((hbt%100)/10)+0x30);
lcddata((hbt%10)+0x30);

```

```

if(val3>100)
{
    if(f1==0)
        if(hbt<100)
        {
            hbt++;
            f1=1;
        }
}

```

```

if(val3<150)
{
    f1=0;
}

```

```

hb++;

```

```

if(hb==100)
{
    val1[1]=hbt*4;
    hbt=0;
    hb=0;
}

```

```

/*    lcdcmd(0xC0);

```

```

        lcddata((val1[1]/100)+0x30);
        lcddata(((val1[1]%100)/10)+0x30);
        lcddata((val1[1]%10)+0x30);
        lcddata(' ');
        lcddata((hbt/100)+0x30);
        lcddata(((hbt%100)/10)+0x30);
        lcddata((hbt%10)+0x30);    */

    if((val1[1]>40)&&(val1[1]<150))
    {

if(val1[1]>100)
{
RB0=0;RB1=1;DelayON(3);RB1=0;oc=1;
}

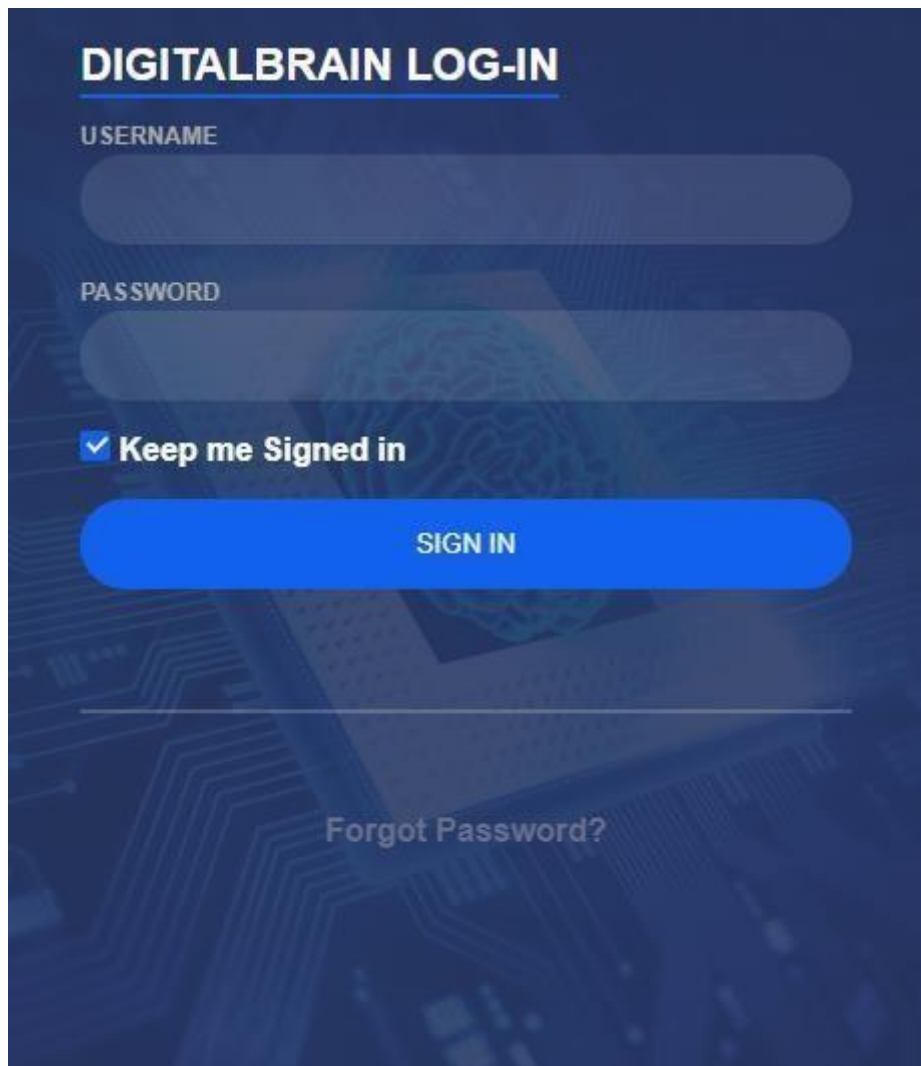
        val1[1]=val1[1];

Lcdcmd(0x80);
        lcddata('B');
        lcddata('G');
        lcddata('=');
        lcddata((val1[1]/100)+0x30);
        lcddata(((val1[1]%100)/10)+0x30);
        lcddata((val1[1]%10)+0x30);
        lcddata('m');
        lcddata('g');
        lcddata('/');
        lcddata('d');
        lcddata('l');
        sms1();

```

```
    val1[1]=0;
}
    }
    val2=RCREG;
    RCIF=0;
    OERR=0;
    CREN=0;
    CREN=1;
}
}
```

SCREENSHOTS



The screenshot shows a login interface for 'DIGITALBRAIN LOG-IN'. It features a dark blue background with a faint circuit pattern. The form includes fields for 'USERNAME' and 'PASSWORD', a checkbox for 'Keep me Signed in' which is checked, and a prominent blue 'SIGN IN' button. A 'Forgot Password?' link is located below the button. The title 'DIGITALBRAIN LOG-IN' is underlined.

Fig : Log in to web



Fig : Application



Brain Status Logs

Click Here To Delete Logs

CLEARLOG

LogID	DATA	Logdate	LogTime
1	Normal	03/21/2019	17:22:12
2	Brain_Dead	03/21/2019	17:24:11

Fig : Data collected

SETTINGS

Enter The Mobile numbers,Email ids and the Messages to be sent in case of Emergency

MESSAGEDELIVERY STATUS:OPEN

Mobile Numbers

MobileNumber:1 Message:

MobileNumber:2 Message:

Fig : Personal information

RESEARCH PAPER

VIRTUAL BRAIN USING IOT

KAVIYA.K

*Department of Electronics and
Communication Engineering
Sathyabama Institute of Science
and Technology, Chennai –
600119,India
[kaviyak775@gmail.co
m](mailto:kaviyak775@gmail.com)*

JAYA KEERTHANA.K

*Department of Electronics and
Communication Engineering
Sathyabama Institute of Science
and Technology, Chennai –
600119,India
[Prithikshakeets2374@gmail.c
om](mailto:Prithikshakeets2374@gmail.com)*

Dr. T.VINO

*Department of Electronics and
Communication Engineering
Sathyabama Institute of Science
and Technology, Chennai –
600119,India
[vinodhevan@gmail.co
m](mailto:vinodhevan@gmail.com)*

Abstract— Virtual brain research is accelerating the growth of the low-cost real-time brain IOT Systems. Hardware improvements that increase the capability of the virtual brain analyzing and brain IOT wearable sensors have made it possible several new software frameworks for developers to use and create applications combining the brain and IOT. It also enables several sensory pathways for communications with large sized data to users' brains. The intersections of these two research paths are accelerating both fields and will drive the need for an energy-aware infrastructure to help the wider local bandwidth demands in the IOT cloud. In this paper, we complete a survey on brain activity extraction in IOT from different perspectives, and electroencephalogram (EEG) based IOT models, machine learning, and current active platforms. Based on our investigations, the main findings of the survey highlight three important development trends of virtual brain development systems, which are EEG, IOT, and cloud computing.

network graph, but also the 3D location information

I. Introduction

In the list of the top US Government's BRAIN Initiative, the ability to map the human brain at different scales with enhanced throughput and resolution. A complete picture of the brain structure will provide new insights into brain functions and may facilitate new treatments and drug inventions for brain disorders. Modern advances in brain imaging, such as the CLARITY and MAP tissue clearing techniques, make it possible to collect huge volumetric images of brain tissue at cellular and sub-cellular resolutions.

The high throughput and high resolution brain imagery, however, create a challenge for well organized processing and analysis. We have developed an automated dense axonal fiber tracing pipeline that can track long-range fibers and construct 3D connectivity maps, which include not only the vertices and edges of a

associated with each fiber's track. In addition to typical graph analysis, we are interested in identifying long-range neuron fibers connections and fibers crossings, both of which could reveal informative patterns when analysed at cellular resolution and at long range (≥ 1 mm). Brain graphs provide a framework to portray the structural or functional topology at multiple levels.

The Apache Accumulate database is based on Google's Big table database and allows for fast ingestion and extraction along with the capacity to hold largescale data

The Dynamic Distributed Dimensional Data Model (D4M) provides a mathematical framework, based on the mathematics of associative arrays that has been applied to diverse domains, such as cyber security, bioinformatics, free text, and social media data. D4M can be used to do linear algebraic operations inside a database. D4M works seamlessly with the pMatlab parallel computing environment that allows users of MATLAB/Octave programming languages to use a single-program-multiple data (SPMD) parallel programming model. PMatlab sits on top of a message passing interface library such as Mat labMPI or bcMPI and has been used for a variety of parallel computing applications.

II. Literatur Survey

Various surveys and conclusions were considered first in which the IOT gateway system based on zigbee and GPRS protocols were considered. According to the typical IOT application scenarios and requirements from telecom operators, has presented the data transmission between wireless sensor networks and mobile communication networks, and conversion of different sensor network protocols. To enhance and improve wireless connectivity for heterogeneous IOT devices to communicate, they need multiple antennas, one for each band or a wideband antenna. The iot revolution is rearranging modern health care with the best technological, economic, and social prospects. These advances in IOT-based health care technologies and

architectures/platforms, applications, and industrial trends in IOT-based health care solutions. In addition, the distinct IOT security and privacy features, include the security requirements, threat models, and attack taxonomies from the health care perspective.

III The Comparison Between Natural And Artificial Brain

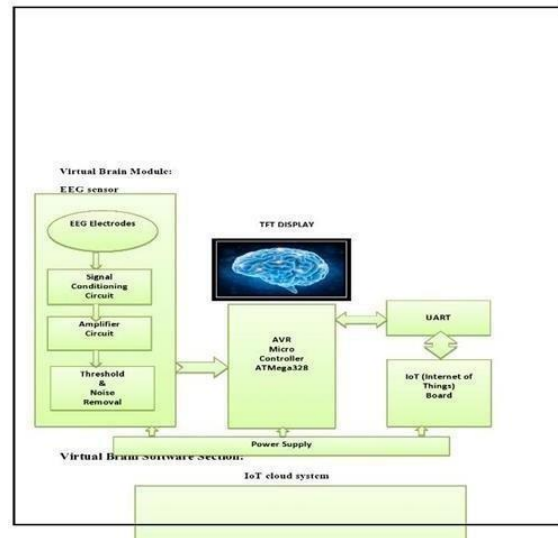
1. Input in the nervous system of our body, the neurons will be responsible for the message passing. In a similar way, the artificial nervous system can be created by artificial neurons by replacing them with a silicon chip.

2. Interpretation: The electric impulses accepted by the brain from the neurons are interpreted in the brain. The interpretation in the brain is done by means of certain states of more neurons. The interpretation of the electric impulses accepted by the artificial neuron can be done by means of a set of registers. The different values in this register will represent different states of the brain.

3. Output: Depending on the states of the neurons, the brain sends the electric impulses representing the responses which are further received by the sensory cells of the body to respond. Similarly, based on the state of the register, the output signal can be given to the artificial neurons in the body which will be collected by the sensory cells.

4. Memory: There are certain neurons in our brain which represent certain states permanently. When required, this state is interpreted by our brain and we can remember past things. To remember things, we force the neurons to represent certain states of the brain permanently or for any interesting or serious matter that has happened implicitly. In a similar way, the required status of the registers can be stored permanently, and when required this information can be retrieved.

5. Processing: When we make a decision, think about something or make any computation, logical and arithmetic calculations are done in our neural circuitry. In the same way, the decision making can be done by the computer by using some stored states and the collected input and by performing some arithmetic and logical calculations.



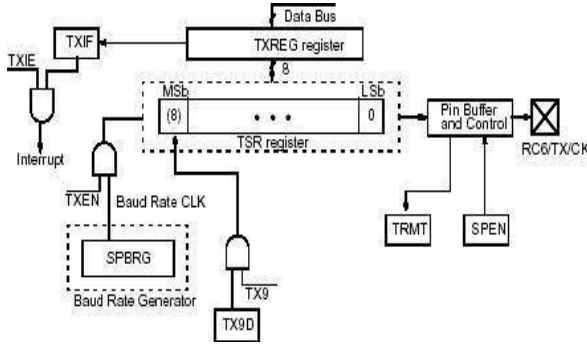
IV. Methodology

The brain collects signals from sensory neurons (nerve cell bodies and their axons and dendrites) in the central and peripheral nervous systems, and in response it generates and produces new signals that instruct the near-by parts of the body to move or respond in some way. It also combines signals collected from the body with signals from adjacent areas of the brain, giving rise to perception and consciousness. The human efficiency to feel, interpret and even see is controlled, in computer-like calculations, by the unique nervous system. It functions through electric impulses through your body. As shown in the above diagram, through an EEG sensor, the data from the brain is collected and given to the amplifier circuit for amplification, which will aid in the transmission of the input signals from the brain. The information is transmitted and stored in a cloud storage system. By this, the memory could be accessed whenever necessary by the user who has the password to access it.

V. Working

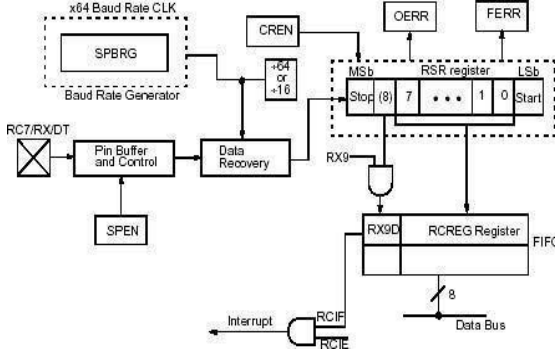
Transmitter:

Initialize the SPBRG register for the exact baud rate. If a high speed baud rate is desired, set bit BRGH. Enable the asynchronous serial port by clearing bit SYNC and setting bit SPEN. If interrupts are desired, then set enable bit TXIE. If 9-bit transmission is desired,



Then fix the transmit bit as TX9. Allow the transmission by setting the bit as TXEN, which will also set the bit as TXIF. If the 9-bit transmission is selected; the ninth bit should be loaded as bit TX9D. Load data to the TXREG register and transmission starts. **Receiver:**

Initialize the SPBRG registered for the appropriate baud rate. If a high speed baud rate is desired, set bit



BRGH. Enable the asynchronous serial port by clearing the bit SYNC and setting the bit as SPEN. If interrupts are desired, then set the bit as RCIE. If 9-bit reception is desired, then set the bit as RX9. Enable the reception by setting the bit as CREN. Flag bit RCIF will be set when reception is complete and an interrupt will be generated if enable bit RCIE is set. Read the RCSR register to get the ninth bit and determine if any error occurred during the reception. Read the 8-bit received data by analysing the RCREG register. If any error occurs, remove the error by clearing the enable bit CREN

VI. Description Of Dataset

The dataset used in this paper consists of microscopy images of axonal fibers from part of a mouse's cerebral cortex. A Thy1eGFP-M mouse was sacrificed according to the protocol approved by the MIT Institutional Animal Care and Use Committee and the Division of Comparative Medicine. A 170- μ m thick brain slice was obtained and processed by MAP protocol. The slice was stained by mouse SMI-312 antibody and IgG -Alexa Fluor 594 antibody to label all axonal fibers. After fourfold linear expansion in deionized water, the sample was imaged by the Leica TCS SP8 microscope system with a 25x, 0.95 NA water-immersion objective.

VII. Query Performance

For illustration purposes, a subset of 16 x 16 subvolumes was used for testing. It included a total of 618,976 nodes. A randomly chosen single node from one random subvolume was extracted using both flat files and Accumulo. Performance time was measured as the average of 5 retrievals. Notice the y axis scale is logarithmic to better show the trend that Accumulo outperforms flat files at every subvolume density.

VIII. Graph Analytics

As described previously, one important application of our system is to perform graph analytics on neuron axon fibers in order to gain insights on brain structural patterns. There are numerous graphic analytics of interest. With the combination of Accumulo and D4M, our system supports fast and easy iterative analysis with just a few lines of code. For example, in a given region of interest, we often want to know which nodes have more connections. We can achieve this by using the BFS algorithm described earlier. Using the same 16 x 16 subvolume test data, extracting all nodes with three or more connections results in 383 nodes, and extracting nodes with four or more connections results in 32 nodes. Based on our visual inspection, all the connections were correctly identified by the algorithm. To find all the nodes connected in some way to the start node takes 3.7 ± 0.5 seconds regardless of the subvolume density and size. The performance of the BFS algorithm is fast and consistent, thus scalable to any data size. D. Interactive Web-based 3D Visualization Leveraging the backend that combines geohashing-based binary tree encoding, Accumulo, and flat files, we implemented an interactive webbased 3D graphical user interface (GUI) in Python. With binary tree encoding, it is possible to visualize large 3D data at multiple zoom levels with low latency. Fig. 6 is an example of dividing a data volume into 16 x 16 subvolumes, and representing the division of data by an eight digit binary tree encoding, which produces four zoom levels. The inset shows an example of fusing 4 adjacent subvolumes extracted from the original volume for a zoomed-in visualization. In the white matter of the brain regions, the density of connectivity can quickly become very high.

IX. TESTING

The concept and functionality of the testing platform was tested on a set of test cases of an on-going project.

Processor Board(Microcontroller):

The first step is to build the processor board for the target processor. The board was built on a breadboard, as displayed in the analogue outputs; digital I/O and communication port were wired to separate pin header connectors. The processor board also contained pins for

PWM output, PWM input and power connections. In addition, the board included an RJ11 connector for the programmer/debugger. This connector was solely used for programming the blank target MCU with the binary image.

Test Case(Power Supply & Sensors):

One of the tests performed is concerned as hysteresis of operation modes depending on the current flow through the UUT. The current is an analogue signal from 0V to +5V. The unit is defaulted to the state A at the power on. The unit should enter state B when the current drops to -0.5A or below and return back to the state A when the current is 0.5A or greater. In addition to monitoring current, the unit is measured in two voltages. To avoid simulating abnormal behavior, the voltages are also scripted so that the voltage potential corresponds with the hysteresis trigger current. An example of the script can be found in appendix 8. The script is divided into three script groups: configuration scripts, dynamic scripts and static or irrelevant scripts. This makes it easier to read the script and separate signals that are essential for the current test from the signals that are irrelevant to the result. The first group is the configuration group where the basic configuration is done. The reference voltage for DACs is set to +5V by Vref command. Next, the input and output ports are defined. The processor board is connected to 3 ports, which are ports A, C and D. Without data direction definition, the port defaults to all outputs, but every port is defined for clarity. The unit also observes fan speed via pulse capture. This is emulated in the platform with PWM output, whose frequency is set to 10Hz. The next group is the dynamic script group. This group includes all the signals that were mentioned in the test case and/or that changed during the test run.

X. Feasibility Study

Depending on the results of the initial investigation, the survey is now expended on a more detailed feasibility study. The “FEASIBILITY STUDY” is a test of system proposal according to its workability, impact of the organization, ability to meet needs and effectual use of the resources.

Economic Feasibility:

This study is carried out to check the economic impact so that the system will have on the organization. The amount of money that the company can pour into the research and development of the system is limited. The expenditures must be justified. Thus, the systems developed as well within the budget and this is achieved because most of the technologies used are freely available. Only the customized products have to be purchased.

Technical Feasibility:

This study is carried out to check the technical feasibility, which is, the technical requirements of the system. Any system introduced must not have a high demand for the available technical resources. This will lead to very high demands on the available technical resources. This will lead to high demands which are being placed on the client. The Internet of Things (IoT) is an environment in which objects, animals or people are provided with unique identifiers.

XII Summary

The main purpose is to upload the human brain onto a machine. So that humans can think and take decisions without any effort. After the death of the body, the virtual brain will act as that person. So, even after the death of a person, we will not lose the knowledge, intelligence, personalities, feelings and memories of that man and be used for the development of human society. The cloud, having the memory, can be accessed from any part of the world when the password is set and shared confidentially. The data is highly confidential, so the cyber safety of the data is very important. The concept is that even if the person is absent physically, his presence could be brought back to make decisions and do calculations. Any brain related ailments like memory loss due to Alzheimer's and Parkinson's can be cured.

XIII Future Perspective

On collecting the data from the sensors and updating it to the main server in the hospital. So, this process occurs irrespective of the presence of the doctor or the patient's relatives beside the patient. The doctor can advise the patient to take any treatment by analyzing the patient's health condition. The medical evaluation process can be made easier by this project using IOT technology and the patient can lead a carefree life.

XIV Conclusion

We described a cloud-based system for performing large-scale brain connectivity analysis using Apache Accumulate and D4M. We demonstrated that our approach can achieve speed of data query and extraction for graph analytics and visualization. Indexing of sub volumes is simple and logical with geo hashingbased binary tree encoding, which helps a Google Map-like viewer with multiple zoom levels. There are many avenues for future work. First, we would like to enhance the web GUI by making it more useful and user-friendly. Further, we intend to scale up processing much larger datasets with the goal of one day being able to perform such analysis on the human brain. We are currently exploring the use of Graphology to perform graph analytics directly inside accumulate. We are also exploring the use of a polystore database such as Big DAWG, as a data viewer showing the original data and

Reference

- [1] E. T. Bullmore and D. S. Bassett, "Brain graphs: graphical models of the human brain connectome," vol. 7, pp. 113–140.
- [2] K. Chung and K. Deisseroth, "Clarity for mapping the nervous system," *Nat Meth*, vol. 10, no. 6, pp. 508–513, 06 2013. [Online]. Available: <http://dx.doi.org/10.1038/nmeth.2481>
- [3] T. R. Insel, S. C. Landis, and F. S. Collins, "The nih brain initiative," *Science*, vol. 340, no. 6133, pp. 687–688, 2013.
- [4] T. Ku, J. Swaney, J.-Y. Park, A. Albanese, E. Murray, J. H. Cho, Y.-G. Park, V. Mangena, J. Chen, and K. Chung, "Multiplexed and scalable super-resolution imaging of three-dimensional protein localization in size-adjustable tissues," vol. 34, no. 9, pp. 973–981. [Online]. Available: <http://dx.doi.org/10.1038/nbt.3641>
- [5] NITRC: GraphVar: A user-friendly toolbox for comprehensive graph analyses of functional brain connectivity: Tool/resource info. [Online]. Available: <https://www.nitrc.org/projects/graphvar/>
- [6] S. M. Smith, K. L. Miller, G. Salimi-Khorshidi, M. Webster, C. F. Beckmann, T. E. Nichols, J. D. Ramsey, and M. W. Woolrich, "Network modelling methods for FMRI," vol. 54, no. 2, pp. 875–891.
- [7] O. Sporns, "The human connectome: a complex network: The human connectome," vol. 1224, no. 1, pp. 109–125. [Online]. Available: <http://doi.wiley.com/10.1111/j.1749-6632.2010.05888.x>