**Lightweight Easily-deployable Inexpensive Nodes for Temporary Wireless Mesh Networks**

**I. Introduction**

**1.1 Project Context**

Currently, mobile phones are regarded as the main tool for communication. These mobile phones rely on telecommunication networks to send and receive messages. This paper will focus on creating an ad-hoc architecture that lets mobile phones connect to each other using Bluetooth and Wi-Fi without relying on telco network.

Moore's Law states that, as technology advances, hardware cost and size decreases. This makes it possible to create and deploy a network composed of multiple small nodes that are interconnected to increase the range and effectiveness of an ad-hoc network. This network relays the information to each other while following protocols to effectively establish communication between devices. Having more nodes creates a network that accommodates a bigger land area. It can also be easily scalable because the inter connectivity does not rely on a central node so the individual nodes can be increased and decreased as the situation demands. The reduced prices on the hardware components also made network nodes easily replaceable if upon deployment a single device stops functioning.

**1.2 Purpose and Description**

Philippines is a heavy user of mobile phones. According to Business Monitor International (2012), the number of its subscribers will reach 117 million by the end of 2016. Mobile phones primarily use radio communications to send and receive data. With this information, it can be said that Filipinos rely heavily on radio communications as a mean to connect with each other. When the commonly used cellular networks are out or is not available, the mobile phones that majority of people rely on becomes useless as a tool for communication.    
   
Time and time again the services network providers give are cut by natural disasters physically destroying connections and towers or by the occasional power failures. The effect of these cuts are more sever in remote provinces where the equipment are scarce and obsolete. Due to the design of the current network infrastructure, it takes a significant time to repair and restore the connection. A good example of a disruptive event would be Typhoon Yolanda, internationally known as Haiyan. The storm considered to be the most powerful storm to make landfall in recorded history. It struck the Philippines on November 8, 2013 affecting 1,473,251 families with a casualty count of 6,300. The typhoon knocked out power lines and damaged the 63 cell sites of all the carriers in the area. (Camus, 2013) Right after the storm passed, the whole area was leveled and rescuers had to search for survivors with almost no radio communications as the network infrastructure also went down with all the other structures.    
   
With communications down, coordination of actions among response teams in multiple areas was almost impossible. It added an additional layer of difficulty for the teams in handling situations that demand cooperative action. Even radio, TV and news stations found it difficult to communicate with their own teams on the ground. With the severity of the situation, the UN took notice and decided to help because the local agencies in the area are also having difficulty restoring the communications network. (Ambil, 2013)    
   
Despite Filipinos being a heavy user of mobile devices, there are still places in the Philippines that are not reached by telecommunication signals. Examples are mining sites and rural towns who do not even have access to constant electrical supply. Mining sites are usually located far from any city, town, or any place that is being used by people because of its physical hazards and its nature of producing chemical wastes (Section 19 of Republic Act No. 7942). Since the target of cell sites are mostly highly populated areas, mining sites usually are out of their range. This makes it difficult for the workers to communicate with anyone outside the area like family or friends. Besides the issue on priority, the geographical structures of the Philippines also adds to the difficulty of building new cell towers to give telecommunication access to rural places.

This research on an easy-setup implementation of mesh networking will have many applications such as aiding emergency response teams' coordination by creating an open line of communication among them when cellular networks are down and when there are blackouts after events such as typhoon Yolanda. It can also be used by workers in remote mining operations where signals are obstructed and completely blocked big geographic formations or any user that wants to deploy their own personal network.    
With the use of a resource that majority of the Filipino citizens have, the smartphone, the project will be easily implemented and won't require and additional instruction or equipment.    
There are many possibilites on where to apply the end product of this module. In events of a network failure or a power outage after a natural calamity for example, real-time connections are preserved by creating a reliable network infrastructure that can be easily established and removed as needed. This will improve the efficiency on all levels because it is a fact that communication is key to good coordination between parties. This added efficiency in turn reduces the casualties of natural disasters by enabling reliable communication between all the parties involved geographically located within the blanket of the network.

**1.3 Objectives**

This study aims to find a cost efficient and easily deployable method to improve the communication among users of Wi-Fi capable mobile devices who are situated in a place where there is no currently available network infrastructure to connect to.

**Specific Objectives**

• To connect multiple microcontrollers/ microcomputers as intermediary nodes of a mesh network;    
• To enable data transfer between microcontrollers/ microcomputers;    
• To connect mobile devices as nodes of the network;    
• To relay a message to and from endpoints of the network(mobile devices);    
• To enable the network nodes to automatically detect and connect to nearby nodes;    
• To know and use the fastest path between nodes to maintain efficient reliable data transfer.(self healing)    
• To create a solution that will be cost effective.

**1.4 Scope and Limitations**

The scope of this research covers the development of network nodes that can be used to establish a private network using the mesh network topology for areas with no available network service from telecommunications companies.   
   
The research includes the nodes being raised in the air with the use of balloons to cover an area with network connectivity which is anchored to the ground using heavy boxes.   
   
The study covers the development of the network nodes along with its input and output interfaces. Integrating the networking protocols with the hardware of microcontrollers and additional modules to establish a reliable connection between 2 or more smartphones. The connection will be able to allow users to send messages to each other using an existing peer to peer chat application called FireChat.

After the development of the nodes, further studies will be done to fully know its specifications i.e. effective range, speed, and traffic capacity.

The paper will also discuss different scenarios that are likely to happen in the Philippines wherein these nodes will be useful. The ideal way of positioning the nodes based on these scenarios will be included.

**II. Related Literature**   
   
**Wi-Fi Peer-to-Peer**   
Mobile chat applications let users communicate with each other from 2 end devices. Chat applications usually need a web server in order to work. A web server may be offline, such as XAMPP, or online, such as an online server or a registered domain. This research will need a mobile chat application that is not provisioned with a web server. It should only rely on a peer-to-peer networking for the sending of messages.

One way of doing this is through the use of Wi-Fi Direct™ connection which enables 802.11/ Wi-Fi capable devices communicate with each other without an internet connection or an access point. It makes devices emit a signal to let other Wi-Fi Direct capable devices know that a connection is available. It can be used to send files, sync data, and other things that originally needed internet as long as there is a connection to other devices.

The user-end devices in this research that will act as endpoints of the mesh will be the smartphones which the users will use to type and send their messages. The messages will then be passed through the microcontrollers to a destination device.

The source states however that Wi-Fi Direct makes devices act as access points so a simultaneous two way connection is not possible. This is an area to be researched further as the paper progresses.   
   
**Mesh Network**   
Wireless mesh network is a network setup wherein devices (or nodes), are interconnected with each other. When a connection is initiated, a node acts as a transmitter that carries the connection to the next node. This allows connections to be self-healing or being able to reroute broken connections. Unlike Wi-Fi that only uses only one antenna for all devices connected, wireless mesh is decentralized. Nodes only need their broadcast to reach the next node.    
   
There are two kinds of mesh topology: full mesh topology and partial mesh topology. In the full mesh topology, all nodes are interconnected, while on the partial mesh topology only some nodes are interconnected.   
   
This research will be using a partial mesh topology. Each microcontroller will act as a node that that will be used to carry the connection between two end devices. When an end device sends a message using the chat application, the message will be broadcasted. The nodes that will receive this message will in turn rebroadcast the same message. This process will be repeated until the message is received by the mobile device on the other end.

**Raspberry Pi and Arduino**   
Arduino and Raspberry Pi are microcontrollers that are often compared to one another about its functions and capabilities. Many people say that Raspberry is better than Arduino since it can function as a “normal computer”. Yet there are also many that say such assumptions are “unreasonable" because of what Arduino can do.

Raspberry Pi could function similarly to a computer because it has some features a computer has. It has its own memory, graphics driver, processor, etc. Also, it has its own Ethernet port so networking is more convenient. It is capable of doing works like doing spreadsheet, word, internet browsing, and gaming. It also has its own operating system (Linux). But to fully utilize sensors integrated with it, different software is sometimes required. Same as normal computers, it needs to be turned off properly. Arduino, on the other hand, does not act and function like a computer. It does not have all necessary parts of a normal computer and it does not operate a full operating system. It just runs the codes in it. As networking purposes, it does not have an Ethernet port so if someone wants to use it for networking, the user must do all necessary things to make it happen (includes having external hardware for physical connection applications). For integration of sensors, accomplishing it is easier in Arduino because the “interpretation” & “response” can be done effortlessly with the use of codes. It can work and execute its codes by connecting it to a USB cable to a computer (and any other ways). (Bourque, 2015)

**ESP8266**    
ESP8266 is a microcontroller that can access 802.11 connections. It is Arduino compatible, meaning it can run programs that are created for Arduino and could also be used as an integrated module. It was initially introduced as a Serial-to-WiFi adaptor that supports AT command set. AT command set, also known as Hayes command set, is a language that is made up of multiple short strings that is used for simple operations such as hanging up and dialing in modems. In this research, this microcontroller will be used to give network access to Arduino so that they can communicate with each other.   
   
**Routing protocols**   
The key considerations that raised the need for efficient routing and must be considered when implementing mesh networks are key issues, efficient routing, the overhead of ID per hop jumped, maintenance of nodes, send/receive overhead, power consumption, and interference. It is also important to consider that table based protocols grows bigger as nodes increase and packet header grows bigger as more nodes are included when choosing the proper routing protocol for your network.   
   
The research showed that the performance of a proactive routing protocol is better in client mesh network architecture. OLSR succeeded against two other protocols when compared in terms of efficiency in throughput, point to point delay and network load. There is less need for reactive routing protocols because the topology of a wireless mesh network is relatively static once it has been properly implemented.   
   
This suggestion of OLSR will be taken into consideration when we implement our project. These concepts are key to understanding the efficient transfer of data between nodes across a mesh network. It is important to take note of the key considerations to serve as guidelines before trying to establish data transfer to and from any node that will be deployed in this research. 

**Project Byzantium**

Byzantium is a wireless ad-hoc mesh network which connects devices using 802.11a/b/g/n without relying on the internet. It is a distribution of Linux which can be installed to a device or run from a removable media. Any Wi-Fi enabled computer can be made into a Byzantium node just by running Byzantium Linux. It uses OLSR as its routing protocol. Each of these nodes connects to each other directly, forming an ad-hoc mesh network. If one of the nodes in a network has an active internet connection, all other nodes can use this connection too; that is of course only if the ISP permits.

Byzantium Linux is already available in Github however, the last commit was done back in 2014.

**III. Technical Background**

**OSI Model**

Layered models are usually used in network communications to illustrate the processes involved when data is transmitted. The Open Systems Interconnect (OSI) model is the most known layered model and it acts as an abstract representation of how network enabled devices communicate with each other. It was developed by International Organization for Standardization (ISO) to serve as an outline of protocol stack used by non-proprietary protocols.

The OSI model can be visualized as a diagram with seven layers with each layer receiving message from the one directly above or below it depending on whether the device is receiving or transmitting data. These layers are usually divided into two groups, the upper layers and the lower layers. The upper layers consist of Layers 5, 6, and 7 and their focus is on user interaction and identifying the message so that it will be delivered to the right receiver. These layers are handled in the software side. The lower layers, consisting of Layers 1, 2, 3, and 4, focus on the transmission of the message and are handled by both hardware and software.

The top most layer of the OSI model is Layer 7 or the Application layer. Generally, its services let the end user interact with user applications. It lets the sender of a message interact with the software application to get the desired message into the system. Alternatively, it also lets receiver acquire the message transmitted to him. Examples of software functions that work within the Application layer are file transfer, remote printing, virtual network terminals, and electronic messaging.

The next layer, Layer 6, is the Presentation layer. It is responsible for providing the application the format of a data. Different systems use different data format and protocols. Presentation layer acts as a translator to translate incoming messages to a format that the application uses. Some of its functions are character encoding translation, data compression, and data encryption.

Layer 5 or the Session layer is responsible for managing sessions between devices. It maintains the connection of devices allowing them to start sessions and restart them when the connection is interrupted.

Transport layer, Layer 4, handles the procedure called segmentation/desegmentation. When data is sent to the Transport layer, it is divided into different segments so that they would be transmitted easier. It keeps track of each segment to ensure that all of them are transmitted. Because of the nature of data transmission, segments do not arrive at the right order so the Transport layer reassembles these segments by numbering them. This layer enables the device to have multiple concurrent connections since it divides the data into smaller pieces instead of a one whole stream.

Network layer, Layer 3, address packets with a source IP address and destination IP address that are both stored in an IP header. The process of adding these information is called encapsulation. After encapsulation, a segment is then referred to as a packet which is ready to be transmitted over a network. As multiple routes are usually available for a packet to reach its destination, Network layer handles the routing by calculating the best path.

Data Link layer, layer 2, enables the layers that deal with the software processes, layers 3, 4 , 5 , 6, and 7, communicate with the layer that deals with the hardware of a device, layer 1. Layer 2 encapsulates the packets into frames to format the message into pieces than can be interpreted by different media. This layer is usually divided into two sublayers, the upper sublayer that implements the software processes and the lower sublayer that implements the hardware access. Another role of the data link layer is to control how frames are sent to or received by the media. This is called Media Access Control(MAC). Since the data will pass through different media as it travels from its source to its destination, MAC allows it to be handled by most media and devices.

Physical layer, layer 1, is the only layer that is solely implemented by hardware devices. Its primary role is to encode frames from the data link layer into bits and its transmission as a signal over a physical media such as copper wires and fiber cables. These physical media allows the transfer of signals across different devices as an electrical pulse, light pulse, or radio wave.  

**VII. Appendices**

**References**

<http://www.digitaltrends.com/computing/arduino-vs-raspberry-pi/><https://learn.sparkfun.com/tutorials/what-is-an-arduino><https://www.raspberrypi.org/help/faqs/#introWhatIs><http://www.wi-fi.org/discover-wi-fi/wi-fi-direct><http://stackoverflow.com/questions/29018122/android-pure-p2p-chat-application><http://info.hummingbirdnetworks.com/blog/wireless-mesh-networking-recommended-maximum-distances><http://archive.oreilly.com/pub/a/wireless/2004/01/22/wirelessmesh.html><http://www.idosi.org/wasj/wasj29(3)14/14.pdf><http://robot-kingdom.com/best-flight-controller-for-quadcopter-and-multicopter/><http://quadcoptergarage.com/quadcopter-parts-list-what-you-need-to-build-a-diy-quadcopter/><http://diydrones.com/profiles/blogs/basic-sonar-based-collision-avoidance-for-a-quadcopter><http://www.gizmag.com/solarcopter-solar-helicopter/26645/><http://sts.bwk.tue.nl/7y500/readers/InstellingenRegelaars.pdf><https://www.techopedia.com/definition/575/at-command-set><http://makezine.com/2015/04/01/esp8266-5-microcontroller-wi-fi-now-arduino-compatible/><https://gsmaintelligence.com/research/?file=141201-philippines.pdf&download><http://www.mgb.gov.ph/images/stories/RA_7942.pdf><http://pluto.ksi.edu/~cyh/cis370/ebook/ch05b.htm>

http://project-byzantium.org/