International Journal of Engineering & Technology, 7 (4) (2018) 5806-5814

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET
doi: 10.14419/ijet.v7i4.28634
Research paper

Energy monitoring and verification control interface for split unit air conditioners

Estienne Cronjé [1], Immanuel N. Jiya [1] *, Rupert Gouws [1]

[1] School of Electrical, Electronic and Computer Engineering North-West University, Potchefstroom 2520, South Africa
**Corresponding author E-mail: immanueljiya@ieee.org*

Abstract

The aim of this paper is to present a new energy monitoring and verification control interface for the Eskom demand market participation (DMP) program that can be implemented on a commercial scale on all the split unit air conditioner systems of consumers actively participating in the program. After the detailed design was completed, the sub-system circuits were created and tested, and evaluation commenced. The sub-systems were integrated after each test and evaluation phase to see whether the system functionality performed as it should have. The final integration was implemented and the final system (consisting of a master – and slave controller unit) were placed inside their enclosures. From there on, the programming logic was modified to and the program was evaluated to see if the completely integrated system was still functional. The slave controller unit was attached to the split unit air conditioner together with the appropriate sensors to monitor and verify the operation of the air conditioner. The developed system was able to successfully monitor the current state of the split unit air conditioner, control it to a specific mode (usually fan mode) according to the required demand at that stage, verify that it has indeed been switched to that specific mode and provide an estimate of the power consumption in the current mode.

Keywords: *Air Conditioners; Control; Demand Market Participation; Energy Monitoring & Management.*

1. Introduction

Eskom, South Africa's primary electricity supplier needs to ensure the security of electrical supply to the South African society and ensure economic growth doesn't falter due to a lack of this supply. It is of utmost importance that the electrical demand doesn't exceed the supply. But this is difficult since the demand fluctuates due to peak stages (periods where the demand is higher than usual) as well as an ever-increasing number of citizens demanding electricity [1]. If instability in the power grid occurs, then the entire electrical network can fail which can have disastrous consequences on the country. To prevent this from happening, Eskom implements load shedding or alternatively, load reduction, as an effort to manage South Africa's power grid stability.

Another method used to relieve network constraints is to implement an ingenuity called customer participation. Customer participation, also known as load curtailment, is a method where Eskom agrees with large industrial customers to reduce their consumption in times of grid strain. This is achieved through time-based rates in exchange for electricity of financial enticements. These participants have been able to aid in a 20% load reduction in times of need. This method, however, takes a minimum of two hours to implement [1], [2].

Although load shedding hasn't affected South Africans in 2017 thus far, there are still no guarantees that load shedding wouldn't be implemented in the next three to four years [3, 4]. Thus, from these methods, it is evident that load reduction innovations, that can implement demand response initiatives is in demand for development and can be beneficial for the electrical grid and all its consumers.

The complex engineering problem that exists is that the power demand in in some parts of the world cannot continuously be met. An alternative to cope with the increasing demand is to implement load shedding or load curtailment to reduce the demand [1]. Currently South Africa is not experiencing any load shedding. But a more efficient solution must be put into place to act as a preventative measure to stop the implementation of load shedding in the future. HVAC (heating, cooling, and ventilation) is the largest area in the commercial sector where load reduction can be implemented in such extent that immense improvements can be made.

Demand market participation is ingenuity where consumers actively participate in the operation of the national grid by reducing their electrical usage during peak times. By using this initiative for HVAC systems in the commercial sector can create a noticeable load reduction on the national grid and has the potential to decrease, or remove, the need of implementing load shedding in the future. A problem Eskom faces is monitoring and verifying whether the participating companies have indeed initialized their HVAC load reduction protocol. Hence a verification and monitoring interface, specifically for split unit air conditioners, must be designed to monitor the status of the air conditioners and verify whether the participating companies complied to the change in load reduction status. [1], [5].

It is of great importance to do research on the current available solutions. This is done to gain a better understanding as to why the existing solutions were successful or on the other hand, ineffective. The reason for their inability to solve this problem must be identified. This process creates an opportunity to learn from previous design flaws and how to implement it correctly to this project. Home automation and universal remote controllers are well-known concepts that already exist in the market. They range from wireless controllers with an integrated smart-mobile interface to a more hands-on approach in the form of a wall mounted controller device. The focus of this report will be more towards air conditioner controllers rather than a whole home automation system.

Copyright © 2018 Estienne Cronjé et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

These devices, however, do not utilize control over air conditioners based on demand response such as lower their energy consumption during peak times.

The AUKEY Wi-Fi Switch, Air Conditioner Smart Plug is a smart Wi-Fi air conditioner remote manager and monitors with an in-app, real-time temperature display and energy consumption report [6]. It enables the user to activate or shut off the air conditioner from anywhere by using your smartphone. It enables the user to be more energy conscious by monitoring energy use in the mobile application. The mobile app gives the user a selection of settings for operation, temperature control, schedules, and more.

AirPatol Wi-Fi–smart AC control device also allows the user to access an air conditioner by using a mobile app. The AirPatrol uses wireless channels for communication between the smartphone and the device. The device then communicated with the air conditioner via infrared [7]. The user can adjust the room temperature prior to arrival by using the mobile app, or by setting up time cues. The mobile app lets the user know when the air conditioner temperature or humidity drops too low or rises too high. This device does not implement demand response initiatives.

The Tado cooling intelligent AC control device monitors the user's location and switches off the air conditioner when the last person has left the house. This device also precools the house when any person is on their way home. Tado's indoor positioning system which is based on Bluetooth, low energy, iBeacon® technology allows the user to move around the house without worrying about turning on or turning off the air conditioners. Tado's room to room feature monitors the rooms and cools them as you go [8]. The device does not consider the time of day and it would be most suitable in areas which are mainly cold.

The Ambi climate smart AC control is a smart hub that connects to any remote-controlled air conditioner via Wi-Fi by tapping into the infrared signal and from thereon communicating to the air conditioner. From thereon, the user can control their air conditioned by using the mobile app. The Ambi Climate learns and remembers your temperature preferences by using your feedback and by using its own built-in temperature, humidity, motion and sunlight sensors [9]. The device helps with energy efficiency but it does not have peak time load reduction features.

2. Proposed solution

A basic preliminary conceptual design to the engineering problem is illustrated in figure 1. This shows a visual of the complete control system and how the suggested solution can be implemented. The system has only one main controlling unit and as many office control units as needed to control several split-unit air conditioners. For simplicity sake, the conceptual design (illustrated in figure 1) only has one office controller unit and an air conditioner that needs to be controlled. The office controller unit forms the basis for the successful controlling of an air conditioner. Its basic design is shown in figure 2.

The microcontroller, mounted to the air conditioning unit, will communicate with the air conditioner by using infrared (IR) communication. This form of communication is chosen because it is a default communication protocol that most of the spilt-unit air conditioners use. The microcontroller unit will include a sound analysis sensor (which must be designed) alongside a temperature – and a velocity sensor to form the verification interface that is needed. This unit will function as a wireless device; thus, its power source must be wireless (batteries, etc.).

The main controller unit (MCU), located in the office building, will control all the room units attached to the air conditioning units. This controller can be accessed by the business owner to input the level of demand needed – as instructed by Eskom – and implement this instruction on all the office control units. The communication platform between the main controller and the office controller will be based on RF (radio-frequency) communication since it is a cheap and reliable source of communicating over a long distance.

The verification signal (via temperature and velocity sensor or sound analysis sensor) will be sent from the office controller unit to the main controller which will interpret the signal to verify if the instructed signal has been correctly implemented. Another feature that the main controller will have is its ability to keep and log the time of use. This is done to save wasteful energy consumption due to forgetfulness of personnel. This is implemented by switching off all the split units in the office building after business hours. The main controller will be a plug-in source which can be plugged into a wall socket. This ensures that the controller will operate efficiently without being interrupted by maintenance such as battery exchanges.

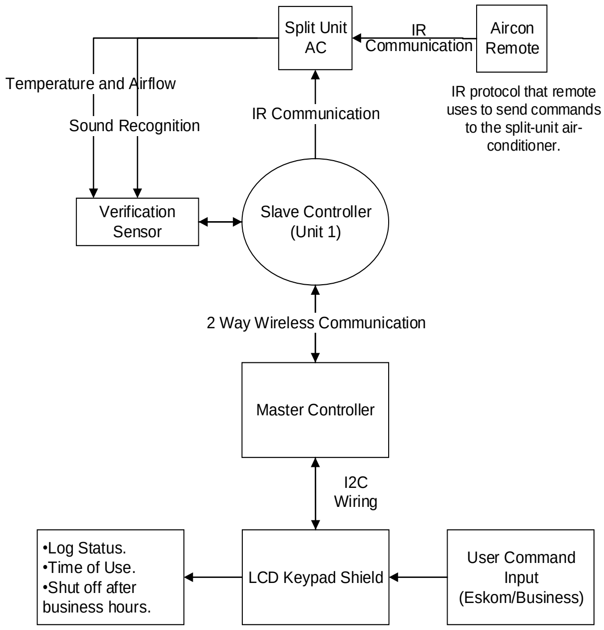

Fig. Error! No text of specified style in document.**:** Basic Conceptual Design of Split-Unit Air Conditioner System.

The bubble, containing the asterisk (*) in figure 2, describes some of the secondary (optional) objectives i.e. the addition of occupancy sensors and the addition of a location tracker which follows the employee's whereabouts relative to the air conditioner. If the employee leaves his/her office the state of the air conditioner changes with respect to their position relative to the office microcontroller unit e.g. if the employee's position is calculated as an "x" amount from the air conditioner unit (in meters), the status of the air conditioner may change from being on to off or from being in air-conditioning mode to fan mode and vice versa.

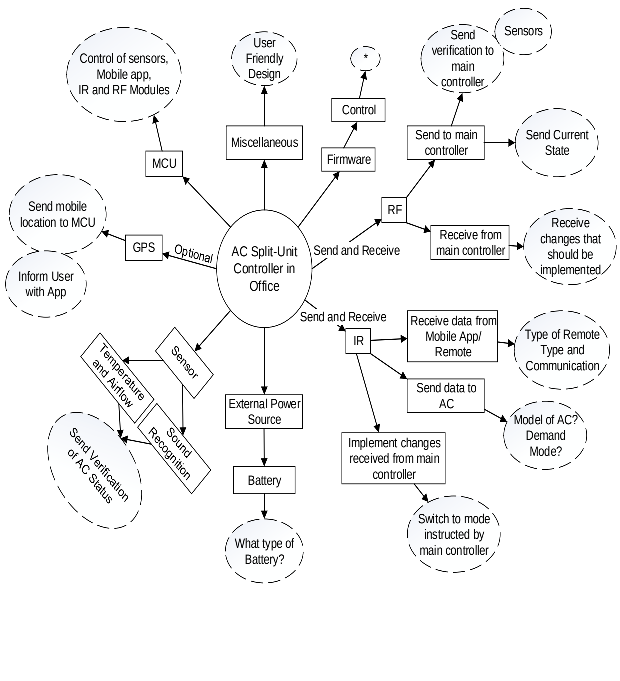

Fig. 2: Basic Preliminary Conceptual Design of Office Micro Controller.

3. Design overview

Figure 3 shows that the main components of the system has many sub sections and that there are numerous component types to choose from. It is therefore needed to set apart the components that are best suited for this research.
The detail design shows how these specific components will be implemented to fulfil their function as a subsection which will evidently form part of the final design.

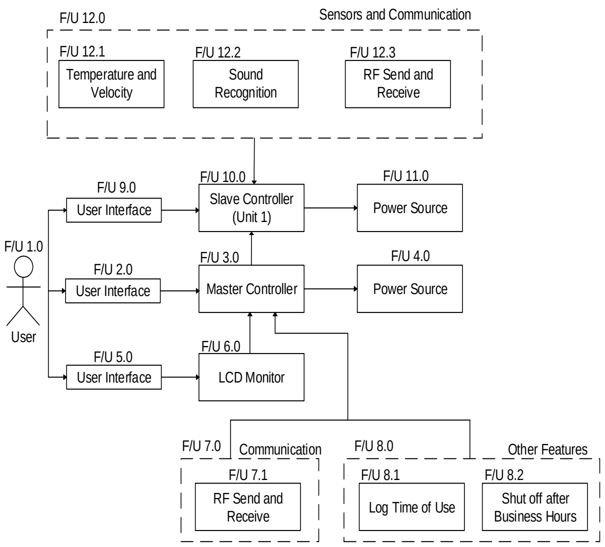

Fig. Error! No text of specified style in document.**:** Basic System Architecture.

3.1. Medium range radio frequency (RF) communication

RF Communication was chosen as the top ranked choice. It will be used to send and receive data from the main (master) controller unit to the secondary (slave) controller unit and vice-versa. Refer to figure 2 to see how the communication link between the two controllers is setup. The following data will be sent from the slave controller to the master controller:

- Which office controller is accessed;
- Sound sensor data to establish whether verification "beep" has been triggered;
- Temperature and airflow sensor data to confirm correct mode has been implemented; and
- The current operational state the air-conditioner is in.

The master controller will use this data to verify whether the correct split-unit air conditioner is in the correct operation mode. Furthermore, the operation status (sent from the slave controller) will be used to determine the daily energy time-of use (TOU) data for data logging. The master controller will send the following commands to the slave controller:

- The operation instruction that must be implemented;
- To reinstate a command if the verification signal failed to change the air conditioner's status.
- Send an after business hours command to shut off the split-unit air-conditioner.

The instruction that the slave controller must implement (which is then enforced on the split-unit air-conditioner) is processed using hexadecimal numbers which are eight characters long. This leads to the following design choice: The master controller will send a nine-bit character to the slave controller where the first bit is used to identify the slave controller and the remaining eight bits will carry the information signal (or command) that needs to be passed on to the air-conditioner. This command and verification signals are transmitted and received by using RF wireless communication.

The RF module which has a frequency range of 418 MHz to 455 MHz offers more than 100, license free, channels. So, when calibrating the module, a clean channel will be picked to prevent other nearby RF signals from interfering with the module so accurate data can be sent and received between the master controller and the slave controller units. Thus, a certain frequency will be calibrated which will only be able to interpret data received and sent on that specific frequency.
The RF module chosen for the job is an APC220 radio communication module. It has a range up to 1000 meters with a default UART rate of 9600 bps (line-of sight) and uses Gaussian frequency shift keying (GFSK) modulation. The master controller unit will transmit a 8 character command signal (via RF communication) to the slave controller unit which will interpret the 8 character command and implement it on the split-unit air-conditioning unit.
The communication module is therefore already designed. The student has to set up the module and the programming thereof must be designed and implemented.

3.2. Power source

The power source that was chosen by the trade-off study is a Lithium Ion power source. This power source will power the slave micro controller unit (office controller) which will carry out commands, given by the master micro controller unit, to the split-unit air-conditioner.
The slave unit will also host the sound, temperature and airflow sensor so it is vital that the verification sensors have enough voltage to function properly. The 9 V battery was chosen because it falls into the specified voltage rating that an Arduino micro controller must have to function properly. This battery is chosen because it can last up to 11 times longer than a normal battery and it weighs 33% less than a standard 9V alkaline battery which is ideal. It also has a 10-year storage life so a company can buy them in bulk and not worry about deteriorating battery life.

3.3. Verification sensor – temperature and airflow sensor

The reason why these sensors were chosen as the verification feedback loop is simply because all air-conditioner units have two common variables that are present during operation (even at different states) which are airflow and temperature. When an air-conditioner unit is switched on, there is an immediate sign of airflow across the blades which intensify as the air-conditioning mode is changed from fan mode to AC mode. Another noticeable factor is that when airflow occurs, there is a temperature drop or a rise in temperature (depending whether the air-conditioner is in heating mode or cooling mode).
The two sensors chosen for the verification operation is the DS18B20 temperature sensor and a Rev C wind sensor. The temperature sensor converts temperature analogue data into 9 to 12-bit digital data in 750ms. The temperature accuracy is ± 0.5 °C and ranges from -10 °C to $+85$ °C. This is a three-wired sensor where 2 wires are used for voltage supply (3-5.5 V) and ground respectively and 1 wire is a digital data pin for communication.
The Rev C is a low-cost anemometer (wind sensor) used to measure the airflow of the air-conditioner unit. It uses a technique known as the "hot-wire" technique which involves the heating of an element to a constant temperature and measuring the electrical power that is required to maintain that heated constant, to determine the wind speed. The measured electrical input is proportional to the square of the wind speed. The supply voltage ranges from 4-10 V and the supply current from 20-40 mA (depending on the wind speed). But to start up takes the element about 10 seconds to start measuring, thus, another method of verification is needed.
Hence, sound recognition will be used alongside the above-mentioned sensors to verify whether the split-unit air-conditioner is in fact in the correct operation mode. This is done by using a sound sensor which consists of a microphone and an audio power

amplifier IC. Its operating voltage is 3.3 - 5.3 V. The sound sensor is used to pick up the "beeping" noise the air-conditioner unit makes when it is switched on or when it changes state. Each state has its own "beeping" sound and thus its own frequency profile which can be used to verify whether the correct state is employed.

The sound sensor is active for a time interval, just after the instruction has been sent from the slave controller to the air-conditioner, to "listen" for a "beep" sound that the split-unit air-conditioner makes when it is tuned on or off. The "beeping" sound is recorded and the FFT is calculated over the recorded audio clip and the magnitude spectrum is measured around the beep central frequency which is around 4.1 kHz.

This is done by using digital sound processing (DSP). The code sweeps through the spectrum to loop for frequency power peaks that correlates to a "beep" noise. When it finds it, it returns a "TRUE" if the signal energy exceeds a "beep-threshold". When a "beep" is detected, the slave microcontroller unit stops listening and the verification signal are sent to the master controller and the status of the air-conditioner is confirmed on a LCD screen.

3.4. Controller arduino

The reason why this controller is chosen is because it scored the highest rating in the trade-off study. Another tipping point is the fact that the Arduino is an open source platform where a lot of information and tips regarding various projects are free to be explored. Lastly, it is low cost and most sensors and components are compatible with Arduino micro controllers, thus due to simplicity and user friendliness that this type of controller provides, it achieved the highest rating.

The micro controller, chosen, must be energy efficient and its processing power must be high enough to be able to send, receive and process the data (coming from the sensors, air-conditioning unit, master – and slave controller unit) effectively to meet the specifications of the project. The program logic followed by the master controller unit is shown in figure 4 followed by the slave unit logic diagram in figure 5.

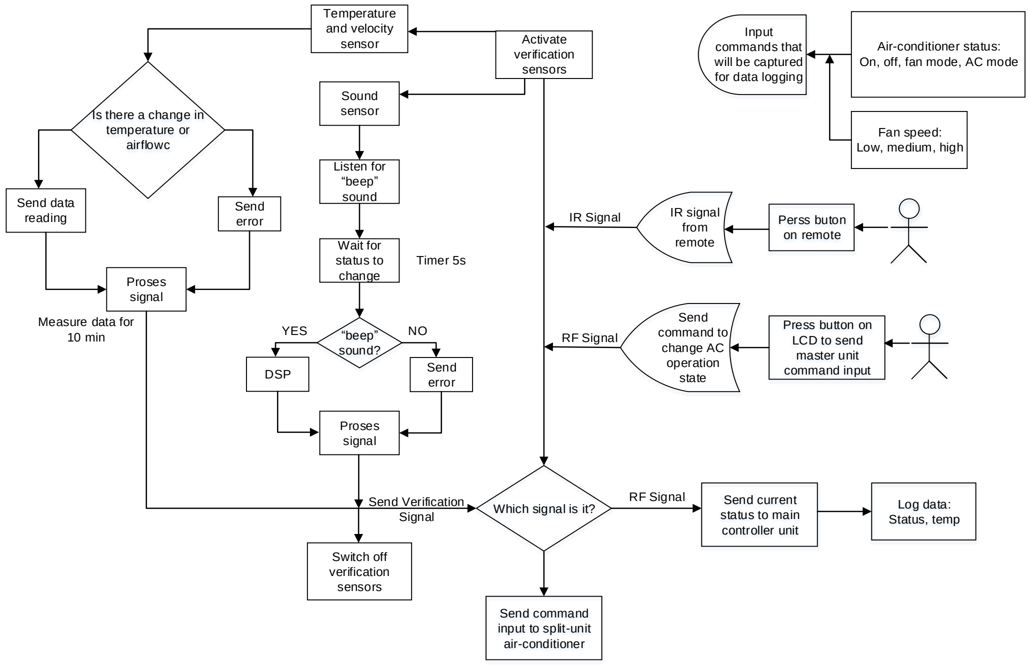

Fig. 1: Slave Controller Unit Program Logic.

The program to that needs to be implemented on both controllers must be successfully designed and implemented to satisfy the outcomes for this project. The program logic diagram indicated that only certain commands will be logged. This is because there are various split-unit air-conditioners on the market where each of them have their own unique functions that differ from the rest. There are only few commands that these split-unit air-conditioners have in common, thus, only these input commands will be logged. These operations are:

- Whether the air-conditioner is in ON or OFF state
- Whether it is in air-conditioning or fan mode
- Whether the fan speed is low, medium or high.

The verification sensors verify whether these states are TRUE i.e. whether the input commands were successfully implemented and that the correct modes were selected. To make the slave controller (located in the office close to the split-unit air-conditioner) more energy efficient the sensors are only active for a certain amount of time.

For instance, when the input command is given to the sensors become active. The first sensor that becomes active is the sound sensor which listens for a time interval of 5 second in which it detects whether a "beeping" sound has been detected by using digital sound processing to measure the magnitude peaks (by using the Fourier Transform) to verify whether the sound is greater than the threshold magnitude around the 4.1 kHz frequency.

The temperature and airflow sensors are much simpler. The (DS18B20) temperature sensor becomes active and stays active for 10 minutes where it reads temperature analogue data and through this reading, determines in which state it is in. The airflow sensor is also active for this amount of time where it reads the airflow intensity to determine whether the air-conditioner is in a low, medium or high fan (or AC) speed state.

The program logic flow diagram also indicates that there are two procedures that exist, one being where the master controller sends an input signal to the slave unit to change the state of the air-conditioner and secondly, where the input command is changed by the user itself (using the air-conditioner remote). In this case the sensors must become active again to monitor and verify the state of operation of the split-unit air-conditioner.

The programming logic diagram in figure 4 shows how the LCD is implemented where the user can push a button to send an input command from the master unit to the slave unit. The LCD monitor will also visually display the time of day, the state of operation the air-conditioner is in, the estimate power consumption of the split-unit air-conditioner and the logged data.

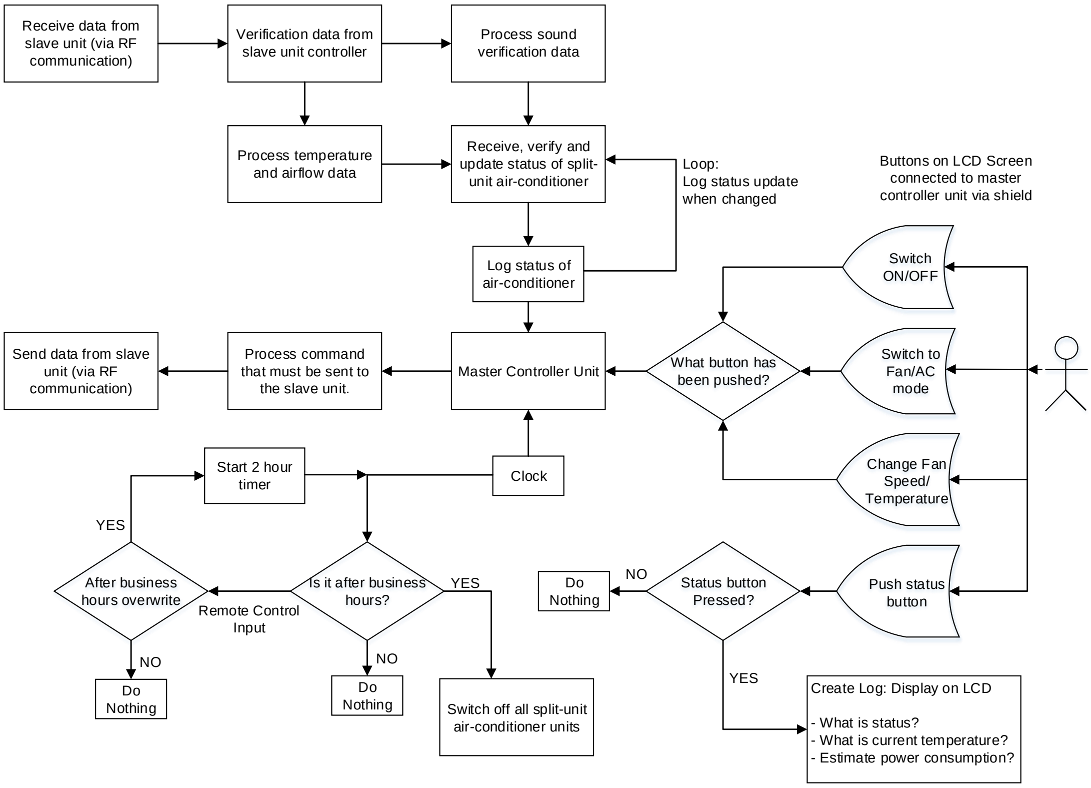

Fig. 2: Master Controller Unit Program Logic.

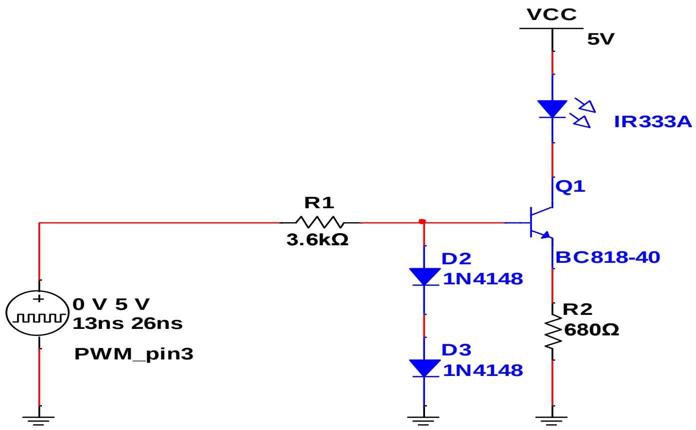

Fig. 3: IR Transmitter Circuit.

The master unit will receive user command updates from the slave unit (e.g. when the air-conditioning unit remote is used instead). This information will then be logged by the master controller unit. At the start of the day, a timer starts at 8 am which enables the air-conditioner to be accessed by the micro controllers. When the business closes (usually at 5 pm) then the master controller triggers an after-business hours protocol which sends a "switch off all split-unit air-conditioners" command. This is done to ensure that no air-conditioner has been left on by the employees.

When the employee is working later than usual, he can overwrite this protocol, which starts another timer for 2 hours. After this time has elapsed, the master controller unit will send a switch-off signal to the slave unit and this will then be implemented on the split-unit air-conditioner.

A LCD Keypad Shield interfaces with the master controller unit. It has six usable push buttons which all have specific functions as seen in the previous figure. When the user presses one of the split-unit air-conditioner operation buttons, the program code determines what button has been pushed and sends the command (via RF communication) to the slave controller. The slave controller processes and interprets the command and implements it on the split-unit air-conditioner.

3.5. IR transmitter

The IR sender circuit is designed using the approach in [10] to build an IR sender and receiver circuit. To transmit an infrared signal, a TSHG5410 IR LED is used. Its colour is centred around a peak wavelength (λ) of 850 nm. This circuit will be used to transmit the commands sent, through the slave micro-controller, by the master micro-controller to control the split-unit air-conditioner.

A pulse width modulated (PWM) signal of 38 kHz will be used to match the emitter to the IR receiver circuit. To ensure accurate IR transmission, the slave controller will transmit an IR burst of 2, evenly spaced IR LEDs, placed at an angle to ensure that a perfect line of sight connection can be established. This gives a specification which leads to an indication to what shape the device (hosting the slave unit) should be. Figure 6 shows the recommended circuit configuration that will be used.

Copyright © 2018 Estienne Cronjé et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

3.6. IR receiver

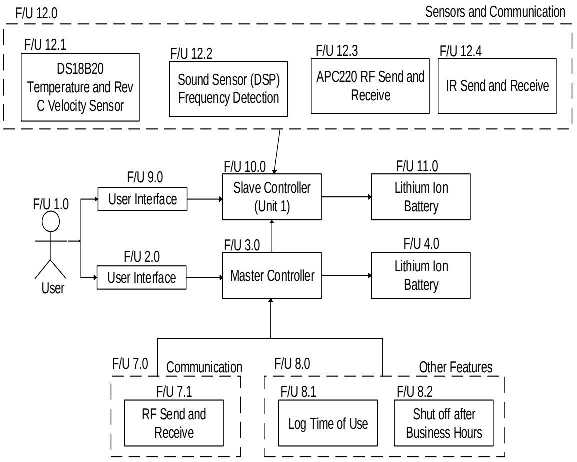

Fig. 4: System Architecture with Chosen Components.

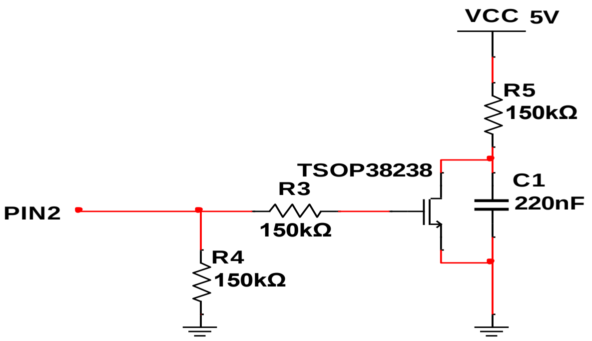

Fig. 5: IR Receiver Circuit.

The IR receiver circuit is also designed by following the approach in [10]. The component chosen is a TSOP312 IR Receiver. This infrared receiver is powered with a 5 V supply voltage. This receiver will be capable of only receiving transmitted IR signals at 38 kHz. The circuit schematic is shown in figure 7 below.

3.7. IR communication type

There are various existing IR protocols used in IR communication devices. Since there are no direct database that can be accessed to collect these protocols for various split-unit air-conditioner models, these IR codes must manually be captured and implemented in the system. Thus, several split-unit air-conditioners must be made available to capture all the needed information for this project. Thus, the IR protocols must still be determined.
After the protocols have been identified, a database will be established to store the needed commands that will be used to conduct the experiments for the projects. The basic programming will be able to decode the information that is obtained from the receiver.

3.8. Main unit input

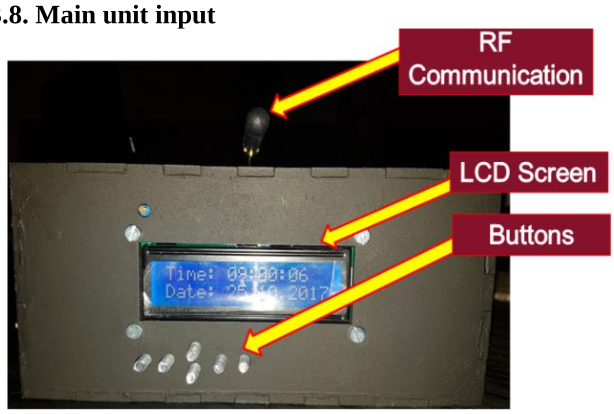

Fig. 6: Master Controller Interface.

The main unit must have some method of input to capture the user input. This is done by designing a graphical user interface (GUI) on the LCD Keypad Shield, where 5 main buttons will be used to navigate and send user commands to the master micro-controller unit. The GUI setup is currently shown in figure 8. At this stage, the GUI can still change when implementation begins.

3.9. Complete system overview

Figure 9 shows how all the selected components interact with each other. This is shown as a final system architecture where the chosen components are added to the system.

4. Implementation and results

The implementation and evaluation process consist of the construction of each sub-section circuit to test the particular sub-section circuit's functionality by writing a small program to test whether the programming logic of the component is working as it is should. From there on each sub-section is integrated to from a larger sub-section circuit with more complex functionality and programming logic. The system functionality is evaluated and tested once more to see whether everything is operating as it should and to see whether each sub-section performs its role in the integrated circuit.
This process of implementation and evaluation continues as each sub-section is added to form part of a larger sub-section until all the sub-sections have been integrated.
The implementation started off with the development of the IR transmitter and receiver circuit which were realised using a breadboard at first. All sub-section components were first built on a breadboard to test their functionality. The program logic was tested by using smaller programs to see whether the component functioned in a desirable way. The split unit air conditioner used to develop and test this project on was a Whirlpool AMD 322.

4.1. IR transmitter and receiver

The raw IR data was received by the IR receiver and stored in an array called "raw". The array consists of raw integer data and has a capacity of 100. The resulting data is then printed on the Serial monitor. This data can then be used in another Arduino program to transmit the resulting IR data via an IR LED and implement it on the split unit air conditioner. The raw captured IR data which is printed on the serial monitor is as the following:

unsigned int raw[100] = {9050, 4550, 650, 1600, 650, 1600, 650, 500, 650, 500, 650, 500, 650, 500, 650, 500, 650, 1600, 650, 500, 650, 1600, 650, 1650, 650, 500, 650, 500, 600, 500, 650, 500, 650, 500, 650, 500, 650, 500, 650, 1600, 650, 500, 650, 500, 650, 500, 650, 500, 650, 500, 650, 500, 650, 1600, 650, 500, 650, 500, 650, 500, 650, 500, 600, 500, 650, 1650, 650, 500, 650, 500, 600, 500, 650, 500, 650, 500, 650, 500, 650, 500, 650, 500, 650, 500, 650, 500, 600, 550, 550, 600, 600, 500, 600, 550, 600, 550, 600, 550, 600};

This specific data stored in the array represents the ON/OFF air conditioner command as decoded from the air conditioner remote itself. By following this exact method, the IR commands for the split unit air conditioner "mode" function was decoded and stored. This was the first step taken to acquire remote replacement.
The next part needed to realise remote replacement is to include an IR LED to the design. The raw IR data which was recorded previously must now be stored in the memory of the Arduino. This can be done in two ways, where the first way is to store the IR data directly in the dynamic memory of the Arduino. The drawback of doing this is that there are only 8 Kbytes of SRAM (dynamic memory) available.
Thus, the second option was chosen, which was to store the IR data in the flash memory of the Arduino Mega. The Mega has 256

Kbytes of flash memory available so for this reason, keeping program memory in mind, it was a good choice. The IR raw data was stored in flash memory by using the PROGMEM function. To retrieve the IR data from the flash memory, the memcpy_P() function was used.
When the IR command is needed, the memcpy_P() function retrieves the IR data from the flash memory and stores the data in a new variable with the exact same size as the array where the IR data were stored in. This sets up the buffer to wait until the entire IR code (data) is copied before it is ready to be transmitted.
The IR LED transmits the IR code to the air conditioner at a rate of 38 kHz. The IR receiver inside the split unit air conditioner receives the IR data, interprets it, and changes the air conditioner's state in accordance to the command it received.

4.2. Temperature sensor

The initial temperature sensor implementation was done using a breadboard, 3 LEDs, 3 x 100 Ω resistors and a LCD screen.
The Arduino code works on the principle where it receives temperature data from the DS18B20 temperature probe by using the Dallas Temperature library and OneWire library. What the code does is it receives the sensor data from pin 10, which was defined by the OneWire library. The data is stored in the variable "sensors" by using the instruction: DallasTemperature sensors (&oneWire). This code set constraints on the temperature readings and turned the LEDs on in accordance to the temperature constraints that were set.

4.3. The sound sensor

The implementation of the sound sensor was done by connecting the sound sensor analogue output pin to the analogue 0 pin. The digital sound processing algorithm that is used is frequency detection. In order to realise the code, an analogue to digital converter must be set up. In order to get the most accurate result, a large sampling frequency is needed.
The sampling rate for an audible human voice and music is 44.1 kHz. This is because the sampling rate should be at least twice that of the fundamental frequency. The highest sampling frequency that the Arduino can reach by using the AnalogRead() function is 10 kHz. Thus, an alternative approach must be taken, hence, direct port manipulation [11].
Some code are written in the Setup() which tells the Arduino to permanently measure the A0 input pin. This ensures that the Arduino stays focussed on the analogue pin regardless of the other functions that are currently running in the loop. Thus, the Arduino constantly updates a variable "ADCH" with new incoming A0 values at a sampling rate of 38.5 kHz ($\frac{1}{f_s} = 25.97\mu s$).
This was done by manually setting the Arduino's internal analogue to digital converter (ADC) register to 500 kHz and reading an 8-bit value from analogue input A0 from the ADC itself. The ADC normally takes 13 clock cycles to read a new value which resulted in a sampling frequency of $f_s = \frac{500}{13} \approx$ 38.5 kHz.
The next step is to implement a midpoint detection algorithm which keeps track of the times that the audio wave crosses 2.5 V. Considering that the audio wave is sinusoidal and has a peak of 5 V and a minimum of 0 V, then at 2.5 V the slope of the wave should be 0. Thus, A0 is measured with 8 bits precision, where 0 V equals a value of 0 and 5 V equals a value of 255, then 2.5 V should be 127.
Next a timer must be put into place to measure the time between the midpoints (where the slope equals 0) of the sinusoidal analogue signal. Thus, each time the incoming signal reaches 127 (midway) with a positive (rising) slope then the current value in the timer is stored in the variable "period" and the timer is set to 0.
The next step is to determine the frequency of the sinusoidal signal. This is done by dividing the timer rate (how fast the signal moves from a rising to a falling slope) by the period (which is the time between the midpoints). The result is printed on the serial monitor in Hz.
Lastly, the actual audio recording of the air conditioner "beeping" sound for the ON command was analysed by using Audacity. The frequency spectrum was plotted, and the fundamental frequency could be identified. The peak frequency was read off as 3917 Hz, which falls in a B7 musical key. The "beeping" sound for the OFF command was analysed as well and it was determined that the peak frequency was 3969 Hz, which also landed in the B7 musical key region.
This made it difficult to ask the microcontroller to distinguish between the two signals due to them falling within the same frequency range. Thus, a more precise algorithm should be developed for future prototypes.

4.4. Wind sensor

The wind sensor was implemented by directly implementing it with the Arduino Mega. The Rev C wind sensor comes with sample code which analyses the resistor temperature on the chip and the wind that blows onto the resistor. When no wind is blowing, the temperature rises until it settles at a temperature of 28 ℃. When the wind blows on the resistor, the resistor temperature drops and from this voltage spike (which tries to keep the resistor at a constant temperature) the windspeed is determined. These values are determined by an iterative fashion so minimal code changes were made.

4.5. APC220 RF communication module

The APC220 RF modules are implemented by firstly calibrating the devices, using RF Magic software, so they are setup to communicate with each other on the same radio frequency channel. After both modules are setup, testing can commence. The following figure shows how the RF modules were tested. The master controller gives the slave controller a command to turn on its LED when a button is pushed on the master controller's side. This was the first step to implementing the master – and slave controller units.

4.6. The master controller unit

The master controller unit consists of an APC220 RF communication module, LCD, pushbuttons, real time clock (RTC) and a battery power source. The LCD screen is used to monitor the sensor data, to monitor the battery status of the slave controller unit, to monitor whether IR commands have been sent and to monitor whether the sound sensor has detected a "beeping" sound from the split unit air conditioner.
The RTC is to keep track of time and to do time wise (TOU) functions. The push buttons navigate through the menu and selects the command that needs to be sent via RF communication to the slave controller unit.

4.7. The slave controller unit

The slave microcontroller unit consists of a DS18B20 temperature probe, an APC220 RF communication module, a wind sensor, a sound sensor (which is embedded underneath the controller) and an IR transceiver.
The slave controller unit is also battery powered. It communicates with the slave controller unit via RF communication, and receives IR commands that it must implement on the split unit air conditioner. Furthermore, it sends sensor data to the master controller unit to verify the operational status of the split unit air conditioner.
The sensor readings consist of a sound sensor which is used to verify whether the air conditioner is ON, a wind sensor and a temperature sensor which verifies that the air conditioner is operating in the correct state. Lastly, a wind chill temperature is calculated using both wind – and temperature sensors which gives

the direct temperature "feel" of the airflow coming from the air conditioner.

Fig. 10: Fully Implemented Slave Controller Unit.

5. Testing and evaluation

The testing and evaluation of the system began as soon as the entire system was fully integrated. The following tests were conducted and were based on actual results. The tests were as follow:

- Can the master controller unit communicate with the slave controller attached to the A/C;
- The APC220 RF communication module sends a request to the slave controller. The command which was sent was a "turn ON" IR command. The command which was to the slave unit via RF communication, triggered the split unit air conditioner to change its state to "ON". This confirms that the slave controller unit can receive IR commands from the master controller unit.
- Does the slave controller unit acknowledge that the air conditioner has turned ON by detecting the "beep" sound;
- After the air conditioner has turned ON and has made a beeping sound, the master controller receives a verification that the status of the air conditioner has been turned ON. This confirms that the slave controller has detected the "beep" and that it has transmitted the verification data to the master controller unit, which in turn has changed the status of the split unit air conditioner to the "A/C is ON" mode.
- Can the master controller change the state of the air conditioner to any other mode;
- The master controller unit receives a command from the user (by pressing the select push button). The instruction selected is "Fan Mode". The instruction is transmitted via RF communication to the slave controller unit. This can be said because the air conditioner has changed it state to the correct mode.
- Can the master controller receive sensor data from the slave controller unit;
- The user navigates to the "sensor readings" part of the master controller LCD user interface. The temperature sensor is selected. The master controller requests a temperature sensor data input from the slave controller unit. The slave controller acknowledges the command. The master controller starts displaying temperature data on the LCD screen.
- Does it also work if the master controller requests wind sensor data from the slave controller;
- The user navigates to the wind sensor section of the menu and selects the "Get Wind Data" button. The request is sent to the master controller via RF communication to the slave controller unit. The slave controller unit acknowledges the command and transmits the wind sensor data to the master controller unit. Wind sensor data starts to appear on the LCD user interface on the master controller unit side.
- Can the master controller get the wind chill temperature from the slave controller unit;
- The user navigates to the "Get WCT" part of the menu and presses the select button. The master controller unit sends a request to the slave controller unit to retrieve and send wind chill temperature data back to the master controller. The slave controller unit acknowledges the command and transmits the data via RF to the master controller unit. The wind chill temperature starts to appear on the LCD user interface on the master controller side.
- Can the master controller receive the battery status of the slave controller unit;
- The user navigates to the "Get Battery Status" part of the menu and presses the select button. The master controller unit sends a request to the slave controller unit (via RF) to retrieve and transmit the battery status data. The slave unit acknowledges the request and transmits the battery status data to the master controller unit. The battery status of the slave controller stars to appear on the LCD monitor.
- Can the split unit air conditioner receive a command from the IR remote control;
- The user picks up the air split unit air conditioner remote and points it to the IR receiver. The user selects the mode button on the remote control. The slave controller receives the IR command by decoding and interpreting the command. The slave controller accesses the correct IR command and transmits it to the split unit air conditioner. The state of the split unit air conditioner changes to a different mode of operation.
- Does the split unit air conditioner switch off when after business hours is being implemented;
- The master controller unit monitors the RTC data and check the after-business hours state to determine whether it is time to switch the air conditioner OFF. If it is 17:15 on a weekday or 13:15 on the weekend, then the after-business hours are activated. The master controller unit transmits an IR command, to the slave controller, to switch OFF the split unit air conditioner. The slave controller receives the command and implements it on the air conditioner. The sound sensor listens for a "beeping" sound and, if the sound is detected, the slave controller unit sends the verification data to the master controller unit. The master controller unit receives the verification and displays "beep detected" on the LCD. This is confirmation that the split unit air conditioner has indeed been turned OFF.

Because it was stressed that the system remains non-intrusive, it made it hard to determine the actual power consumption. The method that was followed to determine estimate power consumption was by taking reading from the power meter located inside the house. Its units are given in KW/h. The split unit air conditioner was switched on and the mode was changed on a 5min interval. In this time data, relevant to the current state it was in, was gathered.

By punching in the code sequence "i001", the power meter displays current power usage which is displayed in KW/h. From this, power consumption data was gathered and an estimate average power for each of the following operational states was determined. The interval over which the power consumption was measured was 5 minutes.

- Cooling Mode at 21°C had an average power consumption of 760 KW/h after a 5-minute interval.
- Dry Mode had an average power consumption of 862 kW/h after a 5-minute interval.
- Fan Mode (Auto) had an average power consumption of 94 kW/h after a 5-minute interval.
- Heating Mode had an average power consumption of 990 kW/h after a 5-minute interval.

From these estimate power consumptions, it is clear that switching your split unit air conditioner on fan mode will save you a lot on your electricity bills as well as lower the power demand on the power grid. Hence making it possible to see huge improvements in power grid stability when all the split unit air conditioners are switched to fan mode during peak times when the electrical grid is under stress.

6. Conclusion

The program running on the slave controller unit as well as the program that runs on the master unit has been modified quite a few times due to program desynchronization faults that occurred when the entire system was integrated. The effect of program desynchronization became apparent when the sound sensor was asked to listen for a “beep” as soon as the split unit air conditioner has been switched on. The timing between these two components had to be sorted out to ensure that the program timing window between the master controller unit and the slave controller unit overlapped at exactly the correct time.

The sound sensor performed as was expected and the frequency detection algorithm detected the “beeping” sound of the split unit air conditioner. The after time wise functions i.e. after business hours protocol is quite a useful feature to avoid electrical waste from occurring in the case where the air conditioner was left on due to forgetfulness. The estimate power consumption of the split unit air conditioner was calculated by using the power meter located inside the house to see what the current power consumption of the air conditioner was under different operational states i.e. cooling mode, dry mode, heat mode and fan mode. The time interval chosen to do this calculation was 5-minutes. From the data, it was calculated that an estimate of 87% reduction in load could be obtained when the air conditioner was changed from cooling mode to fan mode (when the compressor was off). It was stated initially, that HVAC were responsible for 26% of the total power consumption in the commercial sector. This means that since with this system, the HVAC load of the commercial sector can be reduced by 50%, then the result of implementing this system would be a 13% reduction in the power demand which in turn would help balance the electrical grid by reducing the stress it endures in peak times. Recall that he aim of this paper was to present a new energy monitoring and verification control interface for the Eskom demand market participation (DMP) program, from the results presented, it can be said conclusively that this aim was achieved.

References

[1] Eskom, "Eskom Load Shedding," [Online]. Available: http://loadshedding.eskom.co.za/loadshedding/description.

[2] Generation Communication, "Eskom.co.za," January 2015. [Online]. Available: http://www.eskom.co.za/AboutElectricity/FactsFigures/Documents/TD_0002PowerFailuresRev9.pdf.

[3] R. S. a. D. Blane, "Creating a virtual power station to avoid load shedding through customer participation," in 25th AMEU Technical Convention 2015, Johannesburg, 2015.

[4] 2. I. Conference, "Domestic Use of Energy Brochure," 3-5 April 2017. [Online]. Available: http://energyuse.org.za/due/.

[5] C. Openshaw, "South African Management Association," KZN, 2015.

[6] Wifipoweroutlet.com, "AUKEY Wifi Smart Air Conditioner Socket Remote Control Switch with Smartphone APP," [Online]. Available: http://wifipoweroutlet.com/aukey-wifi-smart-air-conditioner-socket-remote-control-switch-with/. [Accessed 2017].

[7] AirPatrol, "AirPatrol WiFi - Smart AC Control," [Online]. Available: http://www.airpatrol.eu/shop/airpatrol-wifi/.

[8] Freshome, "Control Your Air Conditioning System via Smartphone: tado° Cooling," [Online]. Available: http://freshome.com/2014/05/18/control-air-conditioning-system-via-smartphone-tado-cooling-video/.

[9] Ambi Labs, "Ambi Climate Holistic AC Comfort," [Online]. Available: https://www.ambiclimate.com/.

[10] M. Hollestein, "Demand Response Implementation for Split-unit Air Conditioners," North West University, Potchefstroom, 2016.

[11] A. Ghassaei, "Instructables," 29 Septemer 2017. [Online]. Available: http://www.instructables.com/id/Arduino-Audio-Input/.