The First Generation WIMEA-ICT Automatic Weather Station Prototype: An Evaluation Report

Maximus Byamukama [1], *Mary Nsabagwa*[2], *Emmanuel Kondela*[3]

Revision 4

November 18, 2015

[1]College of Engineering, Design, Art and Technology,Makerere University, Uganda
[2]College of Computing and Information Sciences, Makerere University, Uganda
[3]Department of Computer Studies, Dar-es-Salaam Institute of Technology, Tanzania

Contents

List of Figures

List of Tables

List of Acronyms

AWS Automatic Weather Station(s)

I2C Inter-Integrated Circuit

MCU Microcontroller

PCB Printed Circuit Board

SPI Serial Peripheral Interface

UART universal Asynchronous Receiver/Transmitter

RS-mote Radio Sensor Mote

TMA Tanzania Meteorological Authority

SSMD South Sudan Meteorological Department

TTCL Tanzania Telecommunication Cooperation Limited

UNMA Uganda National Meteorological Authority

WSN Wireless Sensor Networks

Abstract

One of the objectives of the WIMEA-ICT [4] project, a NORAD initiative is assisting National Meteorological Authorities in increasing the density of Weather Station Networks in Uganda, Tanzania and South Sudan. To achieve that objective, a total of 70 AWS shall be designed and deployed in the three countries. Before the mass production of the AWS, three prototypes shall be developed. The first-generation Prototype was deployed in Bergen, Norway hence the name "Bergen prototype". This report presents an evaluation of the first-generation prototype.

Needs and requirements of AWSs form a basis for the evaluation. In this report, we discuss strengths and weaknesses associated with the Bergen prototype. We review literature on Wireless Sensor Networks, power management and uplinks to provide informed design decisions. An assessment of uplink options, power management and Wireless Sensor Networks is done. We propose a design for the second-generation prototype while categorizing design decisions under data capture and transmission, which are the main functional requirements. We base our design on attaining a robust and yet cost-effective AWS. We provide an analysis of our design, conclude the report with recommendations for improvement and also suggest future work.

[4] The full form for this Acronym is Improving Weather Information Management in East Africa for effective service provision through the application of suitable ICTs

1 Introduction

1.1 Background

The purpose of this report is to make a review of the first WIMEA-ICT prototype as described by the working assumptions agreed upon in [1]. The prototype is WIMEA-ICT project's first version of three Automatic Weather Station prototypes. The project aims to improve the accuracy of and access to weather information by communities in the East African region through suitable ICTs for increased productivity. The project is made up of five components, the main component being research, teaching and capacity building. Four other components are organized around this one [2]. Research Component 3 (RC3) has its primary objective as assisting National Meteorological Authorities to set up more AWS in Tanzania, Uganda and South Sudan. We initiated the design of an affordable and robust AWS with a first-generation prototype also called The Bergen Prototype. Before mass production of the AWS, we planned to design and test three prototypes the first being subject of this paper. The first-generation prototype is already operational [3] and is providing us a basis for refinement and from which we shall derive the second generation prototype, which we refer to as *The East African Prototype* in this document. While the first-generation prototype performs the basic AWS functions, we still target to improve a number of performance requirements in order to attain an optimal design. To this end, we discuss requirements of AWS and refer to them for justification of the evaluations and recommendations we make.

The rest of the document is organized as follows:- We begin with needs and requirements discussion in this section. In section 2, we make a review of the first-generation prototype, in 3, we discuss work related work, in 4, we discuss our proposal, in 5 we make an analysis of our design decision and we lastly conclude and give areas that need to be investigated in future work.

1.2 Needs and Requirements of an Automatic Weather Station

An Automatic Weather Station(s) (AWS) is an instrument that measures and records meteorological parameters using sensors without human intervention. An AWS is composed of a processing unit (CPU, volatile and non-volatile memory, and I/O interface), connected to a number of meteorological sensors, and communication unit.The World Meteorological Organization (WMO) recommends standards and requirements for Automatic Weather Stations. The recommendations are a basis for assessing performance of the first-generation prototype. Additionally, we refer to several other requirements derived from our previous country surveys and in consultation with Uganda National Meteorological Authority (UNMA), Tanzania Meteorological Authority (TMA) and South Sudan Meteorological Department (SSMD)[4],[5],[6]. Many AWSs fail to function because of failure to meet the non-functional requirements. We list the functional requirements and explain the minimal non-functional requirements below which the AWS might become non-functional. Below are the core functional requirements expected of an AWS

- Capture Weather Parameters: These include temperature, humidity, pressure, wind speed and many others
- Process captured data: Processing involves converting analog weather parameters into a digital equivalents, calculating derived information such as dew points, data compression and much more.
- Buffer data: Temporary storage of data in either volatile or non-volatile memory before it is moved into permanent storage.
- Transmit data: From AWS to a repository.

Many Automatic Weather stations fall short of the above functions for failure to achieve the non-functional requirements. We give a discussion of the non-functional requirements, which we recommend for the second generation prototype. Below are the non-functional requirements we recommend for the AWS

- **Dependability**: This is defined by [7] as the trustworthiness of a computer system such that reliance can justifiably be placed on the service it delivers. The authors point out five attributes of dependability including availability, reliability, safety, integrity and maintenance.
- **Affordability**: In regards to operational costs, the AWS should be affordable for Tanzania, Uganda and South Sudan. Some components shall be acquired locally to reduce on transportation costs. We want to cut down on maintenance costs by designing a robust AWS. The AWS costs should fit within the budgets of both WIMEA-ICT and that of the National Meteorological Authorities.
- **Usability**: The AWS should be easy to install and configure.

- **Scalability**: The system should allow unrestricted sensor node placement. It should be easy to expand the network of sensor nodes.
- **Integrity**: The data to be sent should be accurate and consistent and the processes of storing this data and fetching it should be error free.
- **Power consumption**:Power consumption should be as low as possible.

Table 1 shows our recommended minimal requirements for the AWS.

Table 1: Quantification of Non-Functional Requirements

Requirement	Notes	Minimum Bounds
Availability	Readiness for correct service	Sensors Must be available to collect data at worst every one hour
		In case of an abrupt change in weather conditions, measurement of weather parameters must be immediate (SPECI[5])
		Climatological and Agricultural weather stations must send data at least once a day
		Rainfall, Maximum and Minimum temperature, soil temperature must be sent at least once a day
		Wind Speed and direction should be collected at most after 20 minutes
Reliability	continuity of correct service	Hardware components must not fail within 1-2 years
Maintainability	ease and speed of restoration to operational status after failure	Detecting, diagnosing and recovery should not exceed one hour
		AWS should be checked twice every year even if no fault is suspected
Integrity		Format of readings must be consistent and differences of measured data from general expected values must not be significant under normal circumstances.
Power		Power consumption low as possible
Scalability		An unlimited number of nodes should be able to be placed in the network.
Affordability	cost relative to available funds	Within Budgets of WIMEA-ICT and National Meteorological Authorities

It is from the above requirements that we base subsequent sections to evaluate the Bergen Prototype and make design decisions for the East-African Prototype.

[5]SPECI is special weather report issued when there is significant deterioration or improvement in weather conditions

2 Problem Definition

There are several problems to be addressed that can be looked at individually and in light of others. Power consumption, power supply design, data transmission and data accuracy can all be improved for the East Afrincan prototype.

2.1 Power Supply

The Bergen prototype uses solar panels to charge supercapacitors that provide power to the circuitry. Insolation in Bergen is much lower and less available than in East Africa and as such, the power supply can be optimized in several ways. One, the size of panels can be reduced. An active research area of one of the PhD students is the smallest possible panel size that can sustain an AWS given the solar irradiance in Uganda. Reducing the panel sizes is also of particular importance because it is a way to deter vandals of AWS equipment as this was found to be a serious challenge for UNMA and TMA in the Weather Station Survey carried out between October and January 2014 [4], [5]. Secondly, some power electronics in the Bergen prototype, such as the step-down DC-DC converter, could be eliminated if such panels, as discussed above, can be used to charge the supercapacitors directly. The availability of more solar energy also provides an avenue to increase the functionality of the AWS. For example, by implementing designs that benefit from a gateway that is ever powered, such as hosting web services or cameras for security.

2.2 Power Consumption

We believe power consumption of the Bergen prototype can be reduced substantially by changing the current gateway, the Raspberry Pi. Sources from the Internet [8] and experiments [9], the bare board consumption is about 1W. Power consumption can be reduced in three ways. First, by changing the gateway platform from the Raspberry Pi to a more power lean one. Secondly, some functionality could be traded off for better power performance; for example by transmitting some data less often. Section 5.1.1 under analysis discusses this in further detail. Thirdly, by implementing the first and second option simultaneously.

2.3 Communication

The Bergen prototype uses a Linux operating system and as such will support a vast number of communication devices with the right drivers. The current implementation at Bergen uses Ethernet but can also use USB based devices such as WiFi or cellular modems. This sub-problem is concerned with two things. First is the most reliable and cost effective way of getting data from the AWS to the central repository given the conditions where the AWS is installed. An algorithm can be developed to enable the choice between cellular network connection, other terrestrial wireless options (VHF/UHF/SHF), copper wire, optic fibre, satellite , sneakernet or some combination of some of these. The selection has to be done keeping in mind the consequences it may have on the reliability of the transmission option, the short and long term costs of using and maintaining the said option, the power consumption, integrity of the data and other issues. Second is the implication that a change of the gateway device could have on the uplink devices to be used. Less advanced embedded systems without in-built controllers for standard interfaces such as USB or Ethernet will require external controllers and, hence, developing drivers for these controllers.

2.4 Data accuracy

Comparison of the Bergen Prototype data and a standardized Automatic weather station, while showing a good correlation, still exhibits some degree of variance in some regions. Figure 1 shows a scatter plot of solar insolation from the BPW20R sensor of the Bergen prototype versus the standardized AWS. The causes of such variance is a case for further investigation in conjunction with RC1.

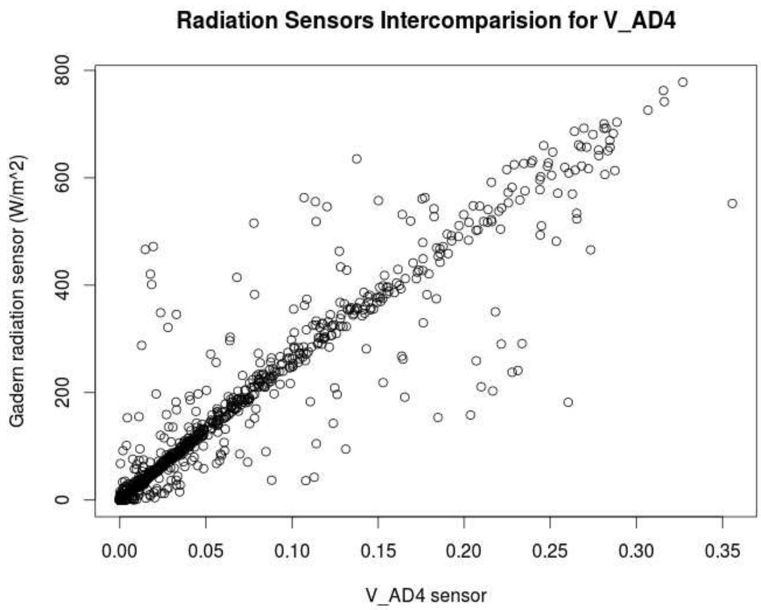

Figure 1: Scatter plot of solar Irradiance measured by Standard AWS versus Bergen Prototype

3 Related Work

In this section, we discuss the research that is being put into the design and deployment of Automatic Weather Stations in both industry and academia, the various technologies in use, the opportunities presented and some challenges faced.

3.1 Wireless Sensor Network

The concept of environmental monitoring using AWS is steadily becoming synonymous with Wireless Sensor Networks (WSN) around the world, especially for remote installations of AWS, such as [10], and IoT applications [11]. WSNs offer several advantages, including unmanned operation, easy deployment, scalability and real-time monitoring among others. Better still, they can be deployed in hard-to-reach areas. The deployment of WSNs in environmental monitoring further gained momentum after the ratification of IEEE 802.15.4 standard in 2003 (and later in 2007) [12].

A similar project is presented in [1], which consists of a WSN where each node is based on the Atmega128RFA1 chip [13], with a built-in 2.4GHz Radio Frequency(RF) transceiver using the IEEE 802.15.4 link protocol. In this installation, the nodes wake up according to a schedule, broadcast packets with the sensor data and go back to deep sleep mode. The data is captured by a sink-node which is always awake. Networking of the nodes is simplified by the use of Rime, a lightweight layered communication stack for sensor networks[14]— and is in this case used on the contiki operating system.

The main challenge in WSNs is power consumption, limited storage and computing resources [15],[16]. These challenges are the center of most of the research in this area. Some other challenges of WSNs and their solutions have also been discussed in a wireless sensor network survey by Yick and others [17]. There have been various efforts to reduce power consumption of the radio such using renewable power sources [18] and duty cycling[19]. While most of such proposals improve power saving, they still cause a degradation of quality of service and their implementation is not trivial. Optimizations are made on a selected set of layers and this may only offer energy conservation on individual layers. Cross-Layer duty cycling has also been proposed [20].

3.2 Data Transmission

An AWS may use a set of various uplinks with variation in transmission requirements such as transmission power and cost. The common uplinks that have been used to transfer meteorological data to the central monitoring systems are wired links [21], terrestrial wireless or satellite links. The wired links realize the transmission of meteorological data with higher Quality of Service, but the deployment cost is relatively high and it is difficult to deploy cables in unfriendly environments, such as mountains. Satellite transmission offers wide coverage, but may be more expensive than terrestrial wireless transmission. 6 gives an example comparison of the cost of transmitting data via cellular networks and satellite.

In [22], the sensor nodes use the IEEE 802.15.4 link protocol to transmit data over a short distance.A GSM/GPRS module is used as the uplink device. However, the GSM module consumes a lot of power and drains the source within hours. Similarly a project called TAHMO [23] has been developing robust network of AWS in Kenya. Each station has GPRS-enabled data logger. Other similar environmental monitoring systems have tried the option of using GPRS modems, and have faced a challenge of stable connectivity [24]. It was found that the modems tend to lock up after extended periods of time (2-4 days) and can be recovered only by power cycling. The majority of the AWS in use in Uganda and Tanzania are supplied by DAVIS, WAGTECH, CASELLA and SEBA and use GSM/GPRS [4], [5].

Besides GSM/GPRS networks, the work in [25] compares alternative wireless links for the transfer of data from sink nodes of remote WSNs to a central repository. Experiments have been made using two of the amateur radio bands, the 144 MHz band and the 433 MHz band. The parameters studied include throughput, range, power requirements, portability and compatibility with standards. Similar environmental wireless sensor network projects have been known to use off-the-shelf solutions such as [26] and [27]. A 900 MHz radio modem called FreeWave Ranger is used in these projects. This device can give 115.2kbps throughput with about 90 km range with clear line of sight. Similarly, the work in [28] presents a vision for a new generation of amateur radio networking. The work offers an opportunity to design, implement and deploy new narrow-band, wireless network protocols, and to experiment with new applications. These applications connect seamlessly with the Internet, enabling amateur devices to appear and be part of and directly accessible from the Internet. This Internet connection forms a unique large-scale test-bed that will enable amateurs and others to use these new networks to collect data from meteorological and other environmental sensors that are distributed over large, remote areas and make these observations available in real time. Vaisala provides for using such RF solutions in their products. In [29], a Vaisala HydroMet Automatic Weather Station MAWS201 is used to transfer a record of meteorological data using a UHF radio modem.

Satellite communication offers long distance transmission but may use more power than terrestrial options. In [30] a long-range environmental monitoring system, uses a Skystar 360E modem and 900MHz ISM Band to transmit meteorological data via KU-band satellite link with a high reliability of 97% with an interval of 10mins.

3.3 Power

Automatic weather stations are powered using a wide range of techniques. In accessible areas with grid power, the stations may be connected to the grid and in remote areas, they are powered by batteries charged using either solar power or wind. Moreover, grid-conected AWS in the african context need some form of back up power because grid power is unreliable. In [31] the Australian Bureau of Meteorology shows a typical AWS used by them that could be powered by AC or other options. In the first generation prototype, the sensors are powered by supercapacitor batteries being charged by solar energy. The use of supercapacitors to power wireless sensor nodes is steadily growing, with some work cited as early as 2006 [32] and some even deployed in the particular case of an AWS[33].

4 Design Proposal

Following from challenges of the Bergen prototype, lessons from similar systems and requirements, we now propose a design for the East African prototype. We classify our design proposal into data capture and transmission, power supply design and management, general issues and finally, an overall system architecture.

4.1 Data Capture

Under data capture we discuss the hardware used in sensing.

4.1.1 Sensors

The sensors to be used have already selected by Research Component 1 (RC1) with the exception of a rain gauge. So far, there is no indication of any serious design constraints the sensors may pose to our design proposal. Table 2 shows sensors to be used in the East African prototype. The parameters to be measured are listed in section 3 of the 2014 WIMEA-ICT progress report [1]. The RC3 concerns on this matter are:

- Type of sensor (Active or Passive): This determines whether the sensor will need power to operate or not and if active, how much power
- Electrical / Electronic interfaces supported by the sensor: The Microcontroller (MCU) processing measurements must have support for these interfaces
- Distance of the sensor from measurement point: This determines the length of conductors and possibly affects data integrity.

Distance of sensors from the ground is crucial for this design and as such is also included in table 2 below.

Table 2: Selected Sensors for use

Parameter	Sensor	Distance From Ground(m)
Temperature	SHT25	2
Relative humidity	SHT25	2
Atmospheric Pressure	MS5611	2 or 10
Solar Insolation	S1223, VTB8440,BPW20R	10
Dew Point	Calculated	
Precipitation	To be Decided	0
Wind speed	InspeedWortex Series II	10
Wind direction	InSpeed E-VANE2	10
Soil Temperature	DS18B20	0
Soil Moisture	VH400	0

The first-generation prototype uses an RS-mote mote [34] which has two analog inputs among other components. We shall use the same for the East African prototype. All sensors will be connected directly to a wireless-enabled microprocessor unit on a Printed Circuit Board (PCB), which will capture and transmit data to the gateway. For sensors at a distance from the PCB, such as the soil moisture sensor, the PCB tracks will be extended by flexible wire to the sensor location. A major consideration to make in this case is avoiding contact of these wires with the ground as much as possible. During the baseline survey in Uganda, there were reports of stations that failed because a wire was cut during lawn mowing.

4.2 Data Transmission

Under this section, we discuss transmission media for both uplinks and sensor to gateway links.

4.2.1 Sensor-Gateway link

Sensors will communicate with the gateway wirelessly using IEEE 802.15.4 link protocol. This protocol is specifically for low data rate and short-range wireless networking[35]. Frames of 127 bytes will be sent out at configured time intervals, one minute being the default. Sensor networking operations will be done using the RIME stack, with 16-bit addressing.

4.2.2 Gateway

In this section we discuss the gateway device and well as the uplink options available for the AWS. We have chosen ATMEGA128RFA1 for the gateway because it contains an integrated IEEE 802.15.4 transceiver and consumes very little current—-less than 200nA during deep sleep mode [13], a feature which will be useful for radio duty cycling in order to reduce power consumption. Also, contiki has been ported to this device, and more specifically, the Radio Sensor Mote (RS-mote) [34] and is already being used in the current prototype. There are also other off-the-shelf prototyping boards for rapid development such as the ATMEGA128RFA1 development board from Sparkfun [13]. Contiki was chosen because of its small memory footprint among other advantages. We shall use the same operating system for Version 2 of the East African prototype.

The gateway will be programmed to perform the following tasks:

1. Receive radio frames. This also includes reading from any sensors attached to it.
2. Buffer radio frames in the device's on-chip memory.
3. Save the frame as a string into an existing ASCII text file
4. Upload the frame using a selected uplink through one of the standard interfaces
5. Set the next wake up time and go to sleep

The firmware architecture that will meet these needs and requirements is proposed in section 4.4.

The East African Prototype shall support the use of all types of uplinks listed below.

1. **Wire (Optic Fibre or Copper)**: Cellular service providers in both Uganda and Tanzania have a copper wire infrastructure. Tanzania Telecommunication Cooperation Limited (TTCL), and **UTL!** (**UTL!**), have copper as a core part of their network, and its use for private services can be arranged if the terminal equipment is close to an access point. The main disadvantage is that this service is only available in urban areas or rural townships. Other commercial cellular service providers like MTN Uganda are phasing out these links and replacing them with optic fibre for higher throughput.

   Using optic fibre as an uplink may be a viable option. Excluding commercial cellular network providers that use optic fibre as part of their infrastructure, there is a national optic fibre backbone in Tanzania [36]. The Ugandan Government and Google are still in the process of laying fibre cable [37],[38]. For the larger part, the use of such uplinks and the tariffs to be associated with them requires agreements between meteorological authorities and the service providers or government.

   If the gateway is deployed indoors for reasons including security or access to grid power, and a wired connection is available, then the uplink can be implemented using a protocol such as Ethernet. In such a case, the gateway can be connected via an Ethernet controller. One such device that was reviewed is the RTL8019AS [39] from RealTek which is a full duplex Ethernet controller. A ready to use module employing this controller and an ATMEGA128 on a single PCB has been found to be readily available [40]. The RTL8019AS is also supported by Contiki.

2. **Commercial cellular service**: These have seen an exponential growth in East Africa, with a penetration rate (subscriber-based and not geographical reach) of 53% in Uganda [41] since their introduction in the 1995 and 69% in Tanzania[42]. Such high penetration rates, and the existence of service providers in competitive oligopolies which lower prices, make these networks major contenders in point to point communication nationwide and as such, they are the main uplinks in use for AWS in Uganda and Tanzania. Terminal equipment is readily available and no additional licensing is required from end users. As such, we looked at a number of GSM/UMTS capable devices that can be used in the East African Prototype.

   Because of the proposed gateway, only non-USB devices have been considered. Several modules have been investigated including SIM900 [43] and SIM800 [44] series from SIMCOM, the SARA-G3 series [45] and LISA-U2 series from U-Blox [46] and the intel XMM 6255 [47]. The active power consumption for all these modules in 2G mode is roughly similar (300-500mA) and all have sleep currents below 1mA. The SIM800 series has a readily available compact breakout board(2.5cm x 2.5cm)[48] and has a wide development community. It is also relatively cheap (about $9). The boards support standard AT commands and manufacturer enhanced AT commands for specific functions. All support at least one standard interface on the ATMEGA128RFA1 such as universal Asynchronous Receiver/Transmitter (UART) (RS-232 port and Serial Peripheral Interface (SPI)) and Inter-Integrated Circuit (I2C).We recommend that several of these be tested before a final decision is made.

3. **Dedicated RF**:UHF/VHF as described in the IEEE 802.11af specification summary [49] and other dedicated RF transmission techniques can be used under two scenarios. That is, as a last mile solution from the AWS to a core network such as optic fibre or commercial cellular network [50] and as a direct uplink to a gateway within range.

   The selection of an appropriate VHF/UHF device will depend heavily on the location of the AWS. The antenna height and transmission power will depend on the transmission distance required and the nature of surroundings (such as vegetation). The appropriate devices are left open for now. However, some devices that were looked at and can be investigated for short ranges (1-2 km clear Line of Sight (LOS)) include those based on the PT2272 and PT2262 pair from Princeton Technologies Corporation [51] and the XBEE Pro 900HP, a long range 900MHz RF module that, according to the manufacturer transmits up to 40km LOS with a high gain antenna [52]. These both expose a serial interface (RS232) that the micro-controller can use and could be suitable for scenarios in which the gateway is powered from mains (for example indoor gateway deployments) such that the power required for these long distances is not a critical issue.

4. **Satellite**: Uploading data via satellite is an option to consider in the absence of cellular network coverage, and more so in locations where there are no other uplink types available, such as high altitude rugged terrains. We looked at available options from Iridium [53] and NuPoint systems [54] including their tariff plans. The devices are quite expensive although the tariffs [55] seem acceptable, depending on the service provider. The Iridium 9602 can be used for short burst machine to machine communication and costs between 250 to 500 USD [56]. It is portable and exposes an RS-232 compatible interface. The systems at NuPoint did not have any prices attached. In Uganda, some projects and schools use VSAT (Very Small Aperture Terminal) for dedicated internet service, such as that offered by Uganda Telecom [57].

5. **Physical Data Transfer (Sneakernet)**:An SD card will be used over the SPI interface of the host device. All local data will be stored on this card. A human observer will be able to retrieve this card and save it to a computer. Other viable options are using mobile devices as data mules to which the data can be uploaded when the device is in the vicinity, for later uploading to the central repository when the device gets an internet connection.

4.3 Power Supply

The power supply design is expected to change because of differences in solar intensity and duration between East Africa and Bergen. A lot of work has already been done using supercapacitors to power the RS-mote [58] and other gateways [9]. For deployments that could be powered directly from the grid, we propose the use of a back-up source such as a supercapacitor or another appropriate battery. For remote deployments, we propose to retain the use of supercapacitors as the primary energy storage. Secondary energy source could be another battery or the solar panel in a direct connection, if, say, the storage devices are full. Some preliminary power-related designs for hybrid power sources are being investigated by undergraduate students at Makerere and we expect that results from these experiments will be ready before June 2016. Figure 2 shows a power supply design schematic for such a hybrid system. The system is modular and in its simplest implementation can be a solar panel together with a DC-DC converter charging a supercapacitor. This is the case in the Bergen prototype implementation. In its full form, it consists of an MCU-based control switch that changes the energy source depending on the requirements of the gateway and uplink device. The size of the energy source could be chosen to sustain an installation for a justified time period, assuming worst case scenarios such as a solar panel disconnection.

The power consumption of the sensor nodes is much lower than the gateway. There is evidence of a mote going six weeks on a single charge of the supercapacitor [58]. Because of such a low consumption, we propose that the power supply design for the sensor nodes remain the same, with the exception of determining the appropriate minimum capacitor and solar panel sizings given the insolation in East Africa.

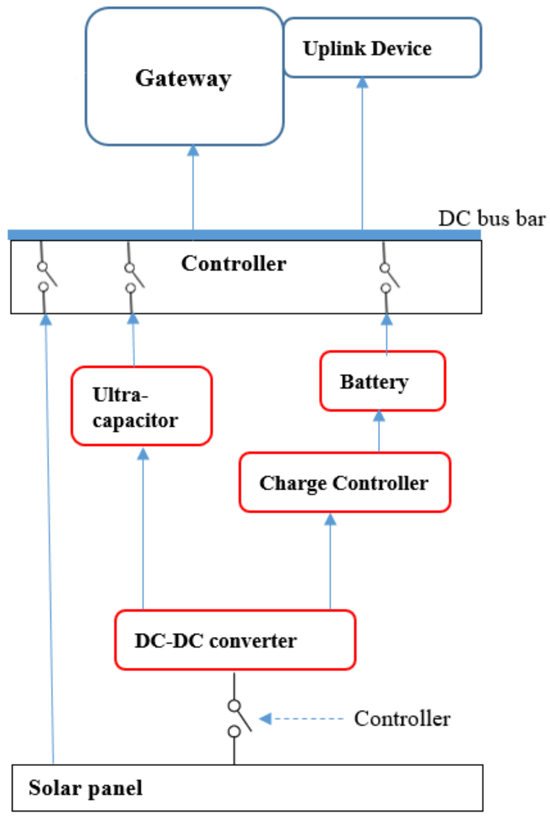

Figure 2: Hybrid power supply concept

4.4 System Architecture

We now give an overview of the software and hardware architecture to be implemented. The implementation details of these design proposals are suggested in the implementation section of this document.

4.4.1 Software

There will be four code modules to perform the functions of receiving a frame, storing the frame, transmitting one or more frames via the uplink and configuring the device via a terminal. The firmware for the gateway and the wireless motes will be the same, with the possibility of disabling some functions and enabling others via terminal commands. Figure 3 shows how these modules interact.
The terminal module will be used to interact with the MCU to alter any configuration settings. Configuration settings include transmission intervals, which sensors to read, etc.
The Read module will read the sensor data, process it where necessary and send to the buffer. Processing involves making appropriate conversions from ADC values to meteorological values, etc.
The Receive module will receive frames from surrounding motes and buffer them, to later be accessed by the Storage module.
The Storage module will retrieve the buffer contents and save them on external memory.
The Transmit module will read unsent data from the buffer or storage and upload them over the uplink.
In the figure, parallelograms and dotted arrows indicate inputs into system. Rectangles indicate software modules and rounded-corner rectangles indicate storage locations. All arrows indicate flow of data.

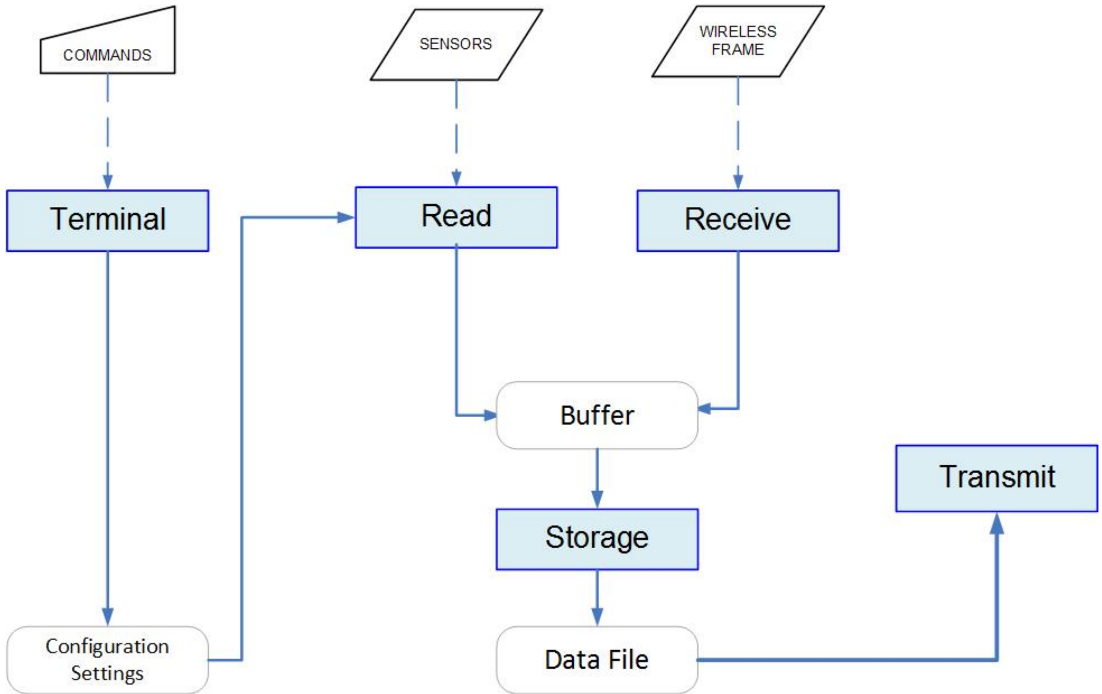

Figure 3: Firmware architecture for gateway

4.4.2 Hardware

The actual peripherals to be interfaced to the MCU will depend on whether the device is acting as a gateway or a wireless mote. The mote will simply be interfaced with sensors and programmed to broadcast data frames. The gateway, in addition, will feature external memory and controllers for standard interfaces such as USB and Ethernet for the various uplink options. Figure 4 shows the typical peripherals for the gateway MCU. A number of the peripherals, such as UARTs, are on-chip. These are indicated as text inside the MCU.

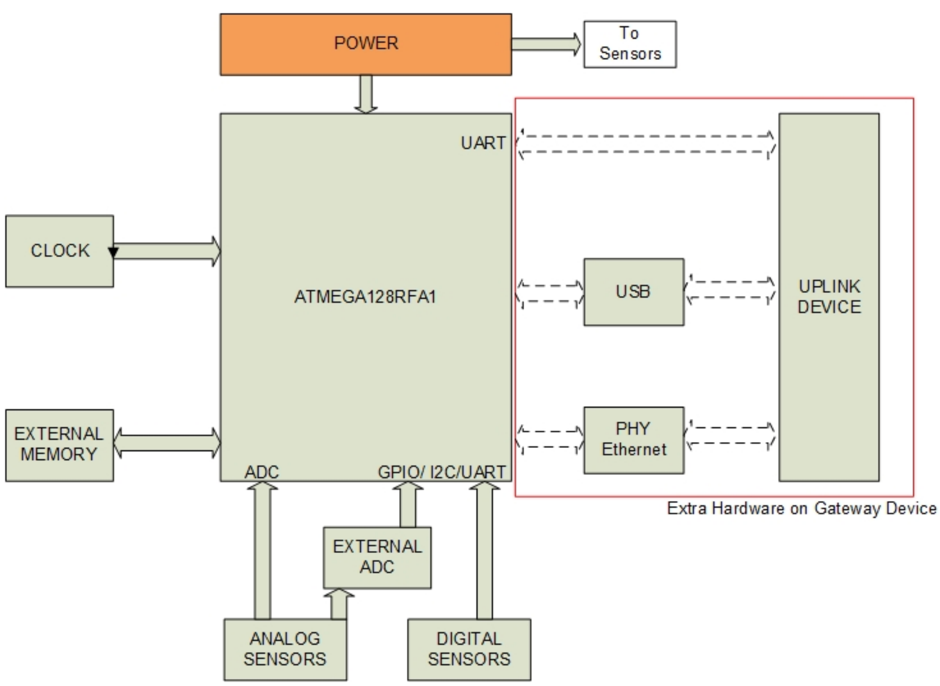

Figure 4: Hardware architecture for gateway

The hardware architecture diagram is the same for the sensor nodes and gateway only differ by the extra hardware shown and its connections to the MCU. In this extra hardware schematic, the dotted arrows represent optional connections, since support for various uplink devices necessitates the use of controllers for various standard interfaces.

Analog sensors such as the VH400 for soil moisture may be connected directly to an ADC which may be internal or external. Digital Sensors such as the MS5611 and devices (such as the external ADC) may be connected to any of the General Purpose Input/Output pins (GPIO) or internal peripherals such as I2C or SPI depending on the interface they use.

4.4.3 System

In a typical full deployment, we have 4 nodes termed as the sink, 2m , 10m and ground node. Each node measures a particular set of parameters following the working assumptions agreed upon in Kampala in 2014 [1]. This is the way the Bergen prototype has also been implemented.

The overall architecture of the system to be deployed will depend on the weather data required from that particular location, the level of redundancy required and even the available connectivity to the network (which may bring the need for relay nodes). Some AWSs will measure only one or just a few parameters while others will measure the complete set of parameters. As such, the number of wireless sensor nodes to be deployed per AWS will vary.

4.5 General Design Issues

The mechanical frame, which will hold the sensor nodes should be strong and corrosion resistant. We propose designing tripod masts made of stainless steel, aluminium, carbon fibre and galvanized iron. Because the metal fabrication industry is very well developed in East Africa, we recommend that the mechanical frame be manufactured locally.The 2m mark should have a flat panel with screw holes to enable the enclosure to be tightened. We recommend using an adjustable panel to ease adjustments in orientation. Material for the pole on which the 10m node will rest must be light and rigid. It also needs to be affordable. A needs and requirements document for this item will be prepared once the the prototype components are ready to be tested outdoors.

Radiation shield design for the node that hosts temperature and humidity sensors has already been started and sample shields have already been printed. One good idea that has been shared is to have a solar insolation sensor on top of the shield. The design of these shields will be a collaboration between RC1 and RC3. The WIMEA project can arrange for the purchase of a 3D printer to enable fast prototyping. Such design could be made a task for undergraduate students to enable faster progress.

We suggest the use of standard drill bits and standard cable glands to create IP66 (or better) compliant enclosure boxes for the electronics. IP66 is a chosen rating because the tropical downpours in East Africa could be sufficiently strong and last a long enough time to penetrate the enclosures.

5 Proposal analysis

We analyze our proposal with reference to the non-functional requirements discussed in section 1.

5.1 Power Consumption

It can be seen from the data file of the current implementation [59], that about 5 packets are received in a 1 minute time frame. The frequency with which the packets are sent increase power consumption. If frequency with which packets are sent is decreased, the gateway will be idle for a larger fraction of time. This calls for implementing power saving strategies like radio duty cycling leading to a substantial power consumption reduction.

The ATMEGA128RFA1 consumes less power than the Raspberry Pi. It also supports a sleep mode and will permit the radio duty cycling mentioned above. A major disadvantage of the Pi is that it runs a Linux operating system, and to implement radio duty cycling on it, we'd have to shut it down and reboot it each time because it does not currently support sleep or suspend functions. This leads to some time wastage in the boot process of the order of 10-30 seconds [60]. Some packets from the sensor nodes could be sent so close to each other that there would not be enough time to power cycle the Pi to and from its off state. The power consumed during this time is also not used to do any real work. On the other hand, a contiki-based ATMEGA128RFA1 has wake-up times between 10 and 650µs, depending on a number of factors [61].

5.2 Scalability

The nodes broadcast data that is picked up by any node configured to be a sink node. As such, the prototype will be highly scalable. Any new node to be added can simply be brought into the vicinity of the sink node and its data will be captured.

5.3 Reliability

An actual measure of reliability is difficult to attach to the system before implementation because several factors, outside the control of engineering design, may affect it. Nonetheless, the design proposal addresses reliability in a number of ways.

First, the nodes will be close to the gateway in the network so there is a rough guarantee of receiving all transmitted frames at the gateway. Second, the power supply will be designed to sustain the AWS for a justified time period and this is complemented with the fact that the solar energy will be readily available and abundant. As such, we expect to minimize off-time due to low power reserves. Also, the fact that the system will be scalable makes it easy to install redundant motes and gateways to decrease the probability of losing data. This is further complemented by the use of multiple uplinks on the gateway. The sensors have also undergone rigorous testing and will be deployed as close to WMO specifications as possible. Throughout the whole design proposal, the techniques chosen ensure reliability of the sub-components and this is expected to bring out a robust AWS system.

5.4 Data Buffering

The AWS availability affects the buffer space required. ATmega256RFR2 provides for buffering. This space is small and may not accommodate all data collected when the radios are unavailable. We propose at least 1 Gb of space to buffer extra data.

5.5 Cost

We evaluate cost in terms of investment, deployment, operational and maintenance costs. 3 , 4 and 5 show the breakdown of the various costs that may be incurred and estimates where possible.

Table 3: Deployment costs

Item	Cost (USD)
Land Acquisition	Depends on location and Land policy
Setting up concrete bases	500
Transporting Weather station and engineers	Depends on location
Payment for casual laborers (per AWS)	100
Fencing	100

Deployment costs are based on a 4m x 4m base for an AWS and average prices of materials in Uganda, Tanzania and South Sudan

Table 4: Operational Costs

Item	Cost (USD)
Observer salary	100
Engineer's salary	500
Uplink data usage	Depends on type of uplink and amount used
Transport	Depends on destination and fuel prices
Power (for grid-connected stations)	10

Salaries are paid monthly

Table 5: Maintenance Costs

Item	Cost
Replacing faulty components	Depends on component
Routine Technical Maintenance	
Covered by Engineer	
Routine Ordinary Maintenance(clearing vegetation etc)	30

In terms of communication, we suggested a combination of uplink alternatives to cater for communication costs. We have suggested switching to different uplink options if they offer cost and reliability benefits. Table 6 gives a comparison of two uplink options. While one may be expensive, it may be the only option in places that may be hard to reach.

Table 6: Sample Transmission Costs

Uplink	per MSG ($)	Max MSG Size (Bytes)	per KB(SMS)	per KB(GPRS)
GSM! (**GSM!**)/GPRS	0.00035[6]	160	0.00224	8e-9[7]
Satellite	0.06	50	1.2288	NA

In the above table, it is shown that the cost of sending a single byte of data using a GSM text message is much less than using a satellite link and even much less using GPRS. Data on the cost of other uplinks such as optical fibre and copper was not readily available for the East African countries in question.

6 Implementation

Implementation of the East African prototype will involve software and hardware prototyping, unit testing and assembly. The following activities will take place:

[6]Using data from cheapest bundle of Airtel Uganda. 4000 SMS for 5000 UGX (approx. 1.4 USD): More at http://www.africa.airtel.com/wps/wcm/connect/africarevamp/Uganda/Home/personal/voice-and-text/extras/sms/

[7]weekly upload = 6 reports x 127 bytes/report/min * 10080 minutes/week = 7MB data (max) and yet considered the price of an 80MB package. A special bundle could be created just for the AWS for even a much cheaper price

1. Commencement of prototyping for open source Contiki firmware for RS-mote. This will include writing modules for data acquisition, RF operations (transmission and reception) and using the terminal for configuring and debugging the mote. These modules have been discussed in the design proposal section. The compiled firmware consists of the contiki operating system, which is open and already available for the ATMEGA128RFA1, and the actual application which is what shall be developed. We are already familiar with the use of the Contiki OS and have started writing modules for particular tasks as we learn further. 7 below shows estimates of the manhours required to implement all these modules.

Table 7: Software prototyping man-hour estimates

Module	**man-hours**
Read	10
Receive	20
Buffer	5
Transmit	20
Terminal	20

2. Hardware prototyping. For the most part, software and hardware prototyping are done together in embedded development because one affects the other and there is a constant need to troubleshoot. The list below shows what we hope to be one-time design activities.

   - Making the physical connections and making sure the software interacts well with the hardware.
   - Design and/or optimization of the solar radiation shield
   - PCB to host the node and the power supply.
   - Fitting nodes in their enclosures

Table 8: Hardware prototyping man-hour estimates

Module	**man-hours**
PHY connections	10
Radiation shield design	40
PCB Design	20
Enclosing nodes	5

3. Unit Testing: Testing shall be done simultaneously with prototyping. The primary goal of unit testing is to take the smallest piece of testable software in the application, isolate it from the remainder of the code, and determine whether it behaves exactly as you expect [62]. The software modules, hardware connections, RF signal and PCB tracks will all be tested as they are developed. The manhours required for this are included in those mentioned in items 1 and 2. We expect testing to take up between 20 and 40% of the total development time.

4. Assembly: In this phase, we shall put everything together to create an installable AWS. This will consist of the nodes installed on the AWS frame. We expect this part to take 10 manhours.

The timeline for this implementation is shown in the Gantt Chart in 5. We expect to finish the assembly by week 4 of February 2016.

The budget for the equipment needed for the prototyping exercise is shown below. This includes prototyping components and deployment equipment and tools.

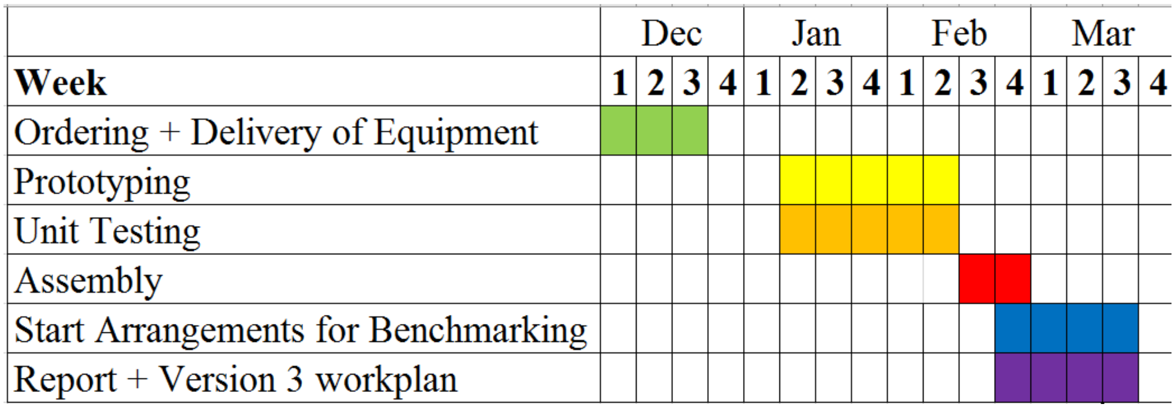

Figure 5: Timeline

7 Future Work

The cost of data transmission is a high both in terms of power and money. The problem is escalated by the frequency with which AWS send data. This data may be meteorological or AWS status data. While meteorological data should be sent frequently, AWS status data does not need this. It is important to look into the frequency at which different types of data are sent. Additionally, coding methods need to be investigated to ensure that smaller packets are sent but yet represent huge amounts of data. Care should also be taken to ensure that the coding does not waste the AWS limited resources like power and memory.

Mobile Telecommunication costs have become a competitive market. Many companies have resorted to promotional offers, which are cheaper than the regular cost of SMS messages and calls. Unfortunately, some good offers are normally available during the night hours. We would like to investigate the possibility of automatically switching uplink options to such offers for cost-saving benefits. Another aspect, which the project can pursue is collaborating with telecommunication companies for customized data packages.

Research on appropriate battery technology to use. It could be the case that other battery technologies with large capacities could power the stations very well until a scheduled human visit. Or, some clients may prefer panel-less designs in which case the choice of battery is critical. Our design is largely based on supercapacitors whose minimum voltage is rather high (2.4V) compared to the lowest voltage at which many micro-controllers can operate (1.8V). Given the load profile of the AWS and environment in which it will be installed, the best storage technology for the worst case conditions should be investigated.

Hybrid-Supply Systems: In a given AWS installation, the load profile of the gateway will vary during reception and transmission of data and during sleep or active modes. It may be the case that a good alkaline battery, which are relatively cheap, can sustain all activities other than transmission, which may need another type of storage that can unleash a huge current flow in a short time, such as an supercapacitor. Also, if the storage devices are full, there is no need to discharge them if the solar panel power is still available. An idea to look into is a hybrid-supply with automatic switch over between battery, capacitor and solar panel.

Sensor Operating voltages: Related to previous two items is a major challenge of sensor operating voltages. For example, the RS mote uses an MCP3424 ADC whose lowest operating voltage is 2.7V. This presents a major problem, since it requires that the voltage of the power source be always above this threshold. The wind vane also requires an input of 5V, which is larger than the maximum 3.6V the supercapacitors in use can provide. Future work into solutions for these issues involves around the use of hybrid power supplies, re-design of the data acquisition stage and looking into new sensors.

Radio duty cycling: The implementation of the sleep/wake up routines mentioned throughout this document is not trivial. Because the transmit time intervals are more or less constant once set, appropriate research and implementation could begin with time-synchronized duty cycling at application level in which the node and gateways are put to sleep and woken up by the RTC clock signal. Contiki itself has 3 duty cycling mechanisms which are implemented in their own layer called the RDC layer, as opposed to the MAC layer for most WSN applications

[63]. The efficiency of using these lower layers compared to the application layer could be investigated further in our particular AWS implementations. Their reliability is also a matter of research, in case we have to make a trade-off between the amount of data we can capture from transmissions within the network and the amount of power we want to save by implementing these sleep routines.

AWS Functionality: Some more functionalities could be added to the AWS, such as the ability for remote configuration, remote firmware upgrades, detection of failing sensors and many more. These would need to be weighed against their impact on the current design and their implementation is dependent on whether the needs and requirements of the stakeholders will change with time.

8 Conclusion

We evaluated the first generation prototype from several angles by looking at the power system, the gateway, the sensors and the uplinks. We compared with available literature, the needs and requirements and make a design proposal for the East African prototype. Once realized, there will be need to perform benchmarking and calibration tests to bring the accuracy of the prototype as close as possible to a standard instrument.

References

[1] B. Pehrson and R. Olsson, WIMEA-ICT Research Component 3 Progress report 2014 **(2014)**, http://dx.doi.org/10.5281/zenodo.30674.

[2] J. Reuder and J. Sansa-Otim, WIMEA-ICT:Improving Weather Information Management in East Africa for effective service provision through the application of suitable ICTs **(2013)**, https://www.uib.no/filearchive/reuder__13_11_05.pdf.

[3] WIMEA-ICT, The Bergen Prototype Plots. http://wimea-ict.gfi.uib.no/sensors/plots.html.

[4] M. Nsabagwa, M. Byamukama, J. Sansa-Otim, and R. Okou, Report on the Status of Weather Stations in Uganda **(2015)**, http://dx.doi.org/10.5281/zenodo.33156.

[5] E. Kondela, A. Nungu, and B. Pehrson, Status of Existing Weather Observation Station Network in Tanzania and the Possibility to Automate and Densify. IEEE Africon 2015.

[6] L. W. L. Nyasik and E. Ashiek, NORHED Project on Improving East Africa s Weather ICTs Preliminary Status of the Survey Findings in South Sudan. pp. 1–6.

[7] A. Avižienis, J. C. Laprie, B. Randell, and C. Landwehr, Basic concepts and taxonomy of dependable and secure computing. *Dependable and Secure Computing, IEEE Transactions on* **(2004)**, Vol. 1, pp. 11–33.

[8] Raspberry Pi Foundation, Raspberry Pi FAQ. Section I. https://www.raspberrypi.org/help/faqs/, accessed: 2015-10-15.

[9] A. Nungu, R. Olsson, J. Guo, and B. Pehrson, Towards single Watt nJoule/bit Routing. UbuntuConnect **(2014)**.

[10] N. Hema and K. Kant, Local weather interpolation using remote AWS data with error corrections using sparse WSN for automated irrigation for Indian farming. *Contemporary Computing (IC3), 2014 Seventh International Conference on* **(2014)**, pp. 478–483.

[11] P. Marino, F. Machado, S. Otero, F. Fontan, and C. Enjamio, Internet-based sensor system for environmental monitoring. *Sensors, 2004. Proceedings of IEEE* **(2004)**, pp. 40–43 vol.1.

[12] D. Miorandi, E. Uhlemann, S. Vitturi, and A. Willig, Guest editorial: special section on wireless technologies in factory and industrial automation–part I. *IEEE Transactions on Industrial Informatics* **(2007)**, Vol. 3, pp. 95–98.

[13] Sparkfun, ATmega128RFA1 Development Board. https://www.sparkfun.com/products/11197, accessed: 2015-10-20.

[14] A. Dunkels, F. Österlind, and Z. He, An adaptive communication architecture for wireless sensor networks. *Proceedings of the 5th international conference on Embedded networked sensor systems*, ACM **(2007)**, pp. 335–349.

[15] L. M. Oliveira and J. J. Rodrigues, Wireless sensor networks: a survey on environmental monitoring. *Journal of communications* **(2011)**, Vol. 6, pp. 143–151.

[16] L. Doherty, J. Simon, and T. Watteyne, Wireless sensor network challenges and solutions. *Microwave Journal* **(2012)**, Vol. 55, pp. 22–34.

[17] J. Yick, B. Mukherjee, and D. Ghosal, Wireless sensor network survey. *Computer networks* **(2008)**, Vol. 52, pp. 2292–2330.

[18] D. Kong, T. Li, X. You, X. Sun, B. Wang, and Q. Liu, The Research of Long-Distance Data Transmission Based on Meteorological Sensor Network. *International Journal of Future Generation Communication and Networking* **(2014)**, Vol. 7, pp. 59–70.

[19] R. C. Carrano, D. Passos, L. Magalhaes, and C. V. Albuquerque, Survey and taxonomy of duty cycling mechanisms in wireless sensor networks. *Communications Surveys & Tutorials, IEEE* **(2014)**, Vol. 16, pp. 181–194.

[20] L. D. Mendes and J. J. Rodrigues, A survey on cross-layer solutions for wireless sensor networks. *Journal of Network and Computer Applications* **(2011)**, Vol. 34, pp. 523–534.

[21] H. S. Bagiorgas, M. N. Assimakopoulos, A. Patentalaki, N. Konofaos, D. P. Matthopoulos, and G. Mihalakakou, The design, installation and operation of a fully computerized, Automatic Weather Station for high quality Meteorological measurements. *Fresenius Environmental Bulletin* **(2007)**, Vol. 16, pp. 948–962.

[22] D. Antolín, A. Bayo, N. Medrano, B. Calvo, S. Celma, and C. Azcona, Analysis and implementation of a wireless sensor network with remote access through SMS. *Circuits and Systems (LASCAS), 2012 IEEE Third Latin American Symposium on*, IEEE **(2012)**, pp. 1–4.

[23] R. L. Wilby, Climate for development in Africa (ClimDev)–climate sciences and services for Africa. Strategic research opportunities for ClimDev-Africa. **(2014)**.

[24] T. Le Dinh, W. Hu, P. Sikka, P. Corke, L. Overs, and S. Brosnan, Design and deployment of a remote robust sensor network: Experiences from an outdoor water quality monitoring network. *Local Computer Networks, 2007. LCN 2007. 32nd IEEE Conference on*, IEEE **(2007)**, pp. 799–806.

[25] A. Sayin, VHF/UHF Uplink Solutions for Remote Wireless Sensor Networks **(2013)**, http://www.diva-portal.org/smash/get/diva2:626842/FULLTEXT02.

[26] G. Werner-Allen, K. Lorincz, M. Ruiz, O. Marcillo, J. Johnson, J. Lees, and M. Welsh, Deploying a wireless sensor network on an active volcano. *Internet Computing, IEEE* **(2006)**, Vol. 10, pp. 18–25.

[27] G. Werner-Allen, J. Johnson, M. Ruiz, J. Lees, and M. Welsh, Monitoring volcanic eruptions with a wireless sensor network. *Wireless Sensor Networks, 2005. Proceeedings of the Second European Workshop on*, IEEE **(2005)**, pp. 108–120.

[28] T. J. Salo, Narrowband ip over amateur radio networks (niparnets) next-generation networking for amateur radio **(2013)**, accessed: 2015-10-27. https://www.tapr.org/pdf/DCC2013-NIPARnets-AB0DO.pdf.

[29] P. Heil, S. Gerland, and M. Granskog, An Antarctic monitoring initiative for fast ice and comparison with the Arctic. *The Cryosphere Discussions* **(2011)**, Vol. 5, pp. 2437–2463.

[30] M. Toledano-Ayala, G. Herrera-Ruiz, G. M. Soto-Zarazúa, E. A. Rivas-Araiza, R. D. B. Trujillo, and R. E. Porrás-Trejo, Long-range wireless mesh network for weather monitoring in unfriendly geographic conditions. *Sensors* **(2011)**, Vol. 11, pp. 7141–7161.

[31] Australian Bureau of Meteorology, Automatic Weather Stations for Agricultural and Other Applications. http://www.bom.gov.au/inside/services_policy/pub_ag/aws/aws.shtml, accessed: 2015-10-27.

[32] F. Simjee, *Everlast: Long-life Supercapacitor-operated Wireless Sensor Node*, University of California, Irvine **(2006)**, https://books.google.no/books?id=RyP_NwAACAAJ.

[33] R. Lajara, J. Alberola, J. Pelegr, T. Sogorb, and J. Llario, Ultra Low Power Wireless Weather Station. *International Conference on Sensor Technologies and Applications* **(2007)**, pp. 469–474.

[34] Robert Olsson, The Universal IEEE 802.15.4 Sensor for WSN and IoT applications. http://radio-sensors.com.

[35] IEEE Standards Association, Part 15.4 Low-Rate Wireless Personal Area Networks (LR-WPANs) **(2011)**.

[36] NICTBB, http://www.nictbb.co.tz/map.php, accessed: 2015-10-23.

[37] John Nasasira, Global South-South Development Expo Mini Partnership Forum on Enhancing ICT Development and Connectivity for LLDCs. http://unohrlls.org/custom-content/uploads/2014/04/Presentation-Hon.John-Nasasira-Minister-of-Information-and-Communications-Technology-Nairobi.pdf **(2013)**, accessed: 2015-10-23.

[38] Google Inc., Google Project Link. https://www.google.com/get/projectlink, accessed: 2015-10-20.

[39] Realtek Semiconductor Corp., http://www.realtek.com.tw/products/productsView.aspx?Langid=1&PFid=15&Level=4&Conn=3&ProdID=22, accessed: 2015-10-25.

[40] http://propox.com, Poland, http://pl110144744.fm.alibaba.com/product/117634927-0/MMnet01.html, accessed: 2015-10-25.

[41] Uganda Communications Commission, http://www.ucc.co.ug/data/qmenu/3/Facts-and-Figures.html, accessed: 2015-10-25.

[42] Tanzania Communications Commission, March 2015 Quartely Report. http://www.tcra.go.tz/images/documents/telecommunication/telcomStatsMarch15.pdf **(2015)**, accessed: 2015-10-25.

[43] http://www.simcom.eu/index.php?m=termekek&prime=1&sub=40&id=0000000147, accessed: 2015-10-20.

[44] http://www.simcom.eu/index.php?m=termekek&prime=1&sub=40&id=0000000246, accessed: 2015-10-20.

[45] SARA G3 SERIES. https://www.u-blox.com/en/product/sara-g3-series, accessed: 2015-10-20.

[46] ”lisa u2 series.

[47] INTEL Corporation, Intel XMM 6255 Modem with Intel SMARTi UE2p Transceiver. http://www.intel.com/content/dam/www/public/us/en/documents/white-papers/xmm-6255-smarti-ue2p-paper.pdf, accessed: 2015-10-20.

[48] SIM800L Quad-band Network Mini GPRS GSM Breakout Module. www.dx.com/p/sim800l-quad-band-network-mini-gprs-gsm-breakout-module-382445.

[49] IEEE Standards Association, https://standards.ieee.org/findstds/standard/802.11af-2013.html, accessed: 2015-10-25.

[50] C Cordeiro, H Gossain, R Ashok, DP Agrawal, in *Brazilian Symposium on Computer Networks*, UFRN/DIMAp **(2005)**, Chapter The Last Mile: Wireless Technologies for Broadband and Home Networks,.

[51] SeedStudio, http://www.seeedstudio.com/depot/2KM-Long-Range-RF-link-kits-w-encoder-and-decoder-p-321.html, accessed: 2015-10-20.

[52] Digi International, http://www.digi.com/products/xbee-rf-solutions/modules/xbee-pro-900hp, accessed: 2015-10-20.

[53] Iridium, https://www.iridium.com/products/types/Transceivers, accessed: 2015-10-20.

[54] Nupoint Systems, http://www.nupointsystems.com/satellite-transceiver/, accessed: 2015-10-21.

[55] https://www.satellitephonestore.com/iridium-short-burst-data-service-plans, accessed: 2015-10-21.

[56] http://www.discoverytelecom.eu/catalog/2321.html, accessed: 2015-10-21.

[57] http://www.utl.co.ug/internet/dedicated-services/dedicated-internet-service-over-vsat/, accessed: 2015-10-23.

[58] B. Pehrson and R. Olsson, Powering Devices using Ultra-capacitor batteries. IEEE Africon **(2015)**.

[59] http://wimea-ict.gfi.uib.no/sensors.dat.

[60] https://www.raspberrypi.org/forums/viewtopic.php?f=29&t=25777, accessed: 2015-10-23.

[61] ATMEL Corporation, ATmega128RFA1 Datasheet. http://www.atmel.com/Images/Atmel-8266-MCU_Wireless-ATmega128RFA1_Datasheet.pdf, accessed: 2015-11-08.

[62] Microsoft Corporation, Unit Testing. https://msdn.microsoft.com/en-us/library/aa292197%28v=vs.71%29.aspx, accessed: 2015-10-25.

[63] Radio Duty Cycling. https://github.com/contiki-os/contiki/wiki/Radio-duty-cycling, accessed: 2015-11-12.