

Design of a Powerline Home Automation System

Final Report

G.A. Richter
9719216

Submitted as partial fulfillment of the requirements of
Project EPR400
in the Department of Electrical, Electronic and Computer Engineering
University of Pretoria

November 2000

Study leader:
Gerhard Korf

Preliminary

Summary

This report describes work that I did on embedded control data communication using the domestic powerline circuitry as channel medium. A home automation system was implemented.

What has been done:

- I did a literature study on the usage of the powerline as communication medium and the various techniques used to implement networks.
- I did a literature study on error detection and error elimination methods.
- I did a literature study on Internet communication and security.
- I designed and constructed a home automation system comprising of one master unit and four slave units communicating across the powerline.
- I wrote client software using JavaScript and HTML (Hyper Text Mark-up Language) through which the user can send commands via the Internet.
- I wrote software in C++ that controls the network from a PC (Personal Computer).
- I wrote server software in C++ that interfaces the client from across the Internet to the home automation network.

What has been achieved:

- I found a modern home automation system could be designed in such a way as to improve the quality of life of the owner while improving the security and providing off-site control.
- I found that data communication using the powerline as medium has enormous potential for growth and is under exploited, especially in local area networking environments where cabling costs can be eliminated.
- I found that security is lacking in the Internet architecture and without careful planning a network is highly vulnerable to abuse from an outsider.

There is a technical report attached to this document, in which additional source code listings, circuit diagrams and other technical data appear.

The work done in this project does not build on any specific previous projects. I hereby certify that all the work described in this document is my own.

G. A. Richter

Date

Table of Contents

List of Abbreviations

ASK	Amplitude Shift Keying
CMOS	Complementary Metal Oxide Semiconductor
CRC	Cyclic Redundancy Check
DHTML	Dynamic Hypertext Markup Language
FCS	Frame Check Sequence
FSK	Frequency Shift Keying
HTML	Hypertext Markup Language
HTTP	Hypertext Transfer Protocol
IC	Integrated Circuit
LDR	Light Dependant Resistor
LED	Light Emitting Diode
LSB	Least Significant Bit
MSB	Most Significant Bit
PLL	Phase Locked Loop
RMS	Root Mean Square
SCR	Silicon Controlled Rectifier
SNR	Signal to Noise Ratio
TCP/IP	Transfer Control Protocol / Internet Protocol
UART	Universal Asynchronous Receiver Transmitter
VCO	Voltage Controlled Oscillator

1. Introduction

1.1 Problem Statement

In the modern home filled with electronic apparatus and appliances, it is useful for the owner to exercise some form of centralised control over the functions in the house. Currently, when the owner needs to turn on the driveway and living room lights before arming the alarm and going to bed, he has to walk to the garage, then to the front door, then to the alarm box, then to bed.

When the owner leaves his house to go on holiday, he forfeits all control over the functions of the house while he is away and cannot tell whether someone has breached the security or whether he had left the bedroom light on. If the alarm has been triggered at his home, there is no way that the owner can become aware of this unless he returns to his compromised house.

In order for the system be useful, the appliances must be able to be moved around the house and still retain their ability to communicate with the system. A degree of automation is needed in a house so that certain functions in the house occur automatically, for example the outside light can turn on when it becomes dark outside.

There is a need for a reliable, secure and interactive system that exercises full control over the electric and electronic aspects of the house, with the potential to be accessed from across the globe. No current solution offers Internet connectivity and most others are expensive and difficult to install.

1.2 Background

1.2.1 Centralised control

Integration of the functions in a house or office building has been an ongoing study. The networking of devices within a confined space such as a house or office building has found many uses and there have been numerous studies in this field.

Idea behind a centralised function building is that a single unit has master control over all the slave units attached to it via the network. The user enters the commands at the master unit in order to exercise control over all the slave units and receives his feedback from the master unit. The master unit is typically implemented as a computer communicating with the slave units via a physical network layer such as a radio frequency (RF) link or the domestic powerline circuit.

As shown in Figure 1.1, centralised control uses a bus topology with a master that controls the data flow.

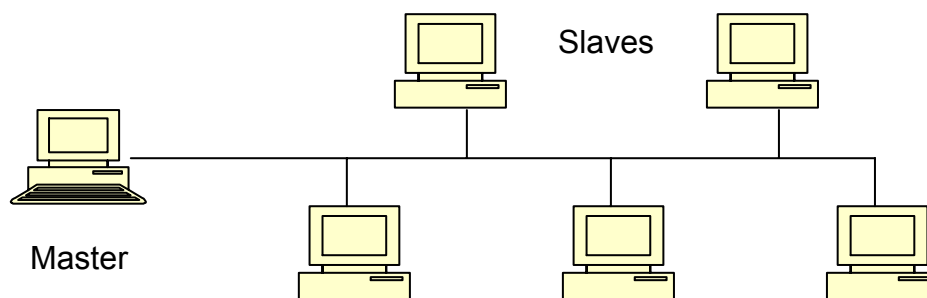


Figure 1. The generic bus topology. Each networked appliance receives commands from the master unit. Figure is adapted from O'Neal [1].

1.2.2 Communications medium

The medium through which communications occurs is crucial to the feasibility of a home automation system. Radio frequency links have been attempted with success but reports¹ indicate that these systems are difficult to implement and are relatively expensive. The available frequency bands that can be used are getting less and less and licensing is becoming prohibitively expensive. Obstacles between the master and slave units attenuate and distort communication signals and the bit error rate becomes relatively high. This method is impractical for home automation use, where cost effectiveness is crucial, although bit error rate need not be very low.

¹ Scott, J. J., Home Networking using Radio Communications, "<http://www.geocities.com/userb/radiocomms.html>"

The communications medium found to be most effective home automation networking is the powerline wiring as reported by O'Neal [1] and McArthur, Wingfield and Witten [2]. This medium is present in all modern houses and offices and presents a opportune solution since all the networked devices are attached to this medium anyway, since they require electric power from it, hence no new wiring is required and costs are dramatically reduced.

Chan and Donaldson [3] and Vines and Trussel [4] report that there are two major impediments when using the residential powerline as communication medium, namely noise and attenuation.

Noise on residential powerline circuits

The 50-200 kHz band of frequencies typically used in powerline communications has been the study for the effects of noise by Vines and Trussel [4]. This study involved the placement of a transmitter on the secondary side of the residential transformer and the measurement of the signal to noise ratio (SNR) at various locations in the building. It was found that the primary sources of noise in residential environments are universal motors, light dimmers and televisions. The noises can be classified in three different categories.

50 Hz periodic noise

Noise synchronous to the sinus powerline carrier can be found on the line. The sources of this noise tend to be silicon-controlled rectifiers (SCRs) that switch when the power crosses a certain value, placing a voltage spike on the line. This category of noise has a line spectra at multiples of 50 Hz.

Single-event impulse noise

This category includes spikes placed on the line by single events, such as a lightning strike or a light switch turn on or off. Capacitor banks switched in and out create impulse noise.

Non-synchronous periodic noise

This type of noise has line spectra uncorrelated with the 50 Hz sinus carrier. Television sets generate noise synchronous to their 15734 Hz horizontal scanning frequency. Multiples of this frequency must be avoided when designing a communications transceiver.

It was found that noise levels in a closed residential environment fluctuate greatly as measured from different locations in the building. Noise levels tend to decrease in power level as the frequency increases, in other words, spectrum density of powerline noise tends to concentrate at lower frequencies. This implies that a communications carrier frequency would compete with less noise if its frequency is higher.

Chan [3] also found that a large amount of noise enters the line at frequencies of 400 kHz and higher, as this band corresponds to the AM radio band, where the powerline wiring acts as a good antennae at these frequencies, creating noise. On the other

hand, frequencies lower than 100kHz tend to contain noise inversely proportional to frequency. This is illustrated in figure 2.

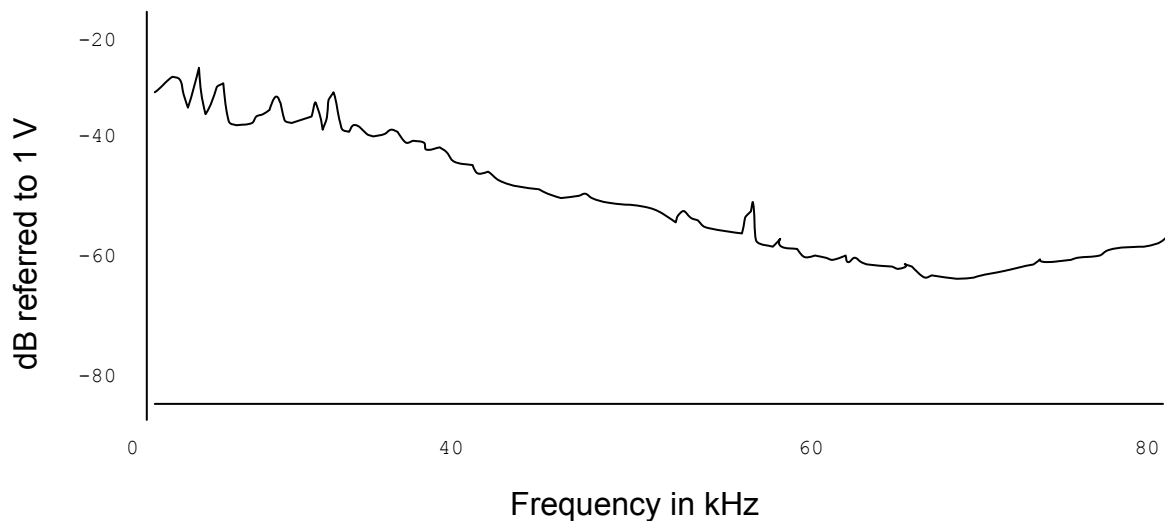


Figure 2.
Measurements of noise spectrum on a residential powerline. Figure taken from Vines and Trussel[4].

Signal attenuation on a residential powerline

A study done by Chan and Donaldson [3] shows that the signal attenuation is neither constant nor linear on the residential powerline over the 20-240 kHz band, as shown in figure 3. The main factor that causes the data signal to become attenuated is that the impedance along the line is very low and drops as loads are encountered. Except over short distances, attenuation normally exceeds 20dB but can be much higher and the design of error-control codes, signal formats and communication protocols are essential in the hostile powerline environment.

It was found that over short distances of approximately 10 meters the attenuation is fairly flat at 5 dB and then increases with distance, across the entire frequency range measured. Over longer but unknown² distances, the attenuation is approximately 25 dB for frequencies below 60 kHz and increases to about 50 dB at 250 kHz, as shown in figure 3.

Narrow frequency band fading also occurred, resulting in periodic attenuation over narrow bands in the frequency range tested. These fading mechanisms are erratic and the modeling of them is highly complicated.

Loads that were present on the line during testing were a radio, cassette recorder, vacuum cleaner, razor, sewing machine, fan and a fluorescent lamp, but the removal of these from the line showed no significant difference in the received signal. An electric kettle and television set both caused large increases in attenuation when connected to the line.

² Distances are unknown due to the unavailability of circuit diagrams and the inherent complexity thereof, but were estimated by Chan and Donaldson [3] to exceed 120m but could be as long as 300m.

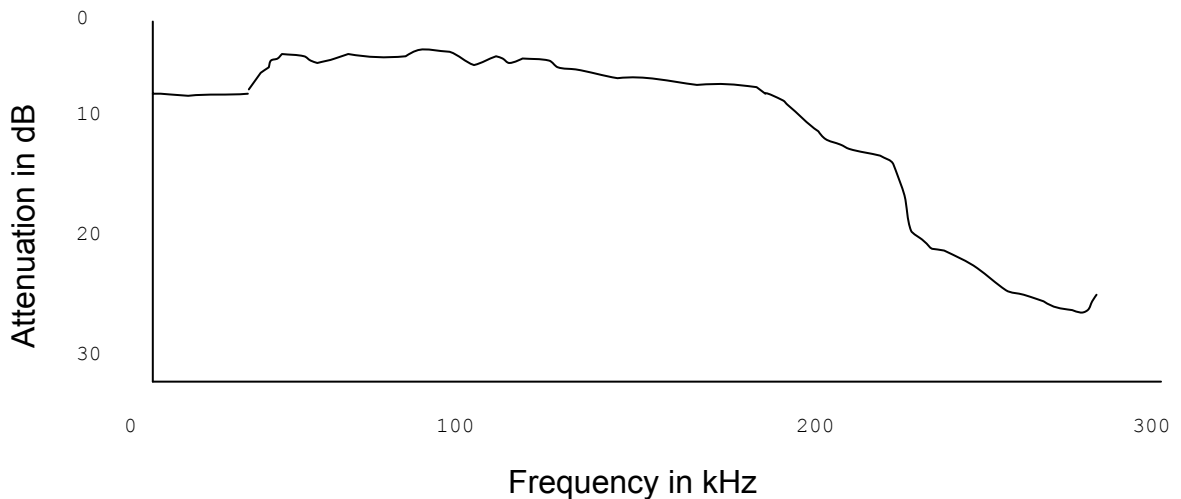


Figure 3.
Signal attenuation on a powerline across the 0-300 kHz frequency band. Figure taken from Chan and Donaldson [3].

From the above analyses of noise and attenuation it is clear that choosing a good carrier frequency for communications is a compromise between high noise, but low attenuation at low frequencies, and low noise and high attenuation at high frequencies. The band on frequencies on the powerline that is desirable for communications lies between 100kHz and 200kHz, giving the channel a bandwidth of 100kHz.

1.2.3 Coupling of the signal

Once the data signal has been generated, it needs to be placed on the powerline by some kind of coupling network. The idea is to superimpose the data signal onto the 240 V, 50 Hz power waveform, and extract it afterwards at the receiving end.

McArthur, Wingfield and Witten [2] argue that there are three possible combinations of lines on which to couple the signal: live to ground, neutral to live and neutral to ground. Neutral to ground has the advantage of safety, but also the disadvantage of the fact that neutral is usually grounded at the transformer, so no interbuilding communications can be made and the line impedance is too low, causing large power amplification requirements.

Coupling methods use a filtering technique to place the signal onto the line and remove the 50 Hz powerline carrier. There are two commonly used methods to implement the filter.

An isolation transformer forms part of the bandpass filter that removes the 50Hz carrier. The inclusion of an isolation transformer is for safety, otherwise a hazardous situation could be caused by operator ignorance.

Filter Design

Secondly, as shown in Figure 4, an LC coupling and filtering network can be designed, omitting the transformer. This method is preferred as it is more economical and the engineer has more direct control over the filter response.

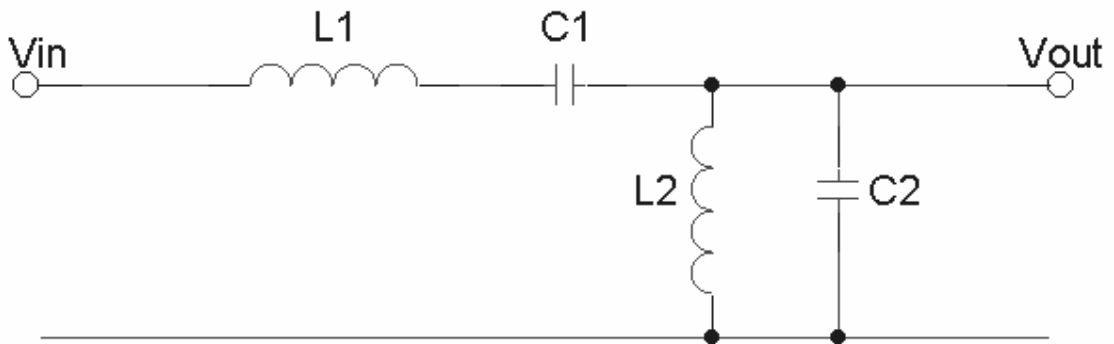


Figure 4.
General form of a 4th order LC filter.

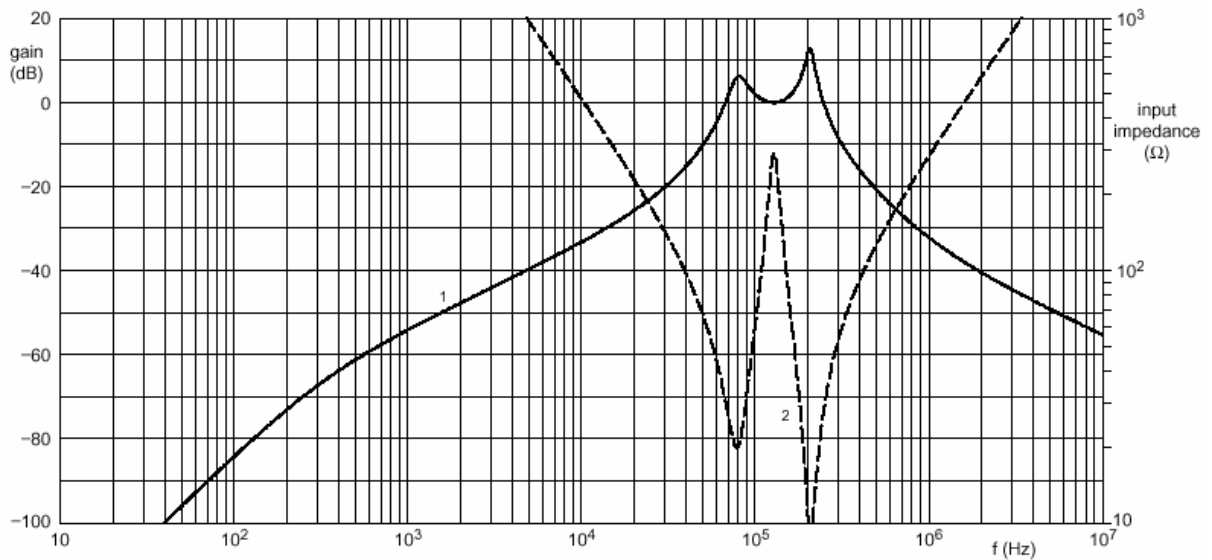


Figure 5.
Response of the LC filter with $C_1, C_2=33\text{nF}$ and $L_1, L_2=47\mu\text{H}$. (1) is the filter gain and (2) is the input impedance into the filter. Taken from Philips [10].

1.2.4 Message Coding Techniques

When using the powerline as communications medium, it is important to understand the channel well, as it is a harsh environment for signals to propagate in.

In a medium that carries any amount of noise, it is possible to transmit data reliably (that is, with an error probability of zero) as long as the data rate is below a certain limit known as the channel capacity. This limit is defined by Shannon (1948) in the Noisy Channel Coding Theorem:

“The basic limitation that noise causes in a communication channel is not on the reliability of communication, but on the speed of communication.”

The channel capacity of an additive white Gaussian noise channel is given by Proakis [9] as

$$C = W \log\left(1 + \frac{P}{N_0 W}\right) \text{ bits/s} \quad (1)$$

W is the channel bandwidth, P the signal power, N_0 the noise power spectrum.

This theorem is very important, as it means that on a noisy powerline circuit, home automation signals can be sent reliably, as long as the rate of transmission is low enough. Since a home automation system does not require high transmission rates³, the system can be truly reliable. One way of reducing the message transmission rate is to introduce ample redundancy into the data stream. Extra bits are placed into each data word and are used to detect and correct errors.

The method employed in this project is combined use of a CRC (Cyclic Redundancy Check) algorithm and simple parity checking.

The CRC method is one of the most widely used and most powerful error-detecting codes and was thus chosen for the noise-prone home automation system. Given a message of k bits, the transmitter generates an n -bit sequence, so that the combined block of bits $k + n$ is exactly divisible by some predetermined number. The receiver can then divide the incoming block by that same number, and if it leaves no remainder, the block can be assumed to be without errors.

The following modulo 2 arithmetic⁴ approach to CRC is given by Stallings [11]:

- T = $(k + n)$ -bit frame to be transmitted, with $n < k$
- M = k -bit message, the first k bits of T
- F = n -bit FCS (Frame Check Sequence), the last n bits of T
- P = Pattern of $n + 1$ bits chosen beforehand

From above,

$$T = 2^n M + F$$

³ The home automation system developed only sends single word instructions consisting of 9 bits. If the maximum comfortable time for an instruction to be executed is assumed to be 100 milliseconds, then the bit rate required is only 80 bits per second. Using (1), we find that the theoretical channel capacity for a domestic powerline at an SNR (Signal to Noise Ratio) of 20dB and a bandwidth of 100kHz, is **200,432 bits/s**.

⁴ Modulo 2 arithmetic is essentially binary addition or subtraction with no carry operations performed, which is identical to the XOR operation. Stallings [11] gives the following examples of modulo 2 arithmetic:

$$\begin{array}{r} 1111 \\ + 1010 \\ \hline 0101 \end{array} \quad \text{or,} \quad \begin{array}{r} 1111 \\ - 0101 \\ \hline 1010 \end{array}$$

T/P should have no remainder. If $2^n M$ is divided by P ,

$$\frac{2^n M}{P} = Q + \frac{R}{P}, \quad (4)$$

Q is the quotient and R the remainder, then R is always 1 bit shorter than Q . R is then used as the FCS,

$$T = 2^n M + R$$

To show that R satisfies the requirement that T/P must have no remainder, consider the following:

$$\frac{T}{P} = \frac{2^n M + R}{P}$$

Now, substituting equation (4),

$$\frac{T}{P} = Q + \frac{R}{P} + \frac{R}{P},$$

but when using modulo 2 arithmetic, any binary number added to itself yields zero, so,

$$\frac{T}{P} = Q + \frac{R+R}{P} = Q$$

which is exactly divisible by P , since there is no remainder.

Thus, to generate the FCS used in the CRC is very simple, $2^n M$ is divided by P and the remainder is the FCS.

1.2.5 Amplitude-Shift Keying (ASK)

In order to send the data to the receiver in a chosen frequency band, it is necessary to modulate the data.

Using this technique, the binary input data is modulated as follows:

$$\mathbf{ASK} \quad a(t) = \begin{cases} A \cos(2\pi f t) & \text{Binary 1} \\ 0 & \text{Binary 0} \end{cases} \quad (2)$$

This modulation technique is highly noise resistant, since all the information is sent at one specific carrier frequency, so the bandwidth of the signal is small and narrow-band filters can be designed to reduce the noise throughput effectively.

The typical frequency spectrum of a binary message modulated using an ASK carrier can be seen in figure 6 and an example ASK waveform is shown in figure 7.

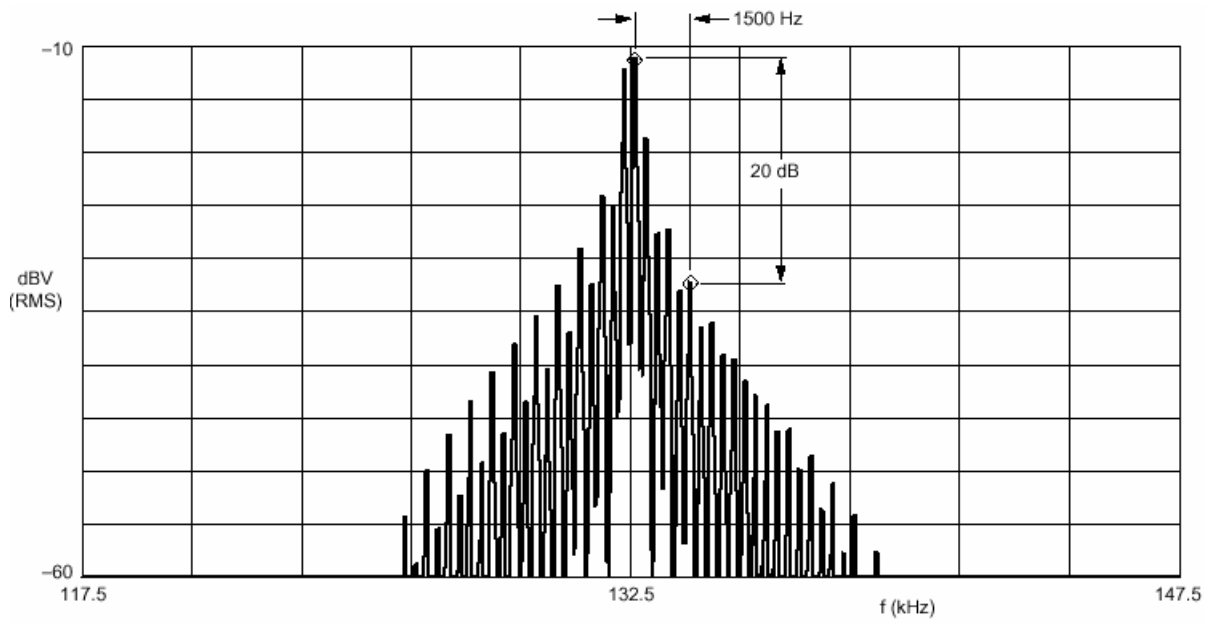


Figure 6.
The frequency spectrum of a random binary message modulated using ASK.
The carrier frequency was 132.5 kHz. Taken from the TDA5051 databook [10].

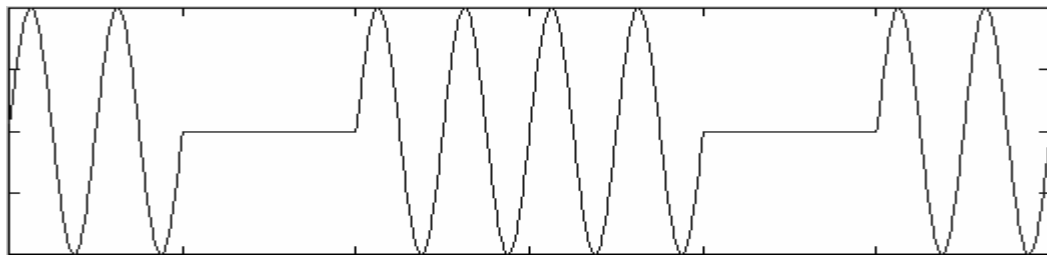


Figure 7.
An ASK waveform representing the binary word 101101.

1.3 Objectives

The home automation system will have a master unit that will control all the slave units across the powerline medium. Whilst designing the network it is necessary to develop a system that seamlessly integrates all electric apparatus in the home or office. A centralised command unit is needed and must be able to send appropriate control commands via the powerline wiring to the entire variety of devices attached to the network. Each device, whether it is a kettle or an alarm system, must be able to identify calls sent to it and change its state accordingly. Sensory devices must be able to report back to the central box. The system must be fully interactive via a secure Internet connection to the control box. The transmission of signals must be reliable under noisy conditions.

1.4.1 Functional Design

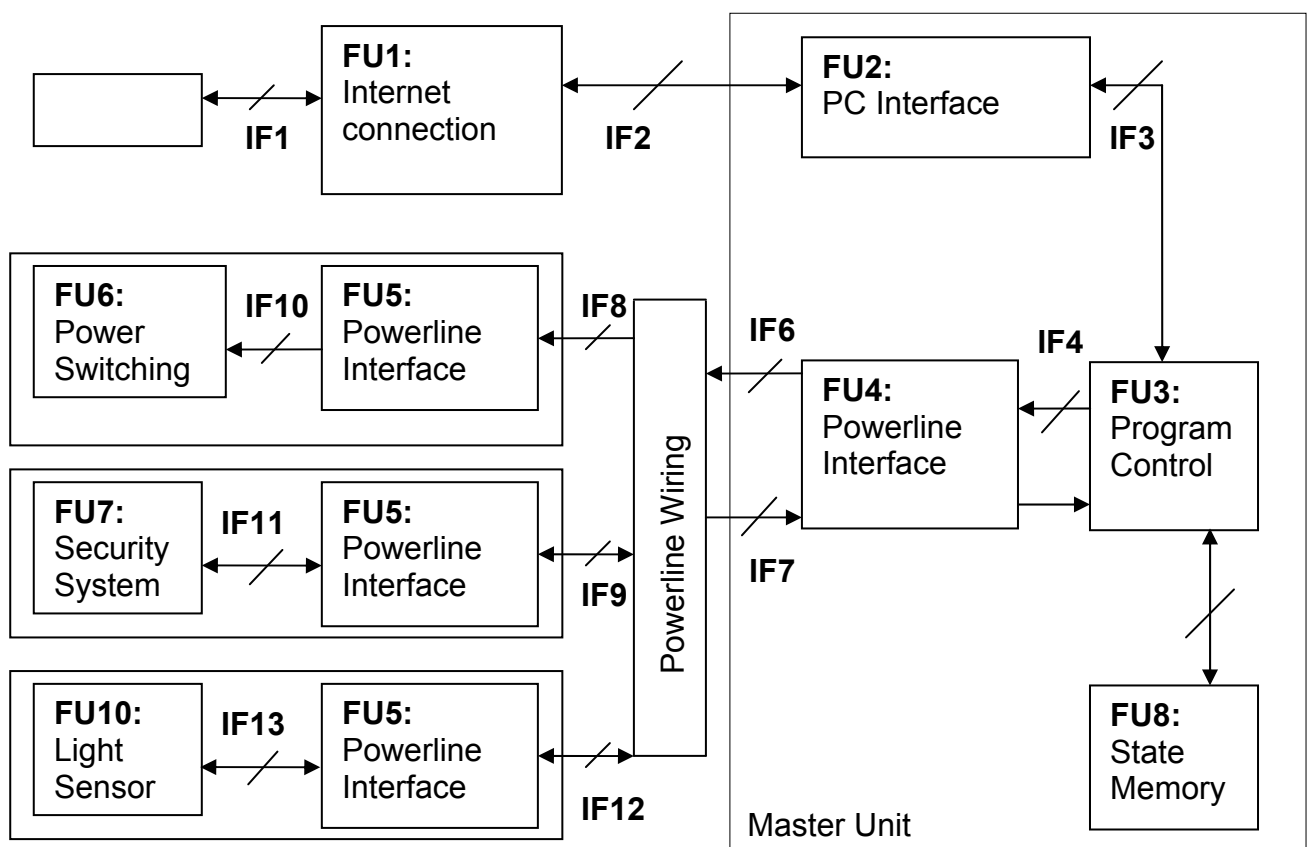


Figure 8.
Functional design of the home automation system.

The system will be able to read commands from the user entered via the Internet and captured to the control box via a serial connection to the PC serial port. These commands will specify whether certain lights or appliances in the house must be turned on or off and at what times these must occur. Commands to the devices will be sent via the 240 V AC (alternating current) powerline circuitry in the house and received by each appliance controller. Interactive units will provide feedback from sensors attached to them for example a light sensor to indicate night or day.

To provide for the automation aspect of the project, the goal is to develop user-friendly interactive software that allows the user to specify when the slave modules must be switched, using one or more of the following criteria:

1. Immediately,
2. At a specific preset date and time,
3. When a specified sensor turns on,
4. When a specified sensor turns off.

For example a bread toaster can be set to start at 6:00 in the morning for breakfast or a light can be set to turn on when it gets dark outside. The system must then be continuously monitor of the state of each sensor on the network.

1.4.2 System specifications:

- The user will exercise full control via a web site on the Internet with a TCP/IP connection or at the local PC (IF1),
- The control box will communicate with the PC at 1200 bits/s via RS232 (IF2),
- The system program can control up to 32 units (FU3),
- States of up to 32 units can be stored and queried via the Internet (FU8),
- All units will plug into 3-point sockets (IF8),
- The units will be compatible with a 240 V AC powerline circuit (IF6, 7,8,9),
- The system will be functional inside one single phase circuited house (IF6, 7,8,9),
- Each power switching unit must be able to deliver 4A current equivalent to 960W (FU6),
- Error elimination will be done using a combination of error detecting codes and repetitive transmission (FU5),
- Up to 32 units can be networked, each possessing a unique ID,
- A light sensor (FU10) will provide a signal to indicate light or dark,
- The existing security system will be provided with a signal to provide an arm/disarm function and the control box will receive a signal from the alarm system to indicate whether the security has been breached.

1.4.3 Deliverables

The following items comprise the project that will be delivered.

- A central control box, connecting to a PC via an RS-232 cable serial link and connecting to the powerline network via a 3-point plug,
- Web server software, web client software and source code to allow access to network via the Internet,
- A device connecting to powerline network via a 3-point plug and allowing the switching of an appliance via a 3-point socket,
- A light switching device that plugs into a 3-point socket and allows a light bulb to be plugged into it,
- A sample security system connecting to the network to illustrate the interface with a more complex existing security system,
- A light sensor device that reports its status when the ambient light level drops below a certain adjustable level.

1.4.4 Verification

Verification of the specifications will be done in a laboratory. Each slave unit will be plugged into the powerline using 3-point plugs. A computer will be connected to the master unit, which in turn will be plugged into the powerline.

Due to the nature of the project, the main verifiable goal will be to check that a light bulb and another electric device switch on and off in response to user commands from the PC from across the powerline, the success of which will be readily apparent. Similarly, the software will respond to a change in light level as reported by the light sensor. The light level can be altered simply by covering the light sensor with a cloth in a room with sufficient ambient light.

If the above tests are successful, it is clear that every aspect of the network system works correctly. Bit rate specification compliancy will be illustrated using an oscilloscope and power switching specification compliancy will be shown by measuring the current switched using a digital multimeter (DMM), as shown in figure 9.

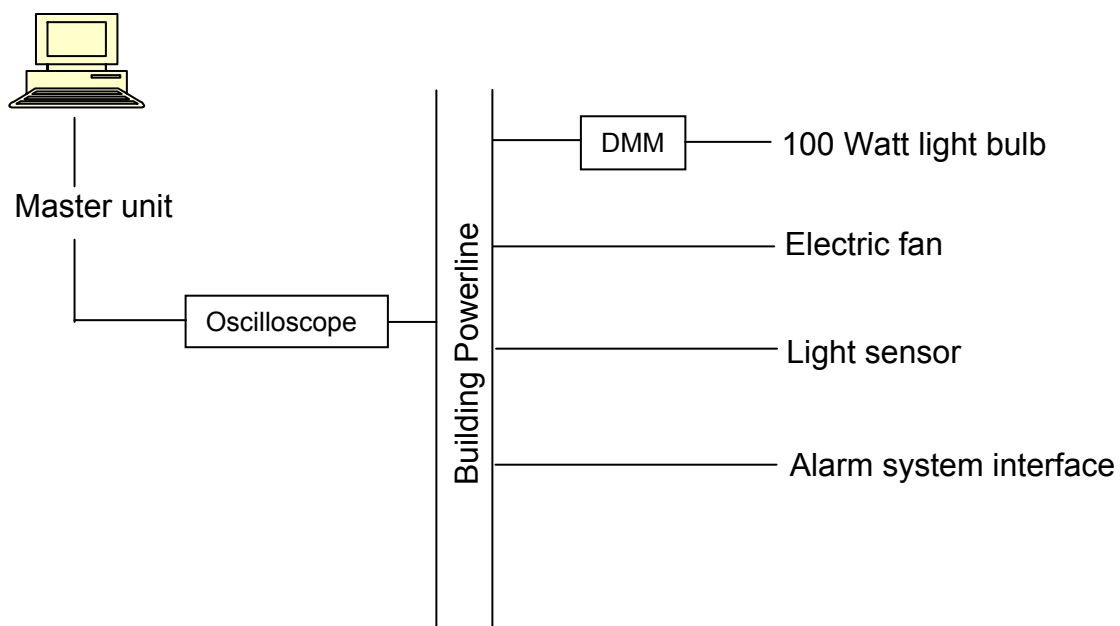


Figure 9.
The setup used to test the system and its compliancy to the given specifications.

1.4 Approach

An emphatic literature study on networking techniques, powerline interfacing, power switching, asynchronous serial communications, Internet interfacing and web security was done, to arrive at the following approach, deemed best for this specific project.

The approach used in the network is a bus topology, with a master/slave system. The master polls each slave unit for a reply and the slave units only use the communications bus after having been polled by the master unit.

The powerline home automation system comprises of the following parts:

1. A *client web page* through which the user programs the schedule,
2. A *web server program* that interprets the instructions and issues commands to the master unit,
3. The *serial interface* between the computer and the master unit,
4. A master unit *microcontroller* that encodes and times the instructions to be sent across the network,
5. A *signal modulator/demodulator*,
6. A *line filter* that isolates the high voltage 240 V, 50 Hz carrier from the rest of the circuitry and couples the signal onto the powerline,
7. Another *microcontroller* that resides in the slave modules, decoding the instructions and controls the slave function,
8. The *slave function module*, which is one of the following:
 - 8.1 A *power switching module* to switch an appliance on or off,
 - 8.2 A *light sensor* to report on the ambient light,
 - 8.3 A *simple illustrative alarm system*, which can be interfaced to a larger existing system.
9. The module *power supplies*.

Each of the above parts will be briefly discussed:

1.4.1 The client web page

A combination of HTML, JavaScript and DHTML (Dynamic Hypertext Markup Language) was used to construct the web page. The basic elements of the page were created using Microsoft FrontPage Express 2.0, and then finalised using a basic text editor.

The schedule works on an event-driven basis, a user can specify when a command is issued across the network by choosing an event which must first occur. The event and accompanying data is stored in a schedule on the server which the user can browse and edit.

When the user has chosen and entered all the event data in a form, the form is then preprocessed and parsed by a JavaScript script run in the user's browser and sent to the server across the Internet using the CGI (Common Gateway Interface) protocol.

1.4.2 The server CGI program

A server CGI program is invoked upon reception of the parsed command from the client and updates the locally stored schedule accordingly. It then signals another server-side monitoring program that the schedule has been updated and then terminates. The continuously running monitoring program keeps track of the time and periodically polls the sensor slaves of the network in anticipation of an event. If an event occurs corresponding to one stored in the schedule, the appropriate command is issued to the slave. Confirmation of the command is received via the network and reported back to the user.

1.4.3 The serial interface

The server-side monitor sends and receives instructions using the computer's built-in UART (Universal Asynchronous Receiver Transmitter), using a serial data speed of 1200 bits/s. On the master module, attached to the computer with a 9-way cable, the RS-232 signals are translated into 5 volt binary signals using a Maxim MAX232 RS-232 transceiver.

1.4.4 Master microcontroller

At the heart of the master unit lies a Microchip PIC16F876 microcontroller. It has a built-in UART module that is used to communicate via the computer and with other devices on the network. The communication settings used with other units are also 1200 bits/second, odd parity and 1 stop bit. Then serial data stream from the UART is multiplexed between the computer and the powerline using a Philips HEF4066BCC quad bilateral switch, wired as a dual multiplexer/demultiplexer.

The microcontroller adds an error control CRC code to the message and issues it to the network.

1.4.5 Modulation

The modulation technique chosen is ASK (Amplitude Shift Keying) and the serial data stream is modulated using a Philips TDA5051 ASK modulator/demodulator. Considering the noise and attenuation properties of the powerline as discussed in 1.2.2 on page 6 above, the most suitable carrier frequency is 125 kHz. A binary 0 corresponds to a burst of a sine carrier at 125 kHz, and a binary 1 causes no signal to be placed on the line.

1.4.6 Line filter and line coupling

The line filter is a passive LC filter, designed to filter out the 240 V 50 Hz to below 80 dB, whilst allowing the 125 kHz carrier to pass with attenuation of no more than 2 dB. The topology of the filter network is shown in figure 6 on page 10.

The output of the modulator is connected to the input port of the filter and the live and neutral connections of the powerline are connected to the output port of the filter.

1.4.7 Slave Microcontroller

The microcontroller used in the slave modules is the Microchip PIC16F84. The UART functions are implemented in software as well as the encoding and decoding of the instructions received. Each slave module on the network receives all the instructions sent by the master and compares its own unique network id with that in the instruction word id field and only executes the instruction upon a match.

1.4.8 Slave functions

Each slave can perform a different dedicated function depending its design. Although the system has been designed with expandability in mind and any conceivable function⁵ can later be added by 3rd party developers, only the following functions have been implemented:

Power switching

A triggered triac circuit is used to switch current through a load. The microcontroller provides a 5-volt signal to turn on the triac as shown in figure 10.

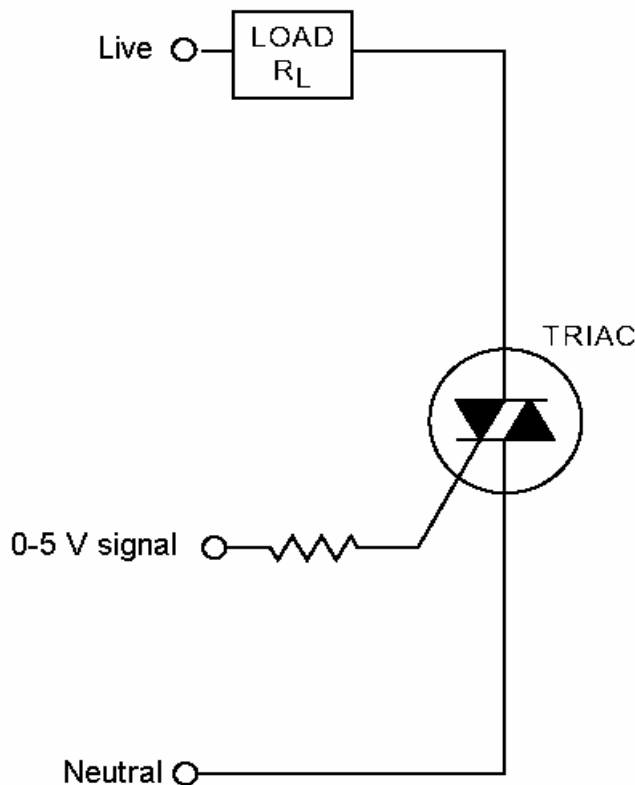


Figure 10.
The implementation of the power switching function. Adapted from AN1007 [12].

⁵ Additional slave modules must be of a switching or sensor nature. In other words they must receive an on/off signal and perform a toggle function, or must, upon being polled, return a binary value corresponding to a measured parameter. For example, a humidity sensor can be built that reports on whether the humidity is above or below an adjustable threshold value. More examples can include a telephone ring detector or sensor that detects whether a door or window is open. Up to 32 units of any nature can be dynamically plugged into the network.

Both power switching units that are delivered, the light switch and the general purpose 3-point plug switch, use this technique.

Light sensor

The light sensor unit uses an LDR (Light Dependant Resistor) to detect the ambient light level. The LDR forms part of a 2 resistor voltage divider between 5 volt and ground. The resulting voltage is compared to a fixed voltage produced by another voltage divider containing a potentiometer and a resistor, using an operational amplifier. The output of the amplifier is fed to the microcontroller as the sensor signal.

Alarm system interface unit

The alarm system delivered does not function as a complete home alarm system, but merely illustrates that the home automation system can interface with a larger existing alarm system.

The alarm interface unit provides the home alarm system with an arm/disarm signal and reports back to the master unit the current integrity status of the building.

1.4.9 Power supplies

All the units use a 5 volt power supply. It has been decided to use safety isolation-transformers for each of the units instead of drawing power directly from the mains using an adaptation network, purely for safety. A 240 - 12 V transformer is used to bring the voltage down, after which the power is rectified using a bridge rectifier and regulated using a 5 volt 100 mA voltage regulator (78L05).

2. Implementation

Each of the aspects briefly named in 1.4. will be discussed in detail in the following sections. Figure 11 shows how the description will follow.

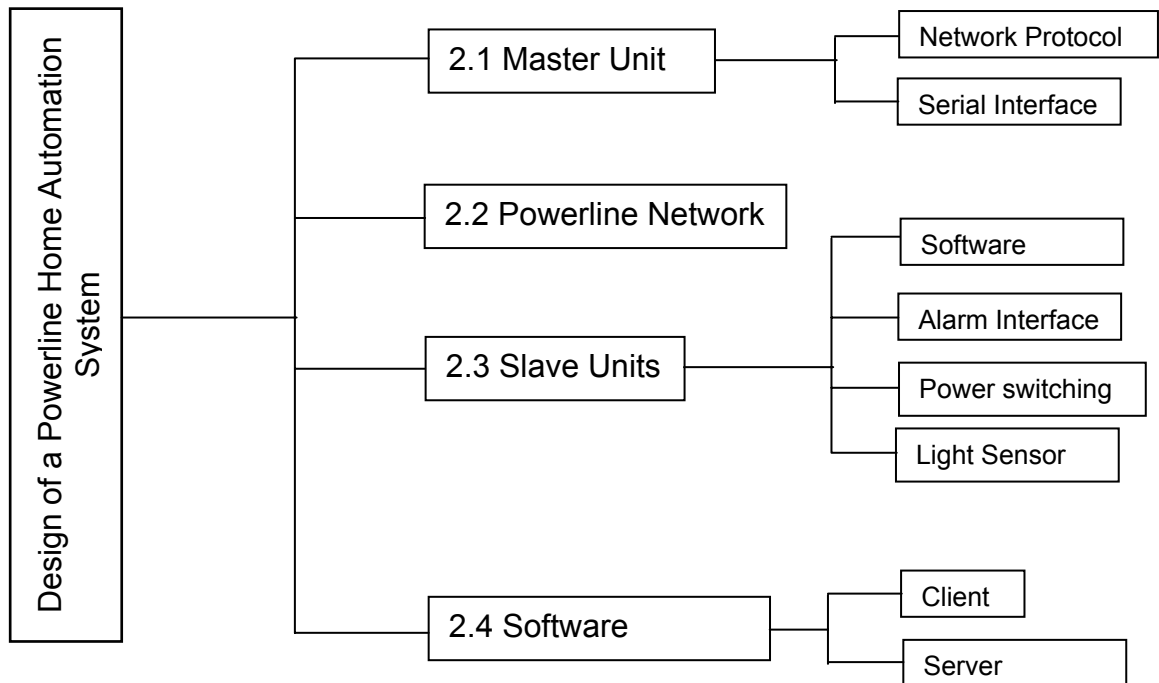


Figure 11.
A breakdown map of the implementation descriptions.

2.1 The Master Unit

The master unit wields control over all the slave units on the network. When the user issues a command to change the status of one of the slave units, the master unit receives this command via the serial port and sends the corresponding network data message over the powerline. Ten copies of the packet are sent with a 2 ms delay in between. It then waits for the slave unit to respond to confirm that the instruction has been carried out and forwards this information to the user via the PC.

The master unit consists of a microcontroller, a serial interface and a powerline interface. The microcontroller used is the Microchip PIC16F876. It is a 28 pin IC (integrated circuit) that contains a built in UART. Its low cost, high functionality and ease of use made it a good choice for the main controller of the network. The assembly source code used on the microcontroller is listed in Appendix 3.

2.1.1 Network protocol

Networking protocols that are currently being used in Local Area Networks across the world are overly excessive for such a low traffic networking environment found in the home automation system. A proprietary protocol has been developed for this project to exploit the exceptionally low traffic and bandwidth requirements. The synchronisation method used to transmit and receive data packets is the standard asynchronous transmission technique used in RS-232 communications. When no data is being transmitted, a constant binary 1 is present on the network medium, which corresponds to the absence of a carrier using the TDA5051 ASK modulator. The fields of a network packet are shown in figure 12.

Specifications of transmission:

1200 bits/s,
1 start bit, 1 stop bit,
Odd parity.

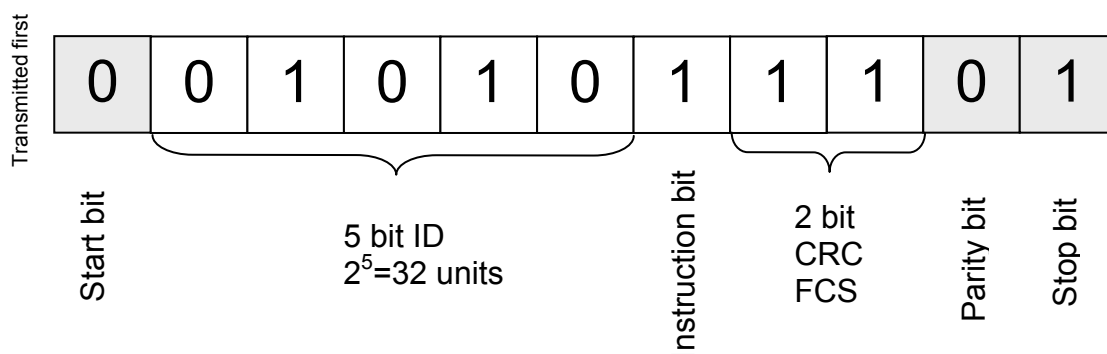


Figure 12.
The network data packet composition.

5 bit ID

The ID portion identifies the destination module, each being assigned a unique ID.

Instruction bit

This bit signifies whether the destination device must switch on (1) or off (0). In the case of a sensor, this bit is irrelevant, the mere recognition of the ID will cause the sensor device to respond with a sample. In the case of the alarm interface, a 1 will arm the alarm and a 0 will disarm the alarm, in either case the device will respond with the current alarm status.

CRC FCS

The Cyclic Redundancy Check Frame Check Sequence has been calculated and stored in a lookup table on each module and is used for error detection. See section 1.2.4 for a theoretical explanation.

Parity bit

Odd parity is an error checking technique that appends one bit to the data word to make the total number of 1's odd⁶. The receiver recalculates the parity from the received data word and if the parity bit corresponds to the transmitted parity bit, there were no single errors in the data word.

Upon reception of an error free packet, the module processes the instruction and responds to the master in confirmation of the successful completion of the instruction.

The complete instruction word set for the network of four primary slaves is listed in table 1.

Unit	Meaning	Word⁷
Master instructions sent to slaves		
Power switch	ON	01010-1-11
	OFF	01010-0-10
Light switch	ON	10101-1-00
	OFF	10101-0-11
Alarm Interface	Arm	11110-1-00
	Disarm	11110-0-11
Sensor	Request sample	10000-1-01
Slave responses to sent to master		
All slaves	Confirm execution	Echo of those listed above
Sensor units	Sensor ON	00111-1-00
	Sensor OFF	00111-0-11

Table 1.

The instruction word set for the four primary slaves and their responses.

2.1.2 Serial Interface

The interface with the PC uses the RS-232 standard and meets the EIA/TIA-232E specifications⁸. Only three of the available channels are used, being Transmit (pin 3),

⁶ For example, if the data character 'W' has to be transmitted, the encoder would add a 0 to make the total number of 1's odd. **W: 01010111 | 0**

⁷ The words are given with hyphens separating the different fields, as listed in figure 12.

⁸ The EIA/TIA-232E standard specifies that the driver must generate voltage levels of +5 to +15 volts for a logic low and -5 to -15 volts for a logic high, as listed by Horowitz and Hill [13]. The signals must be able to drive a load of 3 kΩ to 7 kΩ with a slew rate of at least 30 V/μs while being able to withstand a short to any other output (this can mean up to 500 mA at 5 volts). The receiver must present a 3 kΩ to 7 kΩ load of resistance to the transmitter.

Receive (pin 2) and Signal Ground (pin 5). The CMOS (Complementary Metal Oxide Semiconductor) logic levels are converted to RS-232 levels using a Maxim MAX232E serial interface IC.

Serial communication signals are generated on the microcontroller which has a built-in UART, the code of which is listed in Appendix 3.

The instructions sent via the powerline to other modules are also driven by the same serial interface, although multiplexed to the powerline instead of the PC.

Experiment 1: Serial communications with the master unit.

An experiment was conducted to determine whether the PIC16F876 can communicate correctly with the computer. The microcontroller was programmed to receive a byte from its serial port, add 1 to it, then transmit it back to the PC through the multiplexer. If the byte received was an instruction for the slave modules, the microcontroller would identify this and multiplex the serial output to another computer that represents the powerline. The setup appears in figure 13.

A simple modem terminal program was used to communicate via the serial port, set to 1200 bits/s, odd parity, 1 start bit, 1 stop bit and with no handshaking. The ASCII equivalent instruction codes were typed by holding down the 'Alt' key while typing the corresponding ASCII code of the instruction.

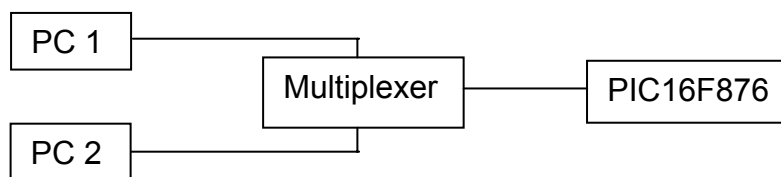


Figure 13.

Experimental setup to test the communications. PC2 represents the powerline network.

The results proved satisfactory, all unrecognised bytes were incremented and sent back to PC1, all those recognised in table 1 were forwarded to PC2, which represents the powerline network medium.

2.2 The Powerline Network

At first it was considered to use FSK (Frequency Shift Keying) modulation to transmit the data signals across the network. The design used is given in figure 14.

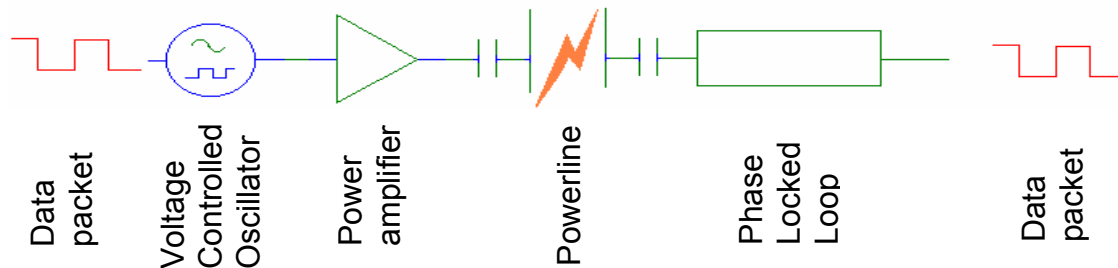


Figure 14.
Data transmission using FSK.

Experiment 2: To test the technical feasibility of FSK modulation.

Firstly 0 – 5 Volt digital signals were fed to a Voltage Controlled Oscillator (VCO), the Motorola 4066A. The two FSK carrier frequencies were chosen at $F_1=130$ kHz and $F_2=140$ kHz. The resulting FSK modulated signal was power amplified using a Class B amplifier and coupled onto the powerline. The signal was then demodulated using a Phase Locked Loop (also the Motorola 4066A).

The results of the experiment are given in figure 15. Although the resulting waveform was sufficiently unambiguous to be used for data input, it was found that the efficiency of the VCO/PLL method of modulation was highly dependant on component variations and other largely uncontrollable factors. A slight accidental nudge of the trim potentiometer or a 0.2 V power supply variation would change the carrier frequencies drastically. This method of modulation was abandoned and it was decided to use ASK modulation with an off-the-shelf integrated circuit, the TDA5051A.

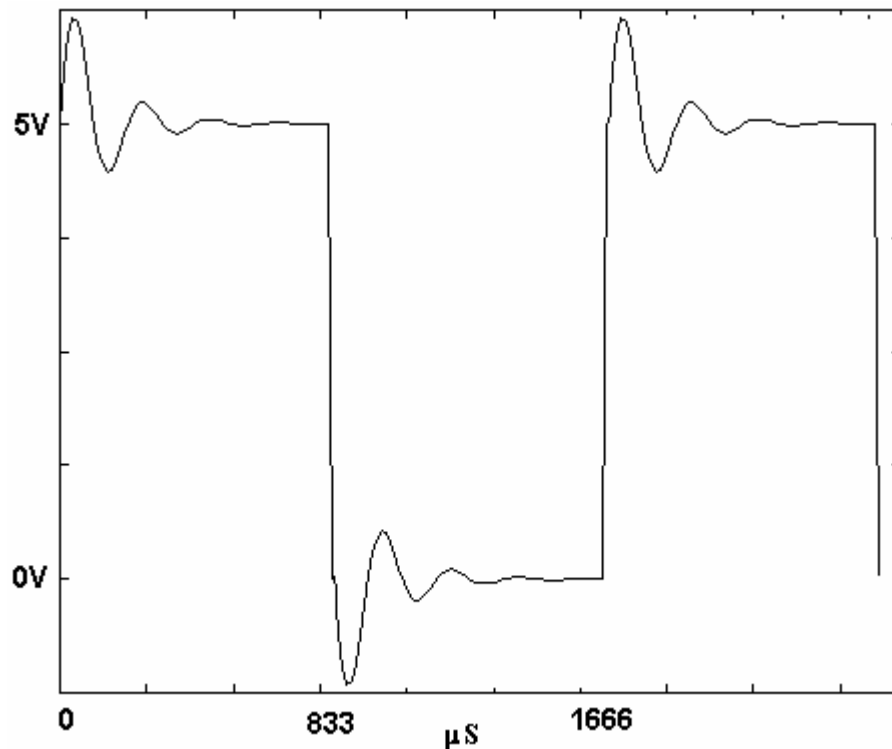


Figure 15.
The response received by using FSK modulation with a phase locked loop. The input signal was a 1200 bits/s square wave.

The interface with the building's powerline lies at the heart of the network and is crucial to viability of the system. Once the instructions have been encoded in the microcontroller, they are sent to the ASK modulator. A Philips HEF4066BCC quad bilateral switch is used to switch the serial bit stream from the master module to the PC serial port or the ASK modulator, as shown in figure 16.

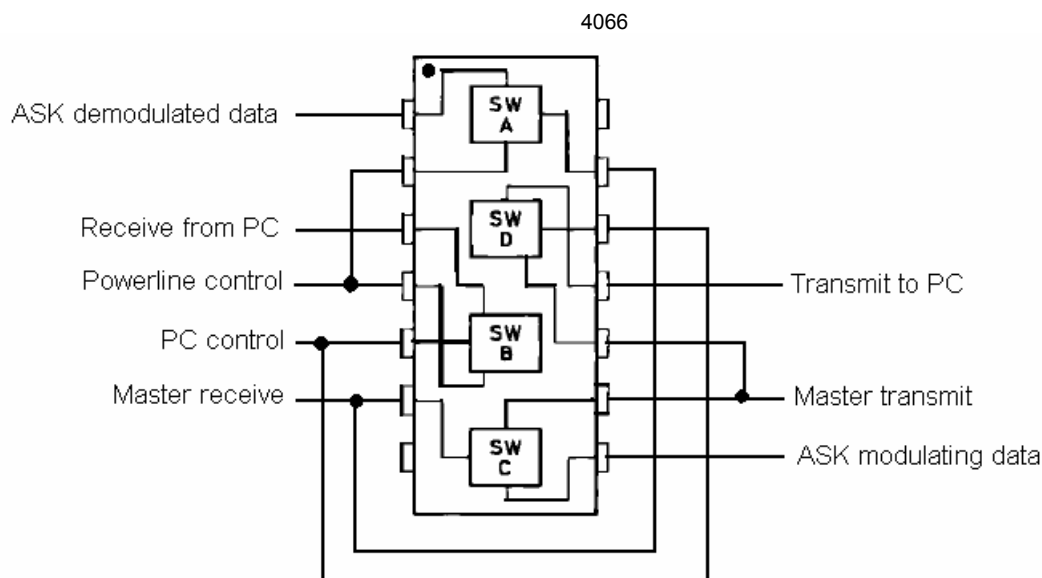


Figure 16.
The wiring diagram to implement a multiplexer/demultiplexer with a 4066 quad switch.

The Philips TDA5051 modulator/demodulator is used to convert the binary data into bursts of 125 kHz sine wave carrier. The TDA5051 also contains a power drive

module, which can drive the powerline, which typically has an impedance of between 10Ω and 100Ω at the carrier frequency as measured by Chan [3].

The line filter

It was necessary to build a filter that would pass the carrier signal, but will remove the 50 Hz, 240 V power signal. The 4th order design suggested by Philips [10] was tested and found adequate. The topology shown in figure 4 on page 7 was chosen, the values changed to the nearest standard available values and simulated in Microsim PSpice.

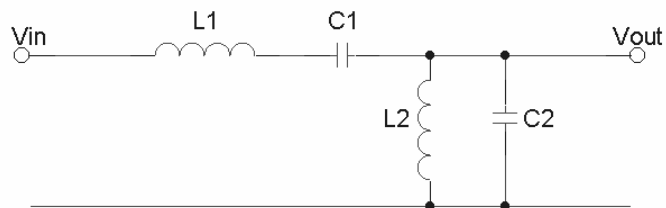


Figure 17.
Topology of the LC filter.

The simulation results are shown in figure 18 and the measured response can be found in section 3.

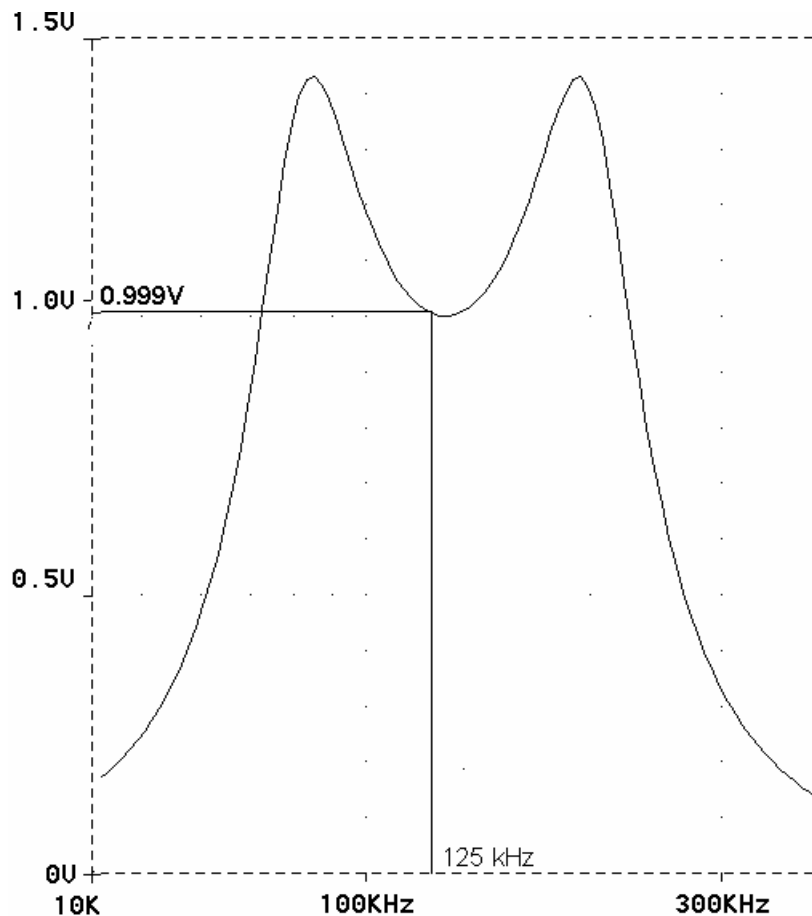


Figure 17.
Simulation data using $C_1, C_2=33\text{nF}$ and $L_1, L_2=47\mu\text{H}$.

Simulation data reveals that the 50 Hz 240 V signal is reduced by 69 dB, corresponding to 37.23 μ V. The 125 kHz signal is passed with attenuation of only 0.3 dB.

The output and input pins of the modulator are protected by a P6KEV68 unipolar transient suppressor against over voltage and negative transients. The circuit is fused for protection up to 630 mA and coupled to the powerline with the filter. A 1 μ F capacitor is placed in series with the transmit pin and the filter, to remove the DC component present on the pin, before filtering. The complete hardware diagram can be found in Appendix 3.

Experiment 3: To test the feasibility of using the powerline as communications medium.

This experiment was conducted to see if digital signals can be successfully sent across the powerline.

The experimental setup is shown in figure 18.

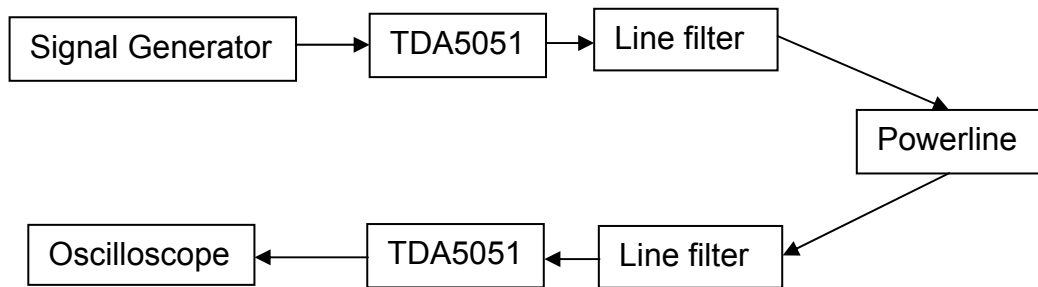


Figure 18.
The experimental setup to test the transmission of digital signals across the powerline.

The signal generator was set to create 0 – 5 V square wave signals with a duty cycle of 50% and a pulse width of 833.3 μ s, corresponding to a bit length at 1200 bits/s. This was fed to the modulator and placed onto the powerline with the filter. After demodulation the signal was viewed on an oscilloscope together with the source signal.

A time delay of 32 μ s (3.6%) was measured, the output pulse shape was square and there was no ambiguity in logic levels. Under the relatively noise free conditions in the office where the experiment was conducted, not a single observable bit error occurred. When the experiment was moved to the University laboratory, a truly harsh environment indeed considering all the electronic equipment plugged into the powerline, the resulting signal was still highly reliable, with very few errors observed.

2.3 Slave Units

Each slave unit consists of a Microchip PIC16F84 microcontroller, a line filter, a TDA5051 ASK modulator/demodulator and a special function unit, as depicted in figure 19.

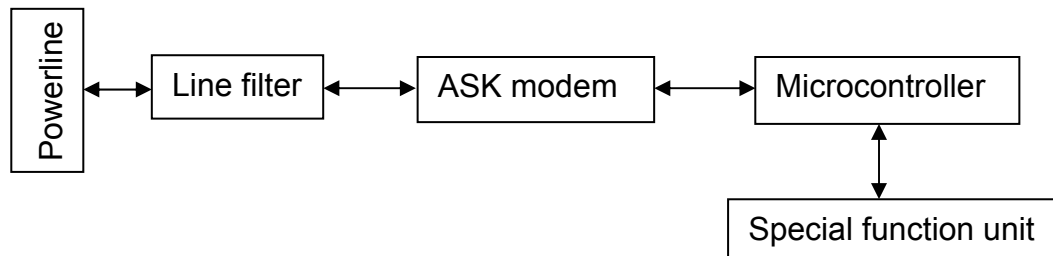


Figure 19.
Flow diagram of data in the slave units.

2.3.1 Protocol software

The network protocol has been implemented in software on the PIC16F84 and the entire source code listing can be found in Appendix 3.

Receive

The microcontroller is programmed to wait for the arrival of a start bit, which is a logic 0. As soon as the logic drop is detected, the microcontroller waits half a bit length (416.6 μ s at 1200 bits/s) and checks to see if the start bit is still present. If it is, it samples the receive pin every 833.3 μ s (one bit length) and shifts each new bit into a receive register. After the 9th sample, it checks for a stop bit (logic high), if present, the reception was successful and the instruction ID is checked against the local ID. If no stop bit is detected, a synchronisation error has occurred and the data is discarded, after which the microcontroller waits for a new instruction.

Transmit

The transmission works similar to the reception. The microcontroller initiates a transmission by sending a start bit (low) for 833 μ s, then each bit for the same length, and finally the stop bit (high). Ten copies of the packet are transmitted, spaced with 2 ms delays.

The timing has been implemented on the microcontrollers by using the PIC16F84's Timer 0 module, as the code snippet in figure 20 shows.

```
onebit    clrf      TMR0      ; Wait one 1200 baud bit period
          incf     TMR0      ; = 833.00us
          movlw   154
          movwf   TMR0
loop      movfw   TMR0
          btfss   STATUS,Z    ; Check for TMR0 overflow
          goto    loop
          nop      ; Fine tuning timing
          nop
```

```

        retlw      0

halfbit  clrf      TMR0          ; Wait half a 1200 baud bit
        incf      TMR0          ; = 417us
        movlw     206
        movwf     TMR0
loop2    movfw     TMR0
        btfss    STATUS,Z      ; Check for TMR0 overflow
        goto     loop2
        nop      ; Fine tuning timing
        nop
        retlw     0

```

Figure 20.
Two functions, *onebit* and *halfbit*, representing a delay of 1 bit length and ½ a bit length respectively.

The four different slave special function units are described below:

2.3.2 Alarm interface unit

This is the simplest of the units and simply acts as an interface to an existing alarm system. This unit receives commands from the master unit and provides an output port which a larger existing alarm system can read. If the master sends an ‘arm’ signal, the alarm interface microcontroller holds its output port at 5 volts until a ‘disarm’ signal is received, at which time it pulls the output port to ground.

If the alarm system supports it, the unit can read a 0 or 5 volt signal at its input port, a high implying that the alarm has been triggered and a low that the alarm is still armed. Each time the unit is polled by the master, it reports the level of its input port.

The alarm interface unit can also be used to interface the home automation system with other electronic devices, not just an alarm system. For example a garage door opener can be incorporated, provided it complies to the specifications listed above.

An LED (Light Emitting Diode) is provided to indicate the alarm status locally.

2.3.3 Power Switching

Both the light switch and the power switch work identically. The slave microcontroller triggers a Philips BT136 triac, which switches the current through the load, as shown in figure 21. The BT136 triac can switch currents up to 4 A, which translates into 960 Watt of power delivered into the load at 240 V.

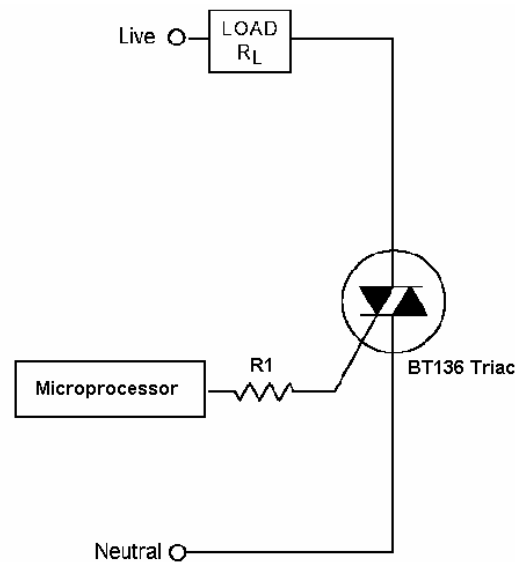


Figure 21.
The implementation of the current switch.

R_1 is calculated as follows:

$$V_{out} = V_{DD} - 0.7V$$

$$\therefore V_{out} = 4.3V$$

$$R_1 = \frac{V_{out} - 0.6V}{I_{R1}}, \text{ but to trigger, } I_{R1} = I_{TRIGGER} = 20mA$$

$$\therefore R_1 = \frac{4.3 - 0.6}{20 \cdot 10^{-3}} = 185\Omega$$

2.3.4 Light Sensor

The circuit diagram used to detect the ambient light level is given in figure 22.

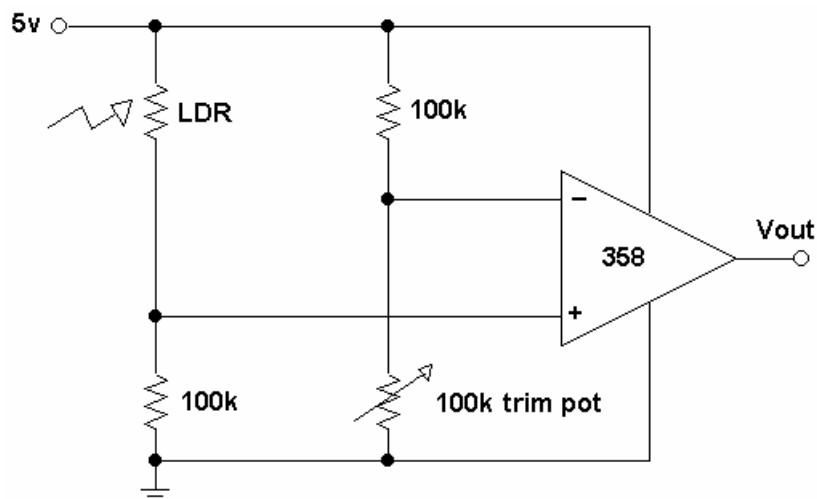


Figure 22.
The circuit diagram of the light sensor, using a LM358 opamp as comparator.

The operation of the light sensor is based on a voltage divider. The LDR (Light Dependant Resistor) changes its resistance in relation to the amount of light falling

on it. In absolute darkness the resistance is about $400\text{ k}\Omega$ and in sharp light it drops to around 400Ω . When placed in series with another resistor, between the two power rails, the voltage at the node between the resistors changes as the voltage division changes. This voltage is compared, using an opamp, to a reference voltage between another fixed resistor and a potentiometer. If the light level is sufficiently high such as to cause the LDR voltage to be higher than the reference, the opamp saturates to near the positive rail, if lower, the output drops to ground potential.

The potentiometer allows the threshold light level to be set by the user. Since the power supply to the opamp is 5 volts, the output produces CMOS compatible logic levels that can be read by the microcontroller.

The LM358 opamp was chosen because of its small outline (8 pin IC), its capability of saturating to within CMOS logic levels and because of its exceptionally high gain (100 dB). High gain is important since the outdoor ambient light level tends to change very slowly and when the light level is very near the threshold, ambiguous logic levels can result. The higher the gain of the opamp, the smaller the transition margin.

2.4 Computer Software

The user interacts with the home automation network by using client software. One of the most appealing features of this system is that the user can exercise full control over the system using an Internet connection. It is necessary for the user to have a permanent Internet connection to the building. The computer which communicates with the master unit must have web server software installed. If the user wants to access the network locally (not through the Internet), a local browser session can simply be opened on the computer attached to the master unit.

The client software takes the form of a web page downloaded from the server to the user's browser using HTTP. The programmability of the network is event-driven, in other words, the user specifies which type of event must occur which would coincide with the change of state of one or more of the slave units. The user fills in a form that specifies the type of event that must occur and which unit must change into a new state at the time of that event.

The events that can be used are:

1. Immediately,
2. At a chosen date and time,
3. When a specified sensor turns on,
4. When a specified sensor turns off.

A user can, for example, choose to have the outside light turn on when it gets dark outside and turn off again when daylight returns. This method of programmability provides flexibility not found in any other home automation system. The web page used to add an event to the schedule is shown in figure 22.

Powerhouse Scheduler

[View current schedule](#)

Add an event to the schedule:

Event	Unit	Action	Date/Time [ddmmyyhhmm]	Sensor Number
<input type="text" value="Now"/>	<input type="text" value="Plug"/>	<input checked="" type="radio"/> On <input type="radio"/> Off	<input type="text" value="1612001448"/>	<input type="text" value="0"/> [0=light]
Password: <input type="text"/>		<input type="button" value="Do It!"/>		
Thursday, October 26, 2000 2:48:38 PM				

Figure 22.
The web page used to program the schedule.

Each new event added to the schedule must be accompanied by a password, which can only be changed at the server, not through the Internet. It was considered to use SSL (Secure Socket Layer) to make the connection to the server absolutely secure, but this idea was abandoned due to the high cost of the license. The CGI program will only execute the instruction if the password is correct.

The web page source code is given in Appendix 3. It was created using a combination of Microsoft FrontPage Express 2.0 and Paul Lutu's Arachnophilia 3.0⁹. Once the form has been submitted by the user, a JavaScript program embedded in the HTML code preprocesses the form, concatenating the values of all the fields into one string. This string is then sent to the web server by calling the CGI program followed by the data in the form of parameters, as shown below, where 'paramstring' is the field data in one string.

```
http://www.cgidemo.co.za/cgiprogram.exe?paramstring
```

Once the web server has received this data, there are two methods for it to send the data to the CGI program which also resides on the server. The first is known as the 'POST' method, where the form data is sent to the CGI program by the web server using Standard Output. The CGI program simply reads its Standard Input to retrieve the field data, as can be seen in the C++ code snippet in figure 23.

```
main()
{
    char *endptr;
    double contentlength;
    char buff[100000];
    const char *len1 = getenv("CONTENT_LENGTH");
    contentlength=strtol(len1, &endptr, 10);
    fread(buff, contentlength, 1, stdin);
}
```

Figure 23.
A snippet from C++ code used to retrieve data sent using the POST method, adapted from Thomas [14].

The second method is called the GET method, used in this project, which is preferred due to its simplicity. Using this method, data is sent via an environment variable called "QUERY_STRING". Figure 24 illustrates how the CGI program retrieves it.

```
main()
{
    data=getenv("QUERY_STRING");
}
```

Figure 24.
A C++ code snippet used to retrieve data sent using the GET method.

Once the C++ written CGI program has been called and passed the event data sent to it by the user, it processes the data string and updates the schedule stored locally

⁹ Arachnophilia 3.0 and updated versions can be found at <http://www.arachnophilia.com/>

on the server. The CGI program then notifies the monitor program that runs continuously that the schedule has been updated, shown in figure 25.

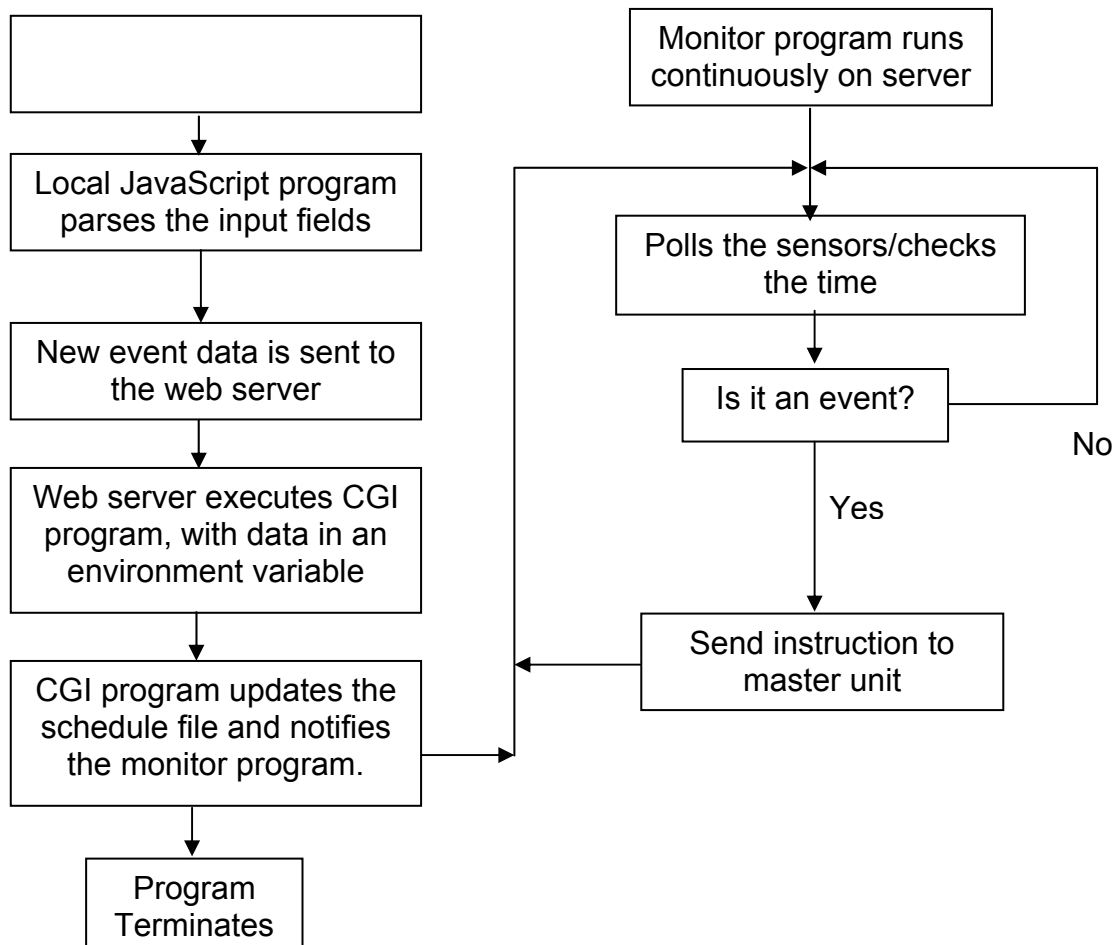


Figure 25.
A flow diagram of the process of adding a new event.

A real-time clock appears on the web page to help the user set the event time and date, as can be seen in figure 22. This clock was implemented in DHTML, taken from Evans [15].

The monitor program is the main control entity of the network. It times the polling of the sensors and waits for events to occur, then implements the necessary changes. The monitor program was written in Microsoft Visual C++ 6.0.

The schedule file ('settings.dat') is structured as follows:

A 12 byte header, containing the password and the amount of events stored in the file.

The information for each event is then sequentially stored in the file with the following information:

- The event number,
- The type of event,
- Which unit it affects,
- To which state that unit should change,

- The date and time of the event,
- If applicable, the sensor which is used.

The monitor program sends all its commands to the slave unit using RS-232, as described in section 2.1.2.

The user can also select “View current status” from the main page. This causes the master unit to poll each unit and check whether they are active on the network. A page is compiled and shown to the user which units are plugged into the network and what is their current state.

2.5 Power supplies

The power supply for each unit is identical. A 240 – 12 Volt transformer is used, which is then rectified by using a bridge rectifier, then finally regulated to 5 volt using a 78L05 voltage regulator. The circuit is shown in figure 26.

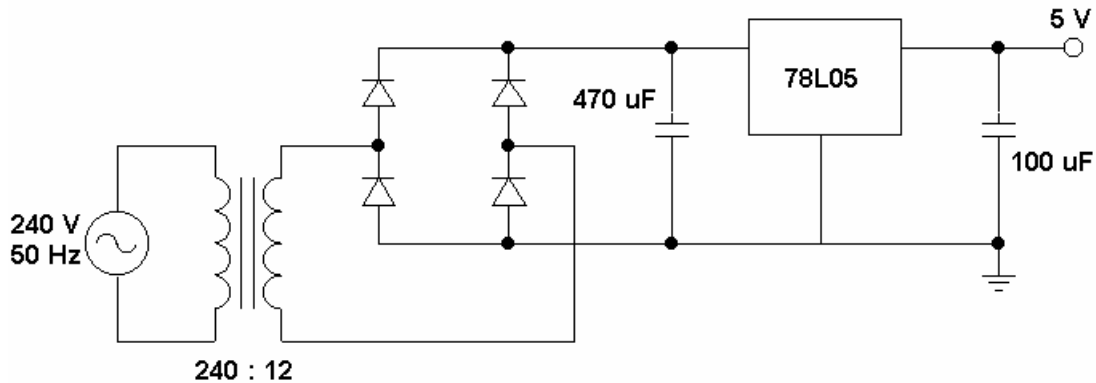


Figure 26.
The power supply circuit used for each unit.

The power supply was simulated using MicroSim PSpice. Figure 27 shows the simulation data when the power supply was giving 100 mA current to a 50Ω load. The supply current budget for the most heavily loaded unit, the power switching unit, is given in table 2.

Component	Maximum current drawn
TDA5051	68 mA
PIC16F84	4.5 mA
Triac trigger current	20 mA
LED	3.5 mA
TOTAL	96 mA

Table 2.
Maximum current drawn by the most heavily loaded unit, the power switch.

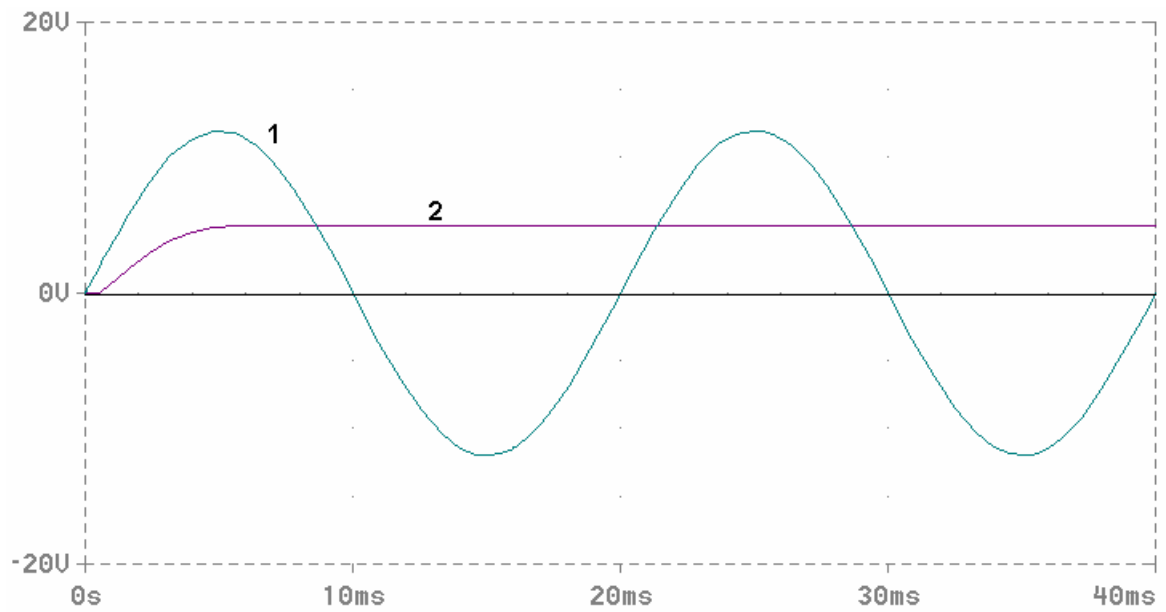


Figure 27. The regulated output (2) with the transformer output (1) plot. The regulated output maintained 4.98 V under a 50Ω, 100 mA load.

3. Results

The measurable results listed here are:

- 3.1 The powerline environment
- 3.2 Modulation
- 3.3 Electronics:
 - 3.3.1 Line filter
 - 3.3.2 Unit electronics
 - 3.3.3 Functionality

3.1 The powerline environment

The basic measured attributes of the powerline are given in table 3.

Attribute	Office Building ¹⁰	Laboratory ¹¹
V _{RMS}	242.34 V	241.52 V
V _{DC OFFSET}	0.556 mV	0.602 mV
Frequency	49.98 Hz	49.98 Hz

Table 3.
Measured attributes of the powerline.

The waveform present on the powerline, with the network inactive¹², is presented in figure 28. This waveform shows a the noise present on the powerline, including the 50 Hz harmonic noise occurring at the zero-crossings.

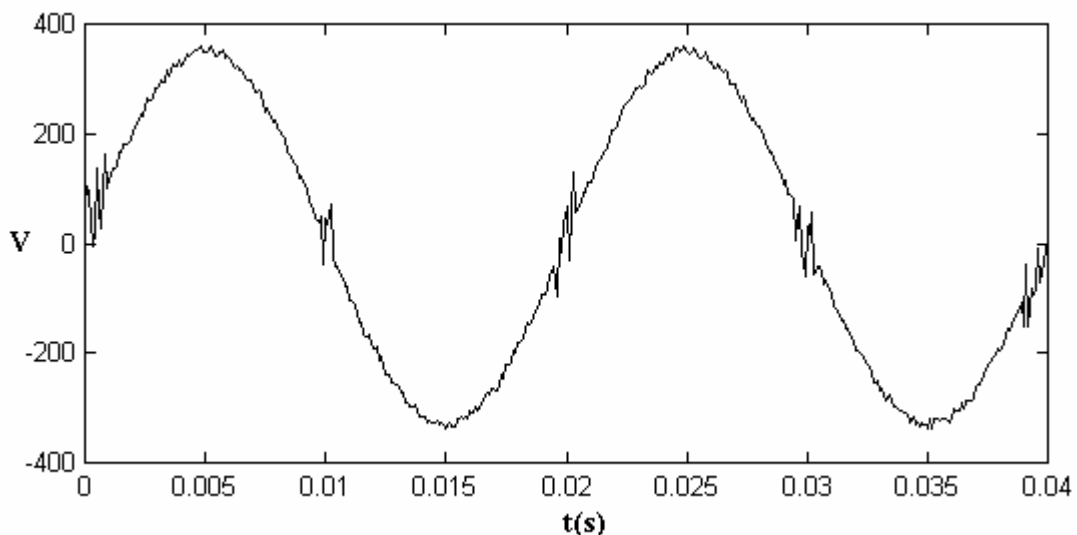


Figure 28.
The 50 Hz, 240 V powerline waveform as measured in an office building.

¹⁰ These measurements took place in the CFIM (Carl and Emily Fuchs Institute for Microelectronics) building at the University of Pretoria.

¹¹ Although geographically close to the office building, the laboratory is a completely different electric environment, due to the heavier load and more noisy conditions present on it.

¹² In fact, with the network active on the powerline, no perceptible difference in the waveform resulted. This is because the ASK carrier is only 1.1 V large, where the power carrier is 240 V large, resulting in only 0.45% of the total waveform.

3.2 Modulation

The input and output waveforms of the TDA5051A ASK modulator/demodulator are given in figure 29. A signal generator and a two channel oscilloscope were used to measure the waveforms.

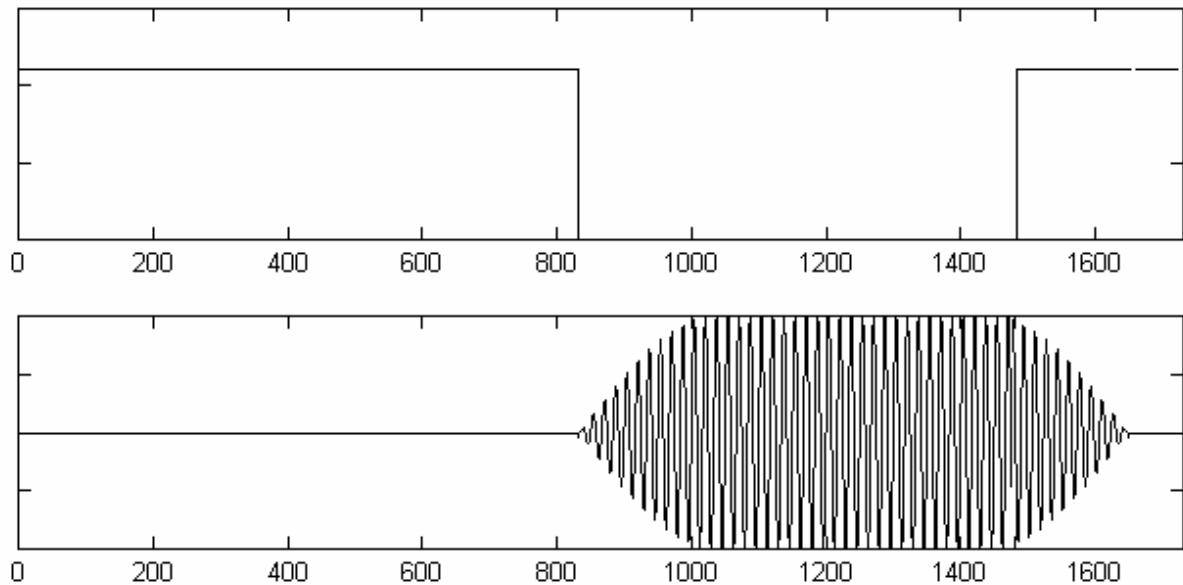


Figure 29.
The input and output of the modulator. Time divisions are in μs , at 1200 bits/s.

The time from receiving a low input to the full carrier shape appearing on the output, in other words the modulation delay from a logic high to low input, was measured as $183 \mu\text{s}$.

The time after input goes high to the complete absence of the carrier, the modulation delay from a logic low to high input, was measured as $180 \mu\text{s}$.

The time delay between data input, modulation, demodulation and data output was measured as $411 \mu\text{s}$, which corresponds to 48.01% of a single bit width, as shown in figure 30.

System attribute	Measured value
Modulation delay: High to low	$183 \mu\text{s}$
Modulation delay: Low to high	$180 \mu\text{s}$
Demodulation delay	$210 \mu\text{s}$
Total system delay	$411 \mu\text{s}$ (48.01% of bit width)

Table 4.
Attributes of the modulation and demodulation system.

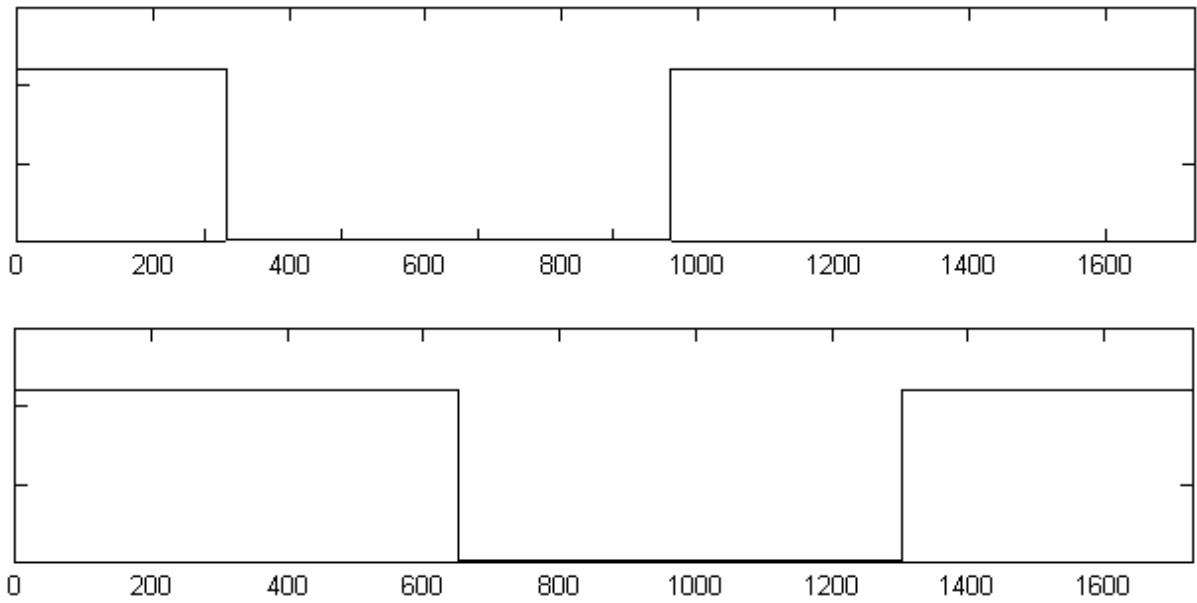


Figure 30.
A graphical indication of the time delay of the input to the output of a modulator/demodulator.

By giving the modulator a constant low input, and measuring the constant carrier wave output on to a powerline load, the following attributes of the carrier were measured:

Attribute	Value
Frequency	125.001 kHz
Amplitude	1.102 V

Table 4.
Attribute of the carrier wave.

3.3 Electronics

3.3.1 The line filter

The response of the line filter was measured using by using a sine wave signal generator, the output of which is passed through the filter, feeding a 30Ω resistor. The voltage across the resistor was then measured.

Frequency	Filter gain
125 kHz	-0.0872 dB (0.96 V)
100 kHz	-5.3 dB
150 kHz	-6.8 dB
1 kHz	-54 dB
50 Hz	-96 dB (3.803 mV)

Table 5.

The response of the line filter.

When a 1.1 V 125 kHz carrier was filtered, 0.96 V passed through. When a 240 V 50 Hz signal was filtered, only 3.803 mV passed through.

3.3.2 Unit electronics

The power supply currents of the different units were measured and are listed below. Current was measured by using a multimeter at the output of the voltage regulators.

Unit	State	Current drawn
Master	Not transmitting	12 mA
	Transmitting	82 mA
Light switch slave	Light off	12 mA
	Light on	28 mA
Power switch slave	No load	13 mA
	4 A load	36 mA
Light sensor	Light present	12 mA
	Light absent	8 mA

Table 5.

Current usage of the units.

3.3.3 Functionality

The functionality, the cardinal test of the home automation system, was tested by setting up the system as follows shown in figure 30. The units were spread around a house, each plugged into the powerline.

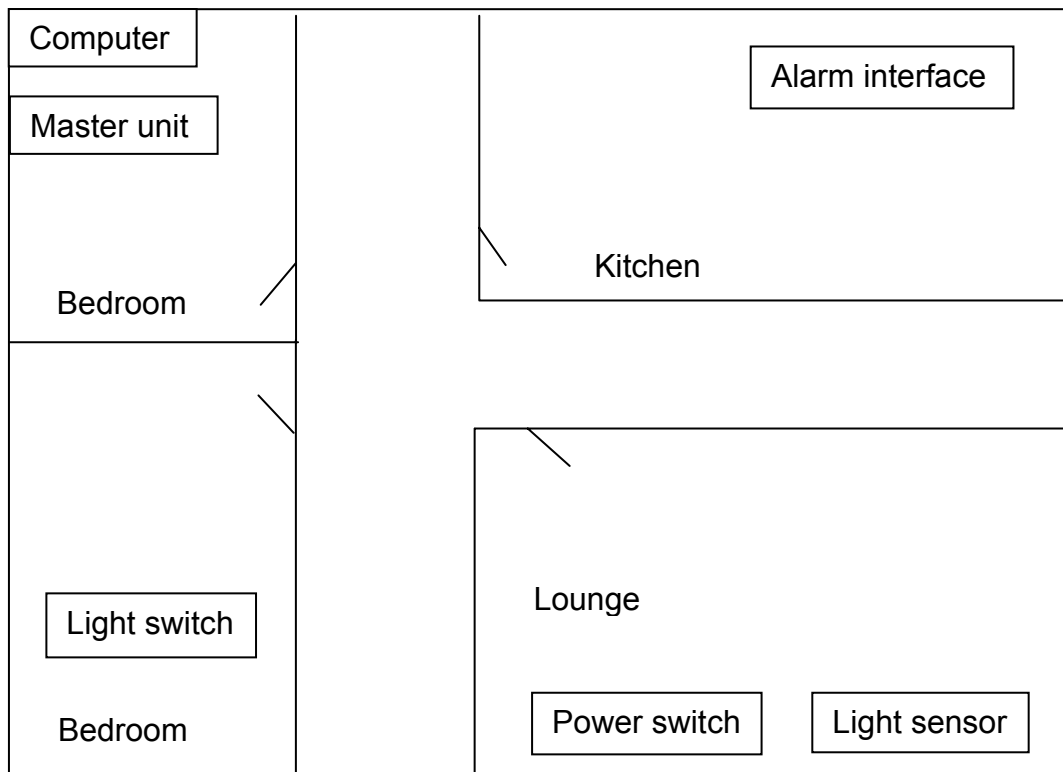


Figure 30.
A simplified schematic of the setup used to test the functionality of the system.

Firstly, the system was programmed to turn on the light switch immediately, and the submit button was pressed. After a delay of about $\frac{1}{2}$ a second, the light turned on.

Secondly, the system was programmed to turn the light on when the light sensor turns off, and vice versa. As there was sufficient ambient light to cause the sensor to be on, the light turned off immediately. When a cloth was placed over the light sensor to dim the light, the light turned on after 2 seconds.

4. Discussion

The results of each of the results in section 3 will be discussed below.

- The powerline environment
- Modulation
- Line filter
- Unit electronics
- Functionality

Powerline environment

Line voltage

The voltage supplied is expected to vary over different parts of the campus. This is because each building has its own spectrum of loads to deliver power to and a very large load tends to cause the voltage to drop. The results imply that the office building carries less of a load than the laboratory.

DC Offset

The DC offset is of a very small nature. The coupling network used in the powerline interface filter contains a capacitor, which decouples the offset anyway.

Power frequency

The frequency of the power carrier, rated at 50 Hz, is also expected to vary. During peak usage hours in the day, the power load is very heavy and the frequency tends to drop slightly as the power generators strain to deliver more power. It is also expected that the frequency will be the same for all locations fed by the same power generator. The frequency is determined at the generator and there are no large non-linear distortion effects on a small geographical scale, for instance on the same campus, that would modify this frequency.

Figure 28 shows large spikes on the 50 Hz 240 V power carrier at those time intervals where it crosses 0 Volts. This is due to the fact that triacs and capacitor banks are designed to switch on and off at the zero-crossing. The switching of many units simultaneously causes spikes to appear. Since the power carrier is a 50 Hz sine wave, and zero-crossings occur twice every full cycle, 100 Hz harmonic noise occurs on the line (every 0.01 second).

Modulation

FSK modulation

A large portion of the system development time was spent on designing and fine tuning the FSK modulator and demodulator. It was decided that it is not necessary to design a completely new modulator when one is already available off-the-shelf, since the focus of this project is more on the design of an entire network than on the design of its physical layer.

ASK modulation

The delay from the time that the digital input goes low to where the full amplitude carrier appears at the output is due to the power amplifier in the modulator that must start up. The total system delay from the input going low to where the output of

demodulator goes low, is ineffectual in terms of the operation of the system. In fact, it could have been many milliseconds and no observable degradation of performance would have resulted. The delay only causes the receiving unit to respond later and this is merely a matter of human comfort.

Line filter

The two frequencies that are of interest are the ASK signal carrier and the powerline carrier. The results show that almost all of the signal carrier is passed onto the powerline by the filter and that the powerline carrier is almost completely filtered out. The tiny amplitude that does pass back into the circuit is so small that it doesn't affect the analog electronics or digital decision levels.

Unit electronics

The current usage of the units is important in terms of the ratings of the transformers and the voltage regulators, but as far as the user is concerned, the extra energy that is charged for on the electric bill.

If all the units run at full load, maximum current drawn, the total current will be around 250 mA. At 5 V potential this translates into 1250 mW or 1.25 Watt. Eskom charges 26 c/kWH of electricity.

The system electric usage for a month:

$31 \text{ days} \times 24 \text{ hours} = 744 \text{ hours in a month}$

Units use : 1.25 Watt

$= 0.00125 \text{ kW}$

$0.00125 \text{ kW} \times 744 \text{ hours}$

$= 0.93 \text{ kWH}$

Total cost per month of system :

$0.93 \text{ kWH} \times 26\text{c/kWH}$

$= 24.6\text{c}$

Therefore the system only costs about 25c per month, which is very low indeed. This only includes the network and control electronics and excludes the power switched with the network.

Functionality

During the test the user entered a command via the web browser, the command was added to the schedule, the light sensor was polled for status, detected a change in light level and notified the master unit, the master unit notified the monitor program, which detected an event and sent a command to the light unit, the light unit received the command and turned on the light, then successfully reported correctly turning on to the master unit, which in turn reported to the monitor, which lastly gave the user feedback of successful execution.

The above procedure proves that the system works satisfactorily.

5. Conclusion

In the modern home or office filled with electric equipment, it can be very convenient to exercise control over all of these apparatus from a central point. To implement this solution a network was designed using almost all conceivable fields of electronics; analog signals, digital signals, microcontroller programming, power switching, power supplying, filter techniques, computer client/server programming, signal processing, error detection and asynchronous communications.

The powerline is a notoriously harsh environment in which to send communication signals and an error detecting scheme had to be designed. The combination of repetition, slow data rate and error detection encoding overcame the erratically noisy conditions on the powerline and communication proved successful.

6. References

- [1] O'Neal, J. B., "The Residential Power Circuit as a Communications Medium", IEEE Transactions on Consumer Electronics, Vol. CE-32, No. 3, August 1986, pp. 567-576.
- [2] McArthur, N., Wingfield, J., Witten, I. H., "The Intelligent Plug", Wireless World, December, pp. 46-51.
- [3] Chan, M. H. L., Donaldson, R. W., "Attenuation of Communication Signals on Residential and Commercial Intrabuilding Power-Distribution Circuits", IEEE Transactions on Electromagnetic Compatibility, Col. EMC-28, No. 4, November 1986, pp. 220-226.
- [4] Vines, R. M., Trussel H. J., "Noise on Residential Power Distribution Circuits", IEEE Transactions on Electromagnetic Compatibility, Col. EMC-26, No. 4, November 1984, pp. 161-175.
- [5] Smith, J., "Coupling Networks", LonWorks PLT-10 Transceiver Datasheet, pp. 3-1 – 3-5.
- [6] Leonard, M., "Networked Controllers Talk over Power Lines", Electronic Design, September 17, 1992, pp. 73–80.
- [7] Kingsley, P., "Telecommunication Techniques", Prentice and Hall, 1990, pp. 172
- [8] Stallings, W., "Data and Computer Communications", Prentice and Hall, 2000, pp. 163.
- [9] Proakis, J., Salehi, M., "Communication Systems Engineering", Prentice and Hall, 1994, pp. 729.
- [10] TDA5051 Home Automation Modem datasheet, September 1997, Philips Semiconductors, Document order number: 9397 750 02513, pp. 21.
- [11] Stallings, W., "Data and Computer Communications", Prentice and Hall, 2000, pp. 203.
- [12] Teccor Electronics Inc., "Thyristors used as static AC switches and relays", AN1007, 1995, pp. 20-1 – 20-6.
- [13] Horowitz, P., Hill, W., "The art of electronics", Cambridge, 1995, pp.723.
- [14] Thomas, S., "Programming CGI in C/C++", Que, 1997, pp.103.
- [15] Evans, T., "HTML 4: 10 minute guide", Que, Third Edition, 1997, pp.143.

Powerline Home Automation System

Project Proposal

G.A. Richter

9719216

Submitted as partial fulfilment of the requirements of Project EPR400
in the Department of Electrical, Electronic and Computer Engineering

March 2000

Study leader: Dr. J.E.W. Holm

Approved:

Dr. J.E.W. Holm (Study leader)

DATE

Approved:

J.J. Hanekom (Project Coordinator)

DATE

Problem Statement

In the modern home filled with electronic apparatus and appliances, it is useful for the owner to exercise some form of centralised control over the functions in the house. Currently, when the owner needs to turn on the driveway and living room lights before arming the alarm and going to bed, he has to walk to the garage, then to the front door, then to the alarm box, then to bed.

When the owner leaves his house to go on holiday, he forfeits all control over the electronics in the house while he is away and cannot tell whether someone has breached the security or whether he had left the bedroom light on.

There is a need for a low-cost, reliable and interactive system that exercises full control over the electric and electronic aspects of the house, with the ability to be accessed from across the globe.

User requirement statement

It is necessary to develop a system that seamlessly integrates all electric apparatus in the home or office. A centralised command box is needed and must be able to send appropriate control commands via the powerline wiring to the entire variety of devices attached to the network. Each device, whether it is a kettle or an alarm system, must be able to identify calls sent to it and change its state accordingly. Some devices must be able to report back to the central box. The system must be fully interactive via a secure Internet connection to the control box. The transmission of signals must be reliable under noisy conditions.

Functional analysis

Inputs to the system:

- User command input via an Internet connection. The owner can also give commands from home via his own PC (personal computer) with a local session,
- Sensors on an interactive apparatus, e.g. a light sensor.

Outputs of the system:

- User feedback via Internet connection,
- Interface with existing security system,
- Power switching to control appliances.

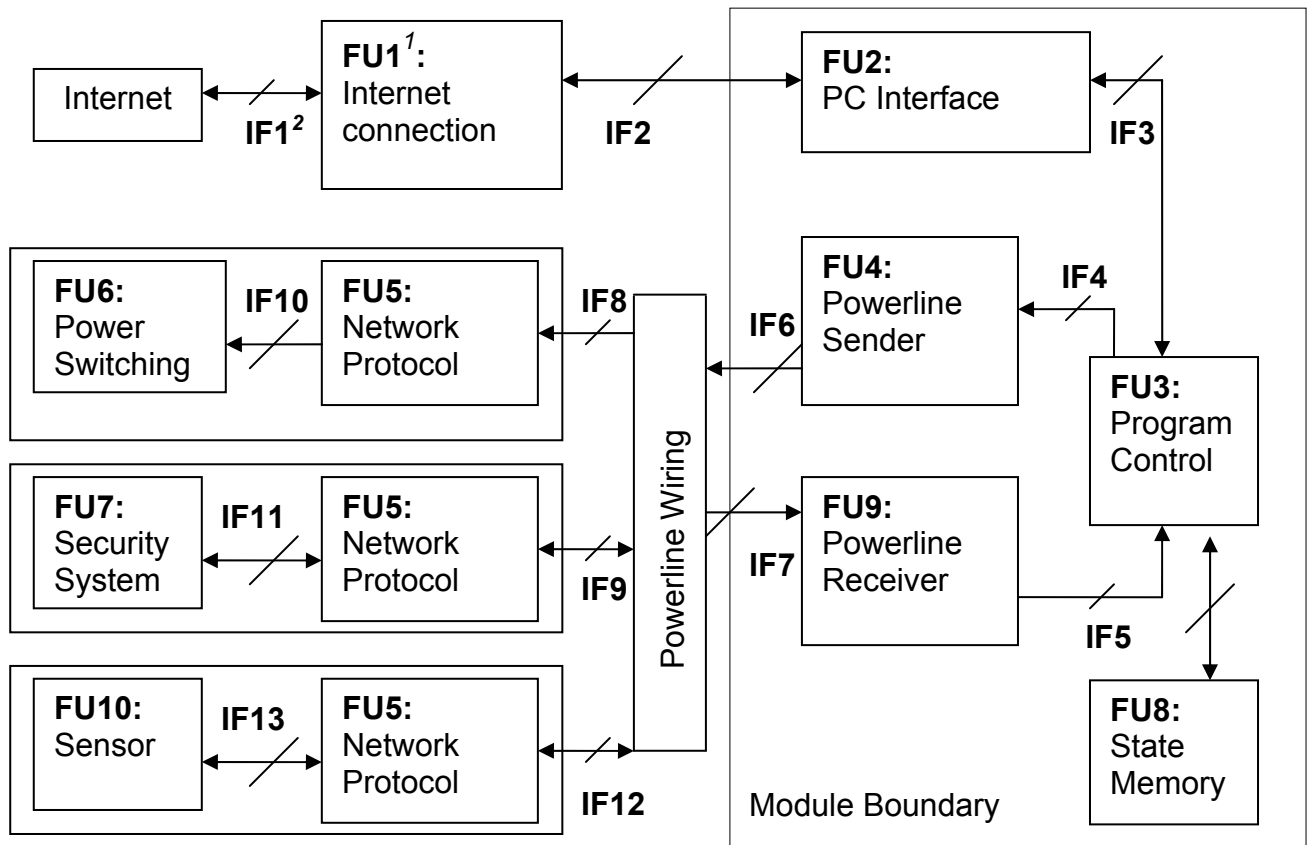


Figure 1: Functional block diagram of the system

System Specification

The system will be able to read commands from the user entered via the Internet and captured to the control box via a serial connection to the PC COM port. These commands will specify whether certain lights or appliances in the house must be turned on or off and at what times these must occur. Commands to the devices will be sent via the 240V AC (alternating current) powerline circuitry in the house and received by each appliance controller.

Light bulbs will be controlled by a unit that will plug directly into the light bulb socket and will allow a light bulb to be plugged into it. Other units will plug into a wall socket and will provide a 3-point socket for any appliance to be switched by plugging the appliance into the unit.

Interactive units will provide feedback from sensors attached to them for example a light sensor to indicate night or day.

¹³ FU means Functional Unit

¹⁴ IF means Interface

System Specifications:

- The user will exercise full control via a web site on the Internet with a 56kbit/s modem or at the local PC (IF1),
- The control box will communicate with the PC at 9600 bits/s via RS232 (IF2),
- The system program can continuously control up to 32 units (FU3),
- States of up to 32 units can be stored and queried via the Internet (FU8),
- All units will plug into 3-point sockets or light bulb sockets (IF8),
- The units will be compatible with a 240V AC powerline circuit (IF6,7,8,9),
- The system will be functional inside one single phase circuited house (IF6,7,8,9),
- Each power switching unit must be able to deliver 4 A current equivalent to 960W (FU6),
- Error compensation will be done, either via error detection and retransmission or by error minimization through repetitive transmission (FU5),
- Up to 32 units can be networked, each possessing a unique ID,
- A sensor (FU10) will provide a bipolar signal to the control box for example to indicate light or dark, or whether the telephone is ringing or not,
- The existing security system will be provided with a signal to provide an arm/disarm function and the control box will receive a signal from the alarm system to indicate whether the security has been breached.

Deliverables

At completion of the project, the following will be delivered:

- A central control box, connecting to a PC via a serial link and connecting to the powerline network via a 3-point plug,
- Web server software and source code to allow access to network via the Internet,
- A sample device connecting to powerline network via a 3-point plug and allowing the switching of an appliance via a 3-point socket,
- A sample light switching device that plugs into a light bulb socket and allows a light bulb to be plugged into it,
- A basic security system connecting to the network to illustrate the interface with a more complex existing security system,
- A sample sensory device that reports its status, for example a light sensor or telephone ring detector.

8. Appendix 2: Cost analysis

Item	Cost	Who paid
Printed Circuit Boards	R833.63	University
Master unit	R217.12	University
Slave units	R370.00	University
Cabling and connectors	R24	University
Package and casings	R120	University

9. Appendix 3: Technical Report

9.1 Software

9.2 Hardware

The circuit diagram for the ASK modulator:

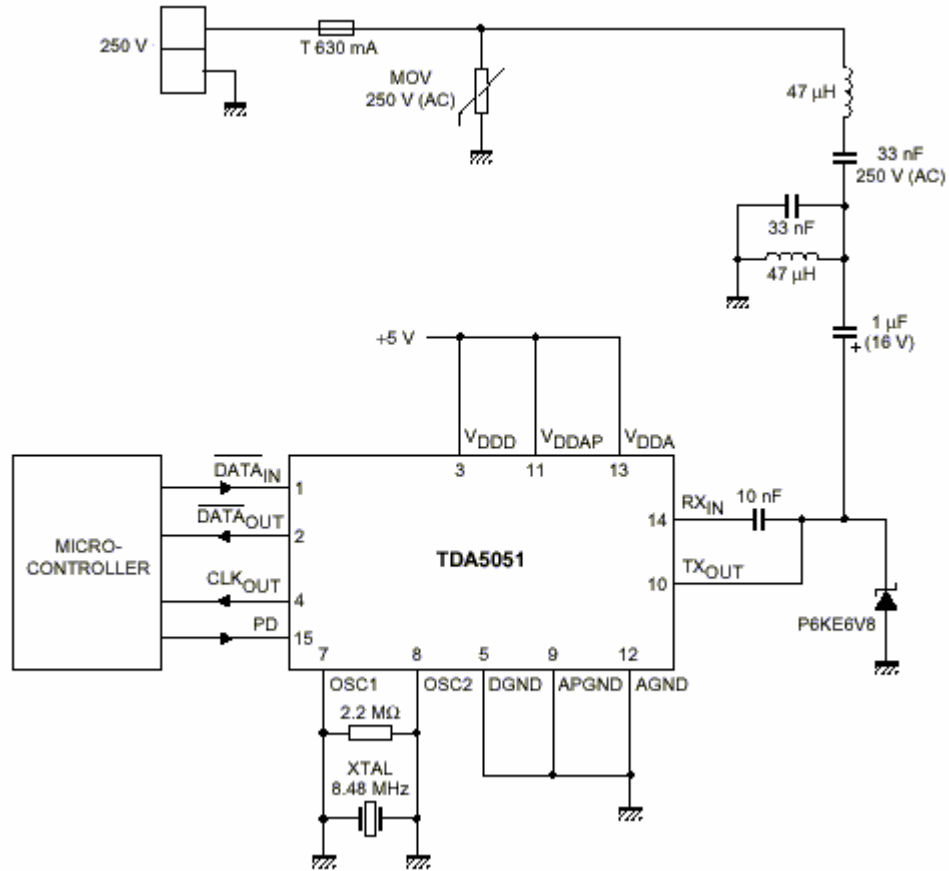


Figure 31.
Circuit diagram for the ASK modulator and line filter.

The printed circuit board layouts are presented here:

Master unit

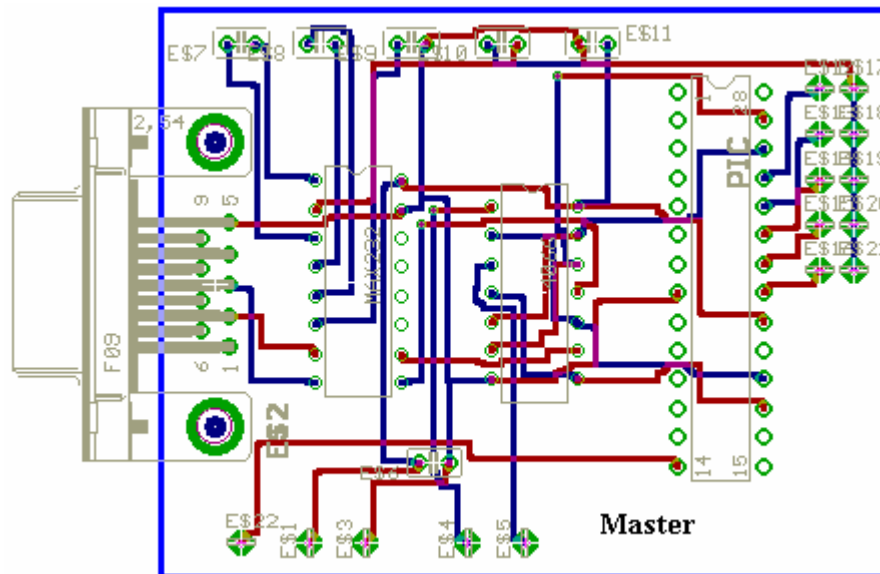


Figure 32.

Printed circuit board for the master unit

Master modulation unit

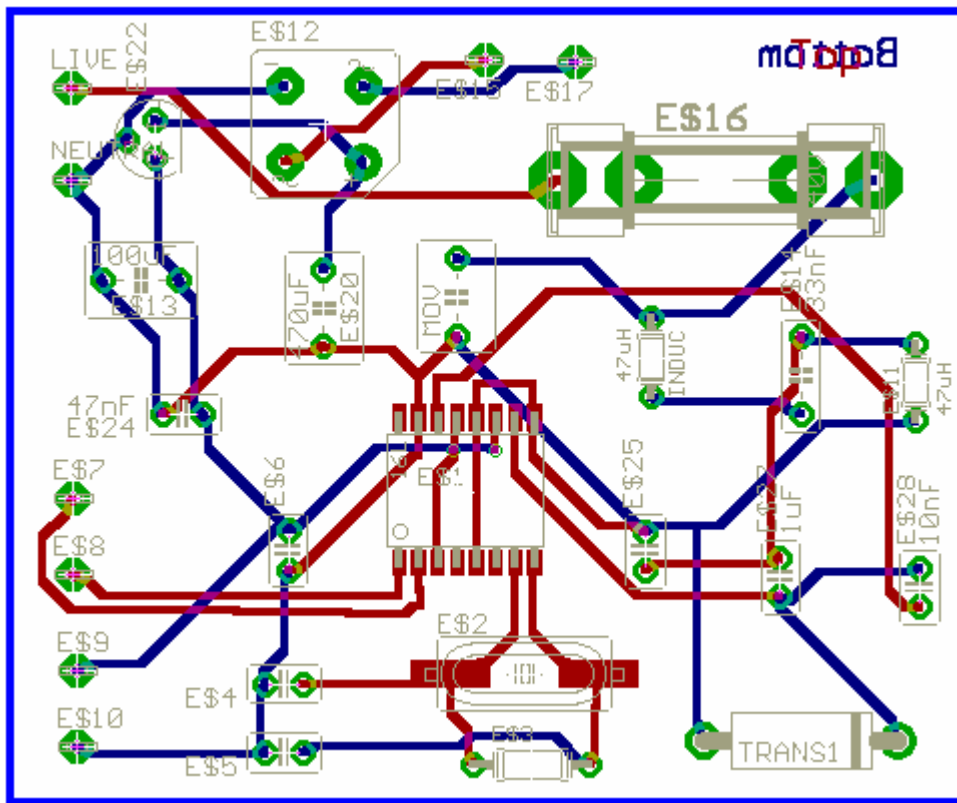


Figure 33.
Printed circuit board for the ASK modulation unit.

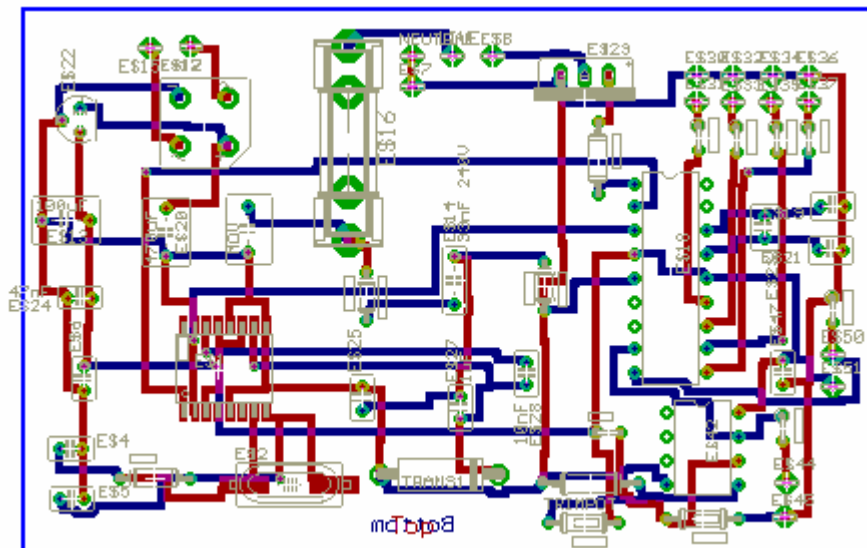


Figure 34.
Printed circuit board for the slave units.