

# BEMOSS: Integrating New IoT Device

Reece Bachman      Jordan Ingram      Robert O'Malley  
Advisor: Dr. Suruz Miah

December 10, 2018

## 1 Project Description

An internet of things (IoT) network is a network comprised of devices that are often not connected to the internet that have been engineered to have the ability to communicate with each other via wireless network. This allows a single node to control countless physical environments, making the process of running and maintaining a building much simpler. An IOT network will generate a large amount of sensor data that will allow neural networks to be developed for the purpose of smart control of devices. This is the idea behind big data which is a trending topic due to the popularization of internet of things as well as the decrease in the price of memory.

The current project is aimed to create a building energy management IoT devices by using Building Energy Management Open Source Software, also known as BEMOSS. We are using BEMOSS as the platform that the devices we make will connect on. Our project is to create the ability for BEMOSS to control a DC motor and to implement a control algorithm to efficiently manage the IOT devices and energy within the network. Having the ability to control a DC motor gives us the opportunity to close curtains, doors, and blinds. Through this, we can further control the climate of a room and building. If a room routinely gets hot around noon, our algorithm would turn on a motor to close the curtains in the room and block the room from the sun. If the building's exterior temperature is below the desired interior temperature, the motor would be used to keep the curtains open as long as possible to increase the usage of the sun's heat. This would add additional cooling and heating to the environment with very low energy consumption. Security could also be improved through this IOT connected device by closing doors during emergencies such as fires and intruder situations.

The project starts with a very simple attempt to understand BEMOSS and try to get a supported device to work first so that we can understand how to use BEMOSS. Once we have an understanding of how to work with BEMOSS, we shall start doing our own interfacing and working with our own parts.

As of September 24, 2018, the BEMOSS does not currently support many electronic devices, such as motors through its IOT In this project, we aim to create an interface device

using hardware/software co-design that will be able to fully control a motor from BEMOSS and it is expected to be added to the list of supported devices of the BEMOSS. In doing so, we create a wireless receiver and emitter so that we do not need to connect an expensive single board computer up to the motor just to control it. Once we are able to successfully complete that, XBee will be connected to the single board computer and sends the load commands entered into the system via the control center.

## 2 System Architecture

Figure 1 shows the high level system architecture of the current project on customized BEMOSS architecture. As it can be seen from Figure 1 that the proposed BEMOSS architecture consists of five main blocks. The portion at the head of the architecture is the "building energy management center" which is responsible for the command and control of the network based on energy requirements and user demand. The commands are conveyed through a TCP/IP building network. From there, the commands from the TCP/IP network are collected and processed in a wireless sensor central node. This node is responsible for delegating the commands to the array of individual loads in the building through radio frequency communications. The individual loads consist of, but are not limited to, HVAC units, lighting units, fans, and actuators (for example DC motors). These loads are connected to a power grid and operate at the commands received from the central node.

The building energy management control center is run on a computer which is running BEMOSS to communicate with the loads. The wireless sensor central node we are using is an XBee in combination with a single board computer. We are using Raspberry Pi as the single board computer to receive commands from the TCP/IP network and relay the proper load commands to a XBee which communicates with the XBees attached to the loads.

## 3 Modes of Operations

The proposed generalized BEMOSS architecture is expected to operate in three different modes: Mode #1 (default), Mode #2 (user-defined), and Mode #3 (energy-efficient usage). A brief description of each of these modes is provided below.

**Mode #1 (Default):** This mode will run on non-optimized usage. The plug-and-play devices will be allowed on and off as the user decides to use them. The user will be able to turn their devices on and off through the BEMOSS server.

**Mode #2 (User-defined usage):** This mode will allow the user to pre-define when they would like the devices to be turned on and off. The user will be able to automate device usage times.

**Mode #3 (Energy-efficient usage):** This mode will use a neural network algorithm to automate efficient energy usage.

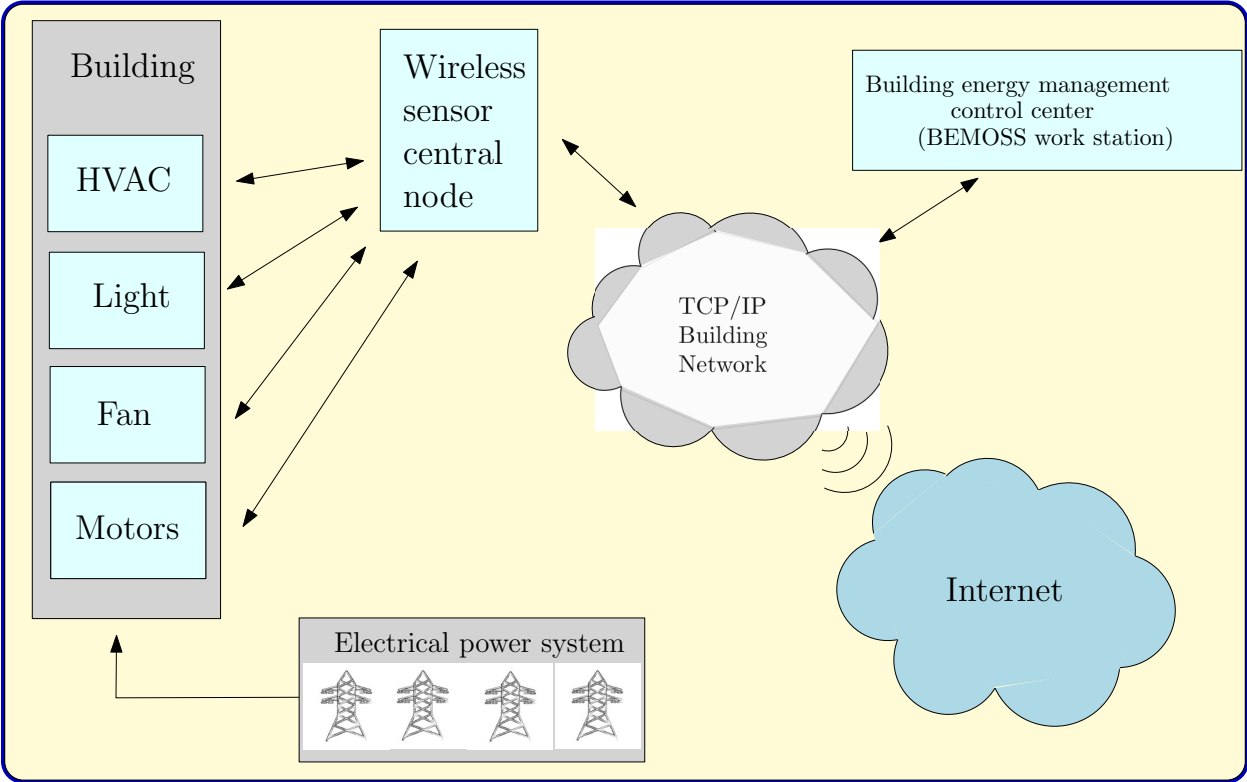


Figure 1: High level system architecture of the proposed BEMOSS architecture.