University of Southern Queensland
Faculty of Engineering & Surveying

Telemetry and Communication Systems for Irrigation Management

A dissertation submitted by

Alison McCarthy

in fulfilment of the requirements of

ENG4111/2 Research Project

towards the degree of

Bachelor of Mechatronic Engineering

Submitted: November, 2006

Abstract

The development of wireless communication systems for agricultural and irrigation applications has enabled farm managers to access real-time data from in-field sensors to improve on-farm operations. Recent research into wireless sensor networks for agriculture has focussed on reducing the power consumption and manual configuration of the nodes by the end-user, while commercial wireless sensor networks use short-range transceivers such as Zigbee or long-range transceivers such as UHF citizen band radio. However, the research wireless sensor networks often experienced reduced reliability under field conditions. Therefore, this project aims to design a telemetry system with a focus on reliability under field conditions.

UHF citizen band radio was investigated as a candidate wireless transmission medium over which audio tones, modulated with serial data, were transmitted. The modulation technique used was frequency modulation, and was implemented using integrated circuit modems. A wireless network was designed with: a polling medium access method; star topology; error handling using repetition and cyclic redundancy codes; and robust protocol.

The developed telemetry system was tested under field conditions, such as in the presence of crop canopy and irrigation water, and near power lines which potentially cause noise in telemetry signals. Data delivery rates of 100% were achieved using 0.5 W radios transmitting at 1200 baud for 2 km, which satisfied all the telemetry system requirements. UHF citizen band radio is a reliable, long distance transmission medium for wireless sensor networks. The feasibility of a wireless sensor network using UHF citizen band radios and medium access methods including polling for on-farm use has

been demonstrated.

Further research on the use of UHF citizen band radios in wireless sensor networks may potentially allow higher data rates and rival data rate capacities of short-range transceiver networks.

University of Southern Queensland

Faculty of Engineering and Surveying

ENG4111/2 *Research Project*

Limitations of Use

Certification of Dissertation

I certify that the ideas, designs and experimental work, results, analyses and conclusions set out in this dissertation are entirely my own effort, except where otherwise indicated and acknowledged.

I further certify that the work is original and has not been previously submitted for assessment in any other course or institution, except where specifically stated.

Alison McCarthy

0050009145

Signature

Date

Acknowledgments

My family for their support throughout my project and program.

My project supervisors, Mark Phythian, Professor John Billingsley and Erik Schmidt for their assistance and guidance.

Nigel Hopson and Lindsay Evans in Jondaryan and Ray Griffith in Dalby for providing field trial sites.

Alison McCarthy

University of Southern Queensland

November 2006

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The development of sensor devices equipped with wireless network interfaces has seen an increase in the application areas of precision agriculture (Wang, Zhang & Wang 2006). A wireless sensor network is a group of sensors or ‘nodes’ that are linked by a wireless medium to perform sensing tasks. Wireless sensor networks are used in agriculture to supply farm managers with environmental data such as soil moisture, water storage levels and irrigation status. In the case of real-time monitoring of furrow irrigations, access to real-time irrigation data such as by wireless communication may lead to timely irrigation cutoff and increase irrigation application efficiency by up to 30% (Raine & Dalton 2003).

Connections between the nodes of a wireless sensor network may be formed using wireless media such as satellite communications, computer wireless technology, personal communication services, and other wireless technologies such as infrared and acoustics (Gibilisco 1999). These methods are all potentially applicable to wireless networks in agriculture, with the chosen technology depending on the required transmission distance and amount of data to be transmitted through the network. An in-field telemetry system may require transmission distances across paddocks of several kilometres.

This project aims to develop a wireless network to transfer data from in-field agricultural sensors to a base station that logs the sensor data.

1.1 Broad Aims and Specific Objectives

The development of a wireless communication link between in-field surface irrigation advance rate sensors and a base station will enable close to real-time monitoring of in-field irrigation application and other on-farm operations.

The system is intended to be used for data transmission with a range of up to 2 km. The specific objectives of this project are:

1. Research sensors that measure surface irrigation advance rate and their methods of data transmission.
2. Critically evaluate existing on-farm sensor networks and wireless link technologies that may be employed on-farm, comparing performance characteristics such as data rate, transmission range and networking configurations.
3. Design a rugged system, with a usage life of 3 years, that transmits wireless data up to 2 km from in-field advance sensors to a base station, thus enabling close to real-time evaluation of farm operations.
4. Evaluate the performance of the designed system under field conditions such as the presence of crops and irrigation water, and consider the effect of interference from other wireless devices.
5. Design an interface that enables farm managers to monitor the status of sensors.
6. Investigate techniques to enable interfacing of the designed wireless network with other existing on-farm networks using a different wireless technology.
7. Investigate and implement techniques for wireless image transmission to monitor on-farm operations, as time permits.

Ease of installation, low on-going maintenance and wireless transmission reliability are other design considerations.

1.2 Overview of the Dissertation

This dissertation is organised as follows:

Chapter 1 discusses the research project aims.

Chapter 2 describes the background of the problem and existing agricultural telemetry systems.

Chapter 3 evaluates existing wireless technologies and systems that utilise wireless links such as in agricultural applications.

Chapter 4 develops strategies for wireless data transfer using UHF citizen band radios from in-field sensors and investigates the feasibility of these strategies for an agricultural telemetry system.

Chapter 5 discusses the modulation and demodulation of data over the citizen band radio network.

Chapter 6 discusses the design of the network architecture and protocols.

Chapter 7 details the development and construction of the sensor node with wireless communication technology.

Chapter 8 discusses the performance of the citizen band radio network in field trials.

Chapter 9 concludes the dissertation and suggests further work in the area of wireless data communication in agriculture.

Chapter 2

Background

2.1 Telemetry in Irrigation in Australia

Wireless sensor networks applied in agriculture have the ability to interconnect several hundred sensor nodes and supply farm managers with data to help them use water more efficiently, such as in furrow irrigation. The efficiency of surface irrigation application in the Australian cotton industry is typically 45-90% (Dalton 2000), but may be improved by up to 30% by changing irrigation management practices (Raine & Dalton 2003). However, collection of data necessary to optimise irrigation performance is labour intensive.

By integrating sensors with wireless technology, labour and costs associated with data collection and sensor maintenance is significantly reduced.

2.2 Furrow Irrigation

Remote monitoring of furrow irrigation events may potentially be achieved using wireless technology. Furrow irrigation is a type of surface irrigation that channels the flow of water along the primary direction of the field using furrows which are small, parallel channels, or corrugations. The crop is grown on the ridges between the furrows. Since surface irrigation uses gravity to transport water, the farm is levelled or has a slope of up to 0.5%. Water is pumped from the farm manager's reservoir to the head ditch of the field, and siphon tubes move this water to a group of furrows (see Figure 2.1).

Figure 2.1: Head ditch of a cotton crop

On a furrow irrigated farm, tailwater is the runoff from the lower end of the field, which is then returned and reused. The water flows from the top of the furrow to the end of the furrow within two days of the start of the irrigation event. The advance rate is the rate at which the front of the water stream flows down the furrows.

2.3 Irrimate^{TM} Surface Irrigation System

The Irrimate^{TM} in-field surface irrigation evaluation system is a commercial package developed by the National Centre for Engineering in Agriculture (NCEA). Sensors in the Irrimate^{TM} system measure the advance rate of irrigated furrows to improve water management in the area of largest water losses, in-field irrigation application (Raine, Purcell & Schmidt 2005).

The advance rate is measured using advance meters that are positioned at intervals along the length of the field. Each advance meter consists of a data logger, a micro-controller and eight advance sensors which consist of a pair of long, open-ended wires. The eight pairs of wire ends are positioned in adjacent furrows. When the front of the water stream passes the wire ends a short circuit is formed. The time of this event is recorded on the microcontroller. Irrimate^{TM} is the only commercially available system that measures advance rate for furrow irrigation.

Irrimate^{TM} equipment sensors include an inflow meter (see Figure 2.2), outflow meter, five water advance meters, and eight advance sensors for each advance meter (see Figure 2.3). The inflow meter records the flow rate of the head ditch, while the outflow meter measures the flow rate of the tailwater. This system requires advance rate data to be collected from each advance meter for a furrow irrigation event to be optimised.

Figure 2.2: The flow meters measure the inflow and outflow rates

Data Transfer From In-field Sensors

The existing Irrimate^{TM} system uses a PDA to download the recorded data from the advance, inflow and outflow meters. The data from the in-field loggers are transferred via an infrared link to the PDA (see Figure 2.4).

The requirement to enter the field to form the infrared link and collect data is a major constraint in the use of the Irrimate^{TM} system due to the inconvenience to the farm manager who must enter the field soon after watering. The NCEA has identified that the Irrimate^{TM} system can be enhanced by adding an in-field communications capability that wirelessly links the sensor data to a central location.

Figure 2.3: The advance sensors are triggered by water

Source: Purcell (2004)

Figure 2.4: Downloading information from advance meter to PDA

Source: Raine et al. (2005)

2.4 Other Advance Rate Sensors

Other advance rate measurement systems have been identified in the literature, in addition to Irrimate^{TM}. Technologies used include wires placed in the furrows, telemetry and machine vision. The former two are discussed in the following sections since they highlight the difference between wireless and 'wired' technologies.

2.4.1 Capacitance Sensors

Turnell, Deep & Freire (1997) developed capacitance sensors on 1 km lengths of twin-wire to measure advance rate. The twin-wire is placed along the length of the furrow to be monitored, with one end of the twin-wire at the edge of the field connected to a laptop that monitors the sensors. This method is undesirable compared to a wireless system due to labour involved in installation and maintenance of the system.

2.4.2 Two-point Measurement of Advance Rate

Latimer & Reddell (1990) describe a real-time automated system for surface irrigation which senses the advance of water at two known distances down the field. The calculated advance rate is used to determine infiltration characteristics of the field, and to predict the performance of the remainder of the irrigation event. A telemetry system was integrated with this automated system that uses both radio and infrared wireless technologies. The infrared system required clear line-of-sight to operate and was limited to transmission distances of less than 800 m. The radio frequency system was low power (100 mW), had a transmission range of less than 400 m, and used a VHF frequency band (166 kHz) which is licence-free in the United States of America.

2.4.3 Agricultural Sensors

Agricultural sensors that measure in-field and meteorological parameters (such as soil moisture and humidity) and that have integrated wireless capabilities are available commercially. Agricultural sensors may have low data rate requirements, with only an analogue sensor value recorded at several intervals (such as a humidity reading or the state of an advance rate sensor), as well as the appropriate preamble, postamble and control bytes. However, data heavy agricultural sensors such as cameras would require high data transmission rates.

Commercial agricultural sensors that utilise wireless technologies are available, such as Adcon Telemetry's weather, soil moisture and water level sensors as shown in Table 2.1. These sensors are compatible with Adcon Telemetry units that operate on 433 MHz band with power output from 10 mW to 2 W and transmission range from 1 to 20 km. Agrilink sensors are also integrated with the Adcon Telemetry Units (*Agrilink Communications* 2006).

Table 2.1: Wireless sensor networks developed for agriculture

Company	**Wireless Technology**	**Transmission Distance**	**Data Rate**	**Sensors**
Adcon Telemetry, Agrilink	Adcon Telemetry Units	1-20 km	64 kbps	Soil moisture, water level
NICTOR	Zigbee	1 km	250 kbps	Pollutants
GME Telemetry System	UHF transceivers	100 km	4800 bps	Fences, pump status

Chapter 3

Literature Review

A literature review revealed the existence of research prototype and commercial telemetry systems for agricultural applications. The technologies utilised in these wireless networks included radio frequencies such as Bluetooth and citizen band (CB) radio. Important characteristics of the wireless technology were identified as reliability, data rate, transmission distance and networking configurations.

In the last few years, research into wireless telemetry systems for agriculture has focussed on reducing the power consumption of the node and manual configuration by the end-user. Sensors in an agricultural telemetry system may be left in the field for up to three years, and minimising power consumption is an important consideration. However, some research papers identified problems with the reliability of wireless sensor networks under field conditions.

3.1 Wireless Technologies for On-farm Sensor Networks

Wireless technologies applicable for use with in-field sensors must be reliable under field conditions. Required characteristics of wireless sensor nodes are identified by Wang et al. (2006) and Akyilidiz, Su, Sankarasubramaniam & Cayirci (2002) to include the use of robust wireless technology, low cost, flexible input/output, long-lifetime energy source, efficient energy use, small footprint and to be autonomous and adaptive to the environment.

Wireless transmission media range from wireless computer technology such as Bluetooth (similar to technologies implemented in research prototypes), to cellular communications and personal communication services such as citizen band radio.

3.1.1 Short-range Transceivers

Short-range wireless computer data transceivers such as Zigbee and Bluetooth typically have a range of less than 100 m. Many of these transceivers operate on the Industrial, Scientific and Medical (ISM) bands which are licence-exempt frequencies. However, there are class licence restrictions on the use of these frequencies that are enforced by the Australian Communication and Media Authority. For example, the power output of short-range transceivers is restricted under the Low Interference Potential Devices Class Licence (*Australian radiofrequency spectrum allocations chart* 2006). Table 3.1 summarises the specifications of some short-range transceiver technical standards.

Table 3.1: Wireless technologies used in wireless sensor networks

Wireless technology	**Frequency band**	**Data rate**	**Nodes per master**	**Range**	**Battery life**
IEEE 802.11a	5.15-5.35 GHz 5.725-5.825 GHz	54 Mbps	52	100 m	Hours
WiFi (IEEE 802.11b)	2.4-2.4835 GHz	11 Mbps	32	100 m	Hours
IEEE 802.11g	2.4-2.4835 GHz	54 Mbps	52	100 m	Hours
Bluetooth (IEEE 802.15.1)	2.4-2.4835 GHz	1 Mbps	7	10 m	1 week
ZigBee (IEEE 802.15.4)	2.4 GHz 915 MHz 868 MHz	250 kbps 40 kbps 20 kbps	64 000	70 m	1 year

Source: Wang et al. (2006)

Equipment using the IEEE 802.11 standard is low cost due to its popularity and the licence-free frequencies it uses. Bluetooth has the ability to penetrate solid objects (Gralla 2002). Zigbee is becoming a wireless standard for remote control in the industrial field which addresses lower speed data transmissions and has lower power consumption than Bluetooth and IEEE 802.11. ZigBee modules can be networked using star, cluster tree or mesh topologies. Radio frequency identification (RFID) tags can operate at low frequency (less than 100 MHz), high frequency (greater than 100 MHz) and UHF (868-954 MHz). A Zigbee transceiver has a retail price of approximately $80.

Wireless technologies that operate on lower frequencies feature a longer transmission range and stronger capability to penetrate obstructions than higher frequencies (Wang et al. 2006). For example, 915 MHz frequency systems have approximately twice the range of 2.4 GHz frequency systems. Wireless standards that have transmission ranges of less than 1 km may be configured in a self-organising, self-configuring mesh network. Routing nodes would have to be placed in the field for the sensor data to hop from node to node to reach the base station and obtain a total range greater than 2 km.

Other short-range transceivers, eg. Keymark Wireless Receiver and Transmitter (available at Jaycar Electronics), operate at 433 MHz, have a range of about 30 m and cost about $10 each for a transmitter and receiver.

3.1.2 Citizen Band Radio

Citizen band radios are licence-free, half-duplex, voice communication devices (*Citizen band radio* 2006). Citizen band uses two frequency bands: 26.965-27.405 MHz in the high frequency (HF) band and 476.425-477.400 MHz in the ultra high frequency (UHF) band. Citizen band radios typically transmit at a range up to 10 km line of sight (see Figure 3.1).

Data for telemetry and telecommands can only be transmitted on UHF channels 22 (476.950 MHz) and 23 (476.975 MHz). Voice communications are prohibited on these UHF citizen band channels. GME Electrophone manufactures UHF citizen band radios that transmit data or voice (such as the TX3400 and TX3600T) based on inputs to the radio through an 8-way telephone socket. These radios cost approximately $429 each, which is not feasible for in-field nodes.

Restrictions on citizen band telemetry channels include a maximum output power of 5

Figure 3.1: UHF citizen band radio

W, transmission duty cycle of three seconds in an hour, and the fitting of a device to shut off the transmitter after three minutes of continuous operation (*Radiocommunications (Citizen Band Radio Stations) Class Licence* 2002).

UHF citizen band radio is the transmission medium of at least two commercial agricultural telemetry systems (CB Monitor and GME Electrophone). The highest data transmission rate offered by these commercial citizen band telemetry systems is 4800 baud (for the GME Telemetry System). Therefore, the timely transmission of data from sensors such as cameras may not be feasible with current technology (due to three second duty cycle restriction). However, for agricultural sensors that measure analogue values, a data rate of 4800 baud is sufficient. Citizen band radios are also inexpensive, with a retail price of as low as $20 each.

3.1.3 Amateur Radio

Amateur radio is similar to citizen band radio, but operates at various frequencies in the HF, VHF and UHF bands, and requires a licence to operate. Data is transmitted over amateur radio using packet radio, which is a well-established form of digital data transmission to construct wireless computer networks. A packet radio station consists of a computer and software, a modem or terminal node controller, and an FM transceiver and antenna. Images such as weather satellite data can be transmitted over amateur radio fax or slow scan television (SSTV).

Modems for amateur radio have been developed that modulate data by converting bits into tones and transmitting these tones over an amateur radio. The data is demodulated by a modem that receives tones over the amateur radio and converts these tones back into bits. The speaker and microphone of amateur radios are connected to the sound card of the computer and software encodes the data and decodes the tones. Hardware developed for amateur radio is compatible with citizen band radio.

3.1.4 Cellular Mobile Telecommunications

A cellular telecommunications system is a network of repeaters that allow mobile radio transceivers to be used as telephone sets. Video, voice and data can be transmitted at data rates between 384 kbps and 2 Mbps through cellular towers, with typical coverage of the whole country. However, several limitations still exist for rural areas as there may be insufficient cellular phone towers.

Wireless pagers use similar technology to cellular phones to send and receive numeric, text or voice signals. The WeatherTRAK irrigation control system uses paging technology to transmit weather data (*WeatherTRAK ET Irrigation Controller* 2006).

3.1.5 Infrared, Visible-light and Acoustic Devices

Technologies such as modulated infrared, visible light and laser communications require clear line of sight between transceivers, and are not widely used due to their distance limitations (of about 1, 15 and 500 m, respectively) (Gokhale 2001). These technologies have data rates over 2.5 Gbps.

3.1.6 Comparison of Wireless Technologies' Specifications

Modulating infrared, light and laser require clear line of sight which would be difficult under in-field conditions (with crop canopy), while short-range transceivers offer a flexible wireless technology for in-field use.

Citizen band radio has a greater transmission distance than short-range transceivers (up to 10 km line of sight compared to 100 m). The short-range transceivers may be arranged in a mesh network in the field in which the nodes find a path to the base station by relaying data from node to node. Therefore the addition of dedicated routing nodes in the network would incur an additional cost per node as well as potential inconvenience to the farm manager due to larger quantities of instrumentation and associated maintenance.

There is a trade-off between power output and power consumption of the node. The batteries of short-range transceiver nodes such as Zigbee last about one year, while the batteries of a citizen band radio last about one day. If the sensor nodes are left unattended on a farm for a number of years, a solar panel would be required to regularly recharge the citizen band radio's batteries. The additional solar panel increases the price of the node; however, the potentially increased reliability of citizen band to short-range transceivers may offset this additional cost.

Short-range transceivers offer significantly higher data rates than citizen band radios, i.e. while Zigbee transceivers can transmit at 250 kbps, the highest data rate offered by an existing citizen band radio telemetry system is 4800 bps. The lower baud rate of citizen band radio is acceptable for the low transmission requirements of most agricultural sensors. The price of a citizen band radio is also less than that of a short-range transceiver ($20 and $90, respectively). However, the price per node for the GME Telemetry System (which uses citizen band radio technology) is $1000 (Steven Rees, pers. comm., 10 June 2006).

3.2 Research Prototype Wireless Sensor Networks

Research trends in prototype sensor nodes for agricultural wireless networks include node characteristics of low cost, low power, multifunctional, small in size, and are able to communicate data between nodes over distances of up to 100 m (Akyilidiz et al. 2002).

Many of the wireless networks researched performed with near 100% reliability in an indoors environment (Zhang 2004), but do not indicate outdoor performance, eg. in Wall & King (2005). Reliable outdoor performance is a necessity for a successful in-field wireless network.

3.2.1 Literature Review of In-field Performance of Research Prototype Wireless Sensor Networks

Results of in-field tests were available for sensor nodes such as the mica mote and BTnode, both of which are academic research prototypes. These sensor nodes feature a microcontroller, a radio modem and connections to sensor boards such as barometric or humidity sensors (Karl & Willig 2005). Mica motes operate on the 915 MHz frequency band and can transmit and receive data at up to 250 kbps (see Figure 3.2), while BTnodes use Bluetooth as their radio technology, operate on frequencies between 433 and 915 MHz and operate at up to 3 Mbps.

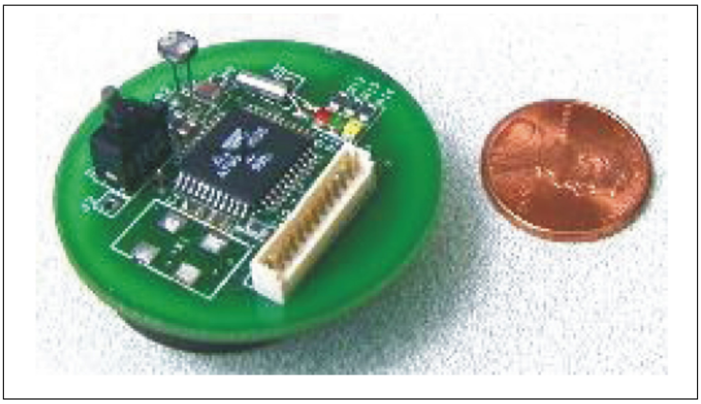


Figure 3.2: Mica mote node

Source: Karl & Willig (2005)

Field tests conducted by Cardell-Oliver, Smettem, Kranz & Mayer (2004) demonstrated that the transmission range of a Bluetooth system operating in the licence-free 2.4 GHz ISM band was affected by ground vegetation reflecting or absorbing some of the radio signals. The water absorption frequency band is 2.4 GHz, so the propagation of the radio signals was reduced by the on-farm water and the high water content of the vegetation. To maximise radio range, the radio antennae were placed away from the plants, with the optimal heights above the ground shown in Table 3.2. Placing the radios at these heights may cause difficulties in the implementation of the network, as it will potentially obstruct on-farm operations such as the use of machinery.

Table 3.2: Optimal height of Bluetooth 2.4 GHz transceivers in field

Environment	**Optimal radio height (m)**	**Radio range (m)**
Bare soil	1.4	44
Soybean field	1.7	37
Corn field	4.0	25

Source: Zhang (2004)

Shih et al. (2001) reported that under field conditions Mica2 (mica mote) sensor nodes had good reception when up to 30 m apart, but intermittently poor reception when as close as 5 m apart. This system had an overall successful message delivery rate of 63.8%. The Mica2 nodes were operating on the 433 MHz frequency band, so the absorption of radio waves by water may not have caused the reduced reliability.

Morais, Valenta & Serhatodio (2005) developed a network of solar powered wireless nodes connected in a mesh topology using the IEEE 802.11 standard. Communications were based on a low-power radio frequency link operating in the 433 MHz frequency band with a 150 m coverage range. This research also identified effective power supplies of wireless sensor nodes as a major problem to be addressed because the scaling down in size of electronics and sensor nodes has outpaced the scaling of energy density in batteries (Morais et al. 2005).

Personal communication with mechatronic engineer, Dr Stuart McCarthy (pers. comm., 16 May 2006), who has experience testing nodes of a wireless sensor network, revealed intermittent reception of 2.4 GHz sensor nodes at transmission distances up to 100 m.

This intermittent reliability was attributed to atmospheric conditions, water content of the vegetation, and transmission line of sight being impeded by vegetation.

In-field testing of a radio frequency system developed by Latimer & Reddell (1990) demonstrated that noise from nearby power lines, surrounding electrical equipment and other radio signals reduced communication reliability. This reduced reliability was associated with the low output power of the radio frequency device, and so could be avoided by using a higher power radio, such as a citizen band radio (which has output power of between 0.5 W and 5 W).

3.2.2 Comparison of Research Wireless Sensor Networks

The literature reports that the mica mote and Bluetooth sensor nodes had significantly poorer performance in the field than in the indoor environment. The presence of the crop canopy and water in the field led to reduced and intermittent reliability of the wireless network. Therefore, in developing a new agricultural telemetry system, there is potential to reduce farm manager labour, and offer a wireless sensor network with improved reliability to existing research sensor networks. Considerations in providing a reliable system include:

- Radios operating in the 2.4 GHz frequency band have a reduced transmission range and have to be placed away from crop to compensate
- Power lines and other radio signals cause interference to low power radios
- Some sensor nodes have intermittent in-field reliability (poor reception when nodes were as low as 5 m apart)

The wireless technology implemented in the wireless sensor network should thus avoid or reduce the limitations of some of the researched systems.

3.3 Commercial Wireless Sensor Networks

Sensor nodes available commercially include NICTOR™, Adcon Telemetry Units and GME Electrophone Telemetry System. NICTOR™ is a wireless sensor network developed by Terracept for environmental monitoring, or in conjunction with infrastructure to control operations for precision agriculture.

3.3.1 Literature Review of Specifications of Commercial Wireless Sensor Networks

Nodes of the NICTOR™ network can be added and removed by the end-user without any manual configuration, with up to 1000 nodes in the network. The nodes are compliant with Zigbee and use the licence-free 2.4 GHz frequency band. The data is encrypted to increase data privacy, and any failed nodes that are identified are automatically reported to the end-user.

Commercial products using UHF citizen band radio to transmit data are available, such as the CB Monitor (by Electrosense) and GME Telemetry System, and are used to monitor water levels in dams, control equipment such as irrigation pumps and open and close gates (*Citizen band radio* 2006). The CB Monitor system is available with only UHF citizen band radios, or with citizen band radios in conjunction with short-range transceivers (*CB Monitor Sensor and Telemetry System* 2006). Up to four outstations are polled at user defined intervals, and the data received by the base station is displayed on a user interface. The outstations are powered by 6 V solar panels.

UHF citizen band is a popular radio communication method for farmers in rural areas. Postdoctorate USQ researcher, Dr Andrew Maxwell, referred to UHF citizen band radio as a reliable and long distance data communication method (pers. comm., 11 May 2006). The GME Telemetry System (see Figure 3.3) is able to achieve transmission ranges of up to 100 km between the base station and an outstation, with typical data rates of between 1200 and 4800 baud.

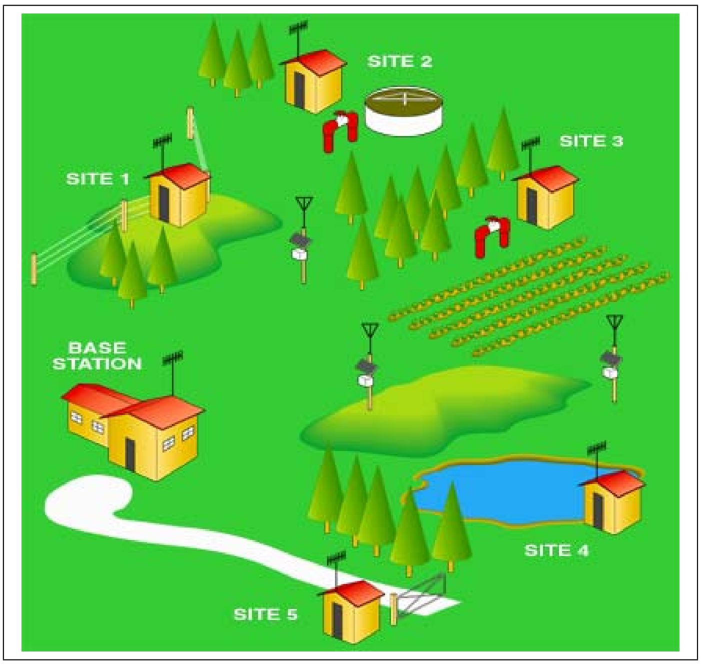


Figure 3.3: Sample installation of GME Electrophone Telemetry System with five remote monitoring sites

Source: *GME Telemetry System* (2005)

3.3.2 Comparison of Commercial Wireless Sensor Networks

The performance of commercial telemetry systems such as the GME Telemetry System and the NICTOR^{TM} was unknown due to no publicly available documentation, and there was limited information available about the technologies and protocols that these systems used. Therefore results from academic research papers were the only resource for actual in-field performance of wireless technologies, and no papers have considered citizen band networks.

3.4 Conclusions

The common difficulty with existing wireless sensor networks is the unreliable performance of the nodes under field conditions such as in the presence of vegetation, water and power lines that cause interference. The need to provide a telemetry system that is reliable and inexpensive heavily influenced the wireless technology chosen for the telemetry system. UHF citizen band radio, with long transmission distance and reliable operation for amateur radio communications, emerged as a promising candidate for the wireless transmission medium.

Chapter 4

Conceptual Investigations

The methods of wireless communication that have been further considered from the literature review include voice or data transmission over UHF citizen band radios. The citizen band radios may potentially also be interfaced with short range transceivers or sound modules for areas of the field where long transmission distances are not necessary. Preliminary field tests were conducted to evaluate the feasibility of a wireless network using UHF citizen band radios as the transmission medium.

4.1 Wireless Technologies for New Telemetry System

4.1.1 UHF Citizen Band Data Transmission

The use of UHF citizen band radio to transmit data is not new, as there are at least two commercial telemetry systems for agriculture that utilise this transmission medium (see Chapter 3). The data rate at which both of these systems transmit is low (between 1200 and 4800 bits per second), compared with wireless communication medium such as IEEE 802.11 and Zigbee which are capable of transmitting at over 250 kilobits per second. However, agricultural sensors, that measure values such as soil moisture, water tank level and flow rate of irrigation water, have low data rate requirements since they only transmit the sensor value, and the day, date and time when this value was recorded.

Therefore, for an agricultural telemetry system, the low data rates typical of UHF citizen band radio is not a restriction, and citizen band radio is a viable transmission medium. However, there are no documented results for a citizen band telemetry system with nodes in canopy conditions, a price comparable to short-range transceivers or a data rate comparable with short-range transceivers.

4.1.2 Integrating Short-range Transceivers with Citizen Band Radio

Short-range transceivers such as Zigbee and sound-playing modules may potentially be interfaced with in-field UHF citizen band radios to relay data to the base station as illustrated in Figure 4.1(b). Benefits of using short-range transceivers include reducing the number of radios required in the field, by replacing the radios in close range which do not require a long transmission distance with short-range transceivers.

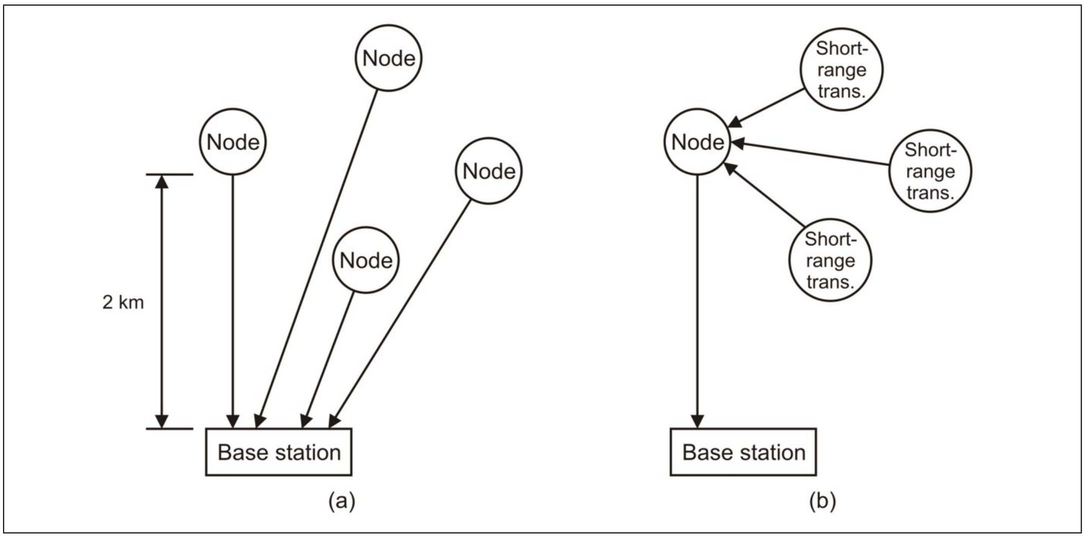


Figure 4.1: Conceptual network configuration with: (a) UHF citizen band radios at all sensor nodes; and (b) short-range transceivers at some sensor nodes

Electrical engineering lecturer, Mark Phythian (pers. comm., 23 March 2006) suggested that transmitters such as Zigbee or those used in garage door openers be used for short-range communications in the field to a nearby citizen band radio. This citizen band radio, which would comprise a Zigbee short-range transceiver to receive the signals from the in-field Zigbee node, would then encode and relay the wireless data over the UHF citizen band radio to the base station.

Sound playing modules, such as those used in musical greeting cards, may be placed in the field on each sensor (Professor John Billingsley, pers. comm., 3 March 2006). An interface with citizen band radios would involve the sound modules playing a unique audible sound to indicate when water is detected, which is received by a close UHF citizen band radio and then transmitted to the base station by that UHF citizen band radio.

4.1.3 Wireless Technology Chosen

Citizen band radio was the transmission medium explored for the agricultural telemetry system to be developed. The large transmission distance and simplicity of a citizen band radio network made this transmission medium an attractive avenue to investigate. However, the wireless technology chosen for the agricultural network depends on the in-field performance of the transmission medium, so a preliminary in-field evaluation using UHF citizen band radios was required to be conducted.

4.2 Feasibility of Citizen Band for Telemetry System

The viability of using UHF citizen band radio as a transmission medium for an agricultural telemetry system was investigated, to indicate the reliability and distance capabilities of citizen band radio in an agricultural environment. These preliminary tests aimed to evaluate voice and tone encoded transmission over radios.

In-field tests were conducted under worst case scenario conditions, with the radio positioned in a crop canopy and either in the presence of irrigation water or close to the farm's water storage. The preliminary tests were conducted in February on a cotton farm in Jondaryan. The data over the citizen band radio was modulated using both voice and audio tones as permitted on UHF citizen band channels (channels 22 and 23 are only for data use and all other channels are only for voice communication (*Citizen band radio* 2006)).

4.2.1 Voice Modulation

Voice encoded transmission was tested first with the signal transmitted by a radio in the field, and received by a radio two kilometres away. The antenna of the transmitting radio in cotton crop canopy is shown in Figure 4.2 mounted on an red tube.

Figure 4.2: Transmitter in cotton crop canopy during an irrigation on Jondaryan farm

The signal transmitted from the in-field transmitter to the receiving station was played from an MP3 player and consisted of a voice speaking the numbers one to ten. The voice signals were received by a radio at the edge of the field two kilometres from the transmitter in the crop canopy (see Figure 4.3) and near irrigation water (see Figure 4.4). Speech was also recorded onto a recordable sound module, however the sound quality was poor. See Appendix B for details of the sound module and MP3 player circuits built.

The receiving station consisted of a laptop, UHF citizen band radio and a 12 V lead acid battery to power the radio. The UHF citizen band radio used was the Uniden uh012sx which had a power output of 5 W (the maximum legal power output for a citizen band radio). The speaker output of the radio was connected to the laptop and the laptop recorded the incoming audio to a wave file.

Figure 4.5 is the waveform of a voice recording of the words one to ten transmitted and received over a UHF citizen band radio. There was a significant amount of noise in the signal after it was transmitted over a UHF citizen band radio.

Figure 4.3: Base station radio on cotton farm in Jondaryan

Figure 4.4: Irrigation water near testing site in Jondaryan

The recorded signal was loaded into a speech recognition software package, SAPI 5.1, in Borland Delphi Version 6 (Long 2006). However, SAPI 5.1 was unable to recognise the words spoken in the recording due to the presence of noise in the signal. Further filtering to reduce the noise in the signal and template matching techniques such as correlation can potentially be conducted to decode the number being sent (Leis 2002).

For sensors with high data rate requirements, the low data rate of voice-encoded data will potentially restrict the usefulness of voice-encoded data over a UHF citizen band radio. If an acknowledgement system is implemented, sensor nodes would require voice decoding functions, i.e. more sophisticated hardware/software.

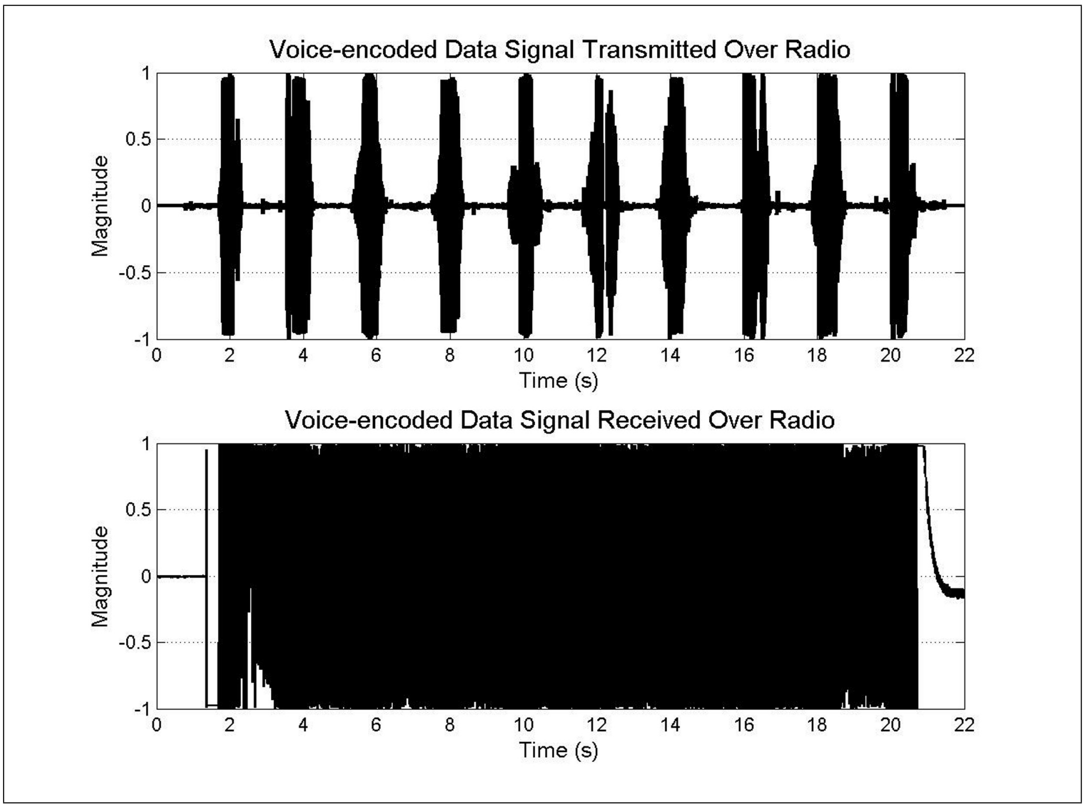


Figure 4.5: Original and recorded voice waveform of spoken words over UHF citizen band radio

The voice to be transmitted over the radio may be synthethised using a text-to-speech synthesiser module (such as the WTS701 integrated chip costing about $170 each) which outputs a speaking voice corresponding to a series of characters it receives. A speech recognition chip would be able to decode the speech remotely for control applications, for example using Sensory Inc's RSC-364 voice recognition module (costing approximately $40 each), in conjunction with appropriate noise-reducing filters. These are expensive when compared to the XBee Zigbee wireless module which costs approximately $90 each.

The modulation and demodulation of data over UHF citizen band radio using audio tones was then investigated.

4.2.2 Tone Modulation

Prerecorded audio tones were then transmitted over the radio and the transmitted and received waveforms were compared. The same hardware was used for the audio tone modulation testing as the voice-encoded data testing in Section 4.2.1, except tones were transmitted instead of voice signals.

A dual tone frequency signal was transmitted from a UHF citizen band radio in the crop canopy and received by a radio two kilometres away. The two frequencies were 697 Hz and 1209 Hz. The Fast Fourier Transform was taken of the received signal (see Figure 4.6) from which the frequency components were accurately obtained.

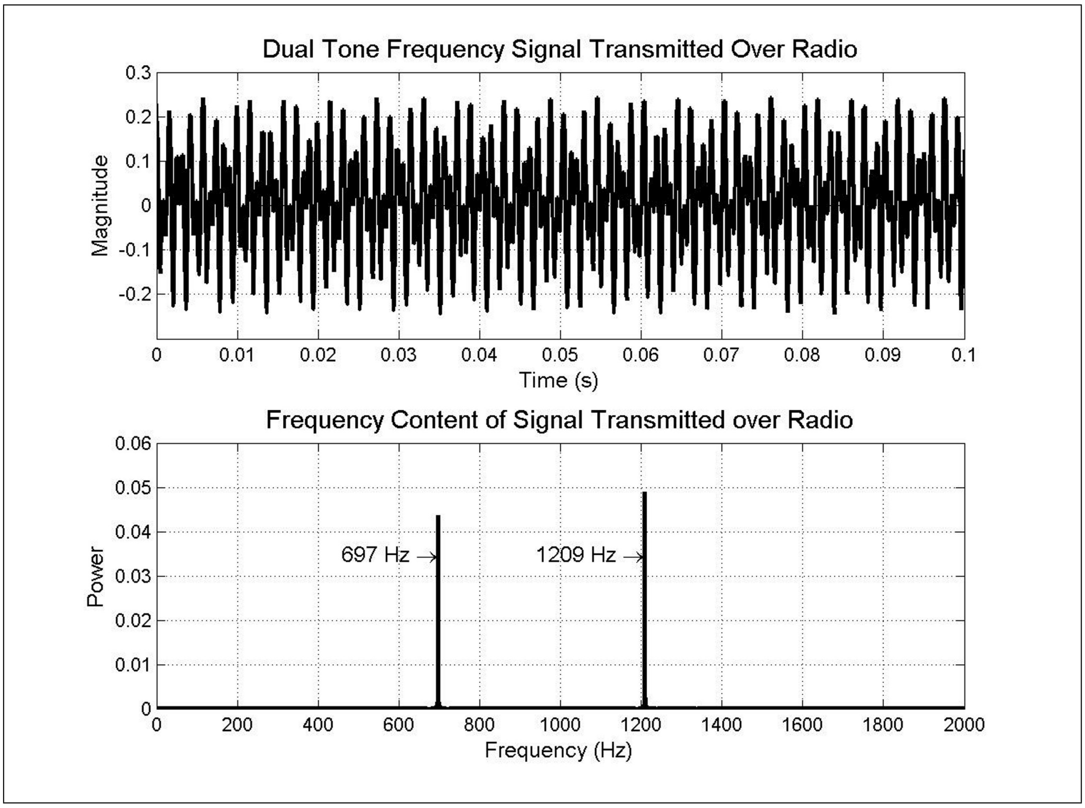


Figure 4.6: Waveform and Fast Fourier Transform of dual tone signal received by radio

4.3 Conclusions

Since the audio tone testing was completed under a worst case scenario (with canopy and irrigation water), this preliminary evaluation of citizen band radio concluded that clear data signals could be transmitted and reliably received over UHF citizen band radio in an agricultural environment. Therefore, the modulation and demodulation of data for the telemetry system being developed will be implemented using audio tones over citizen band radio.

Citizen band radio also reported high reliability for data transmission, while in-field testing of networks using short-range transceivers demonstrated reduced reliability under field conditions such as crop canopy, irrigation water and power lines.

Later chapters will deal with data demodulation and network protocols. These are fundamental concepts for wireless transmission and will affect the developed system's data rate, reliability and scheduling of data transmissions.

Chapter 5

Data Modulation and Demodulation

Modulation is the varying of a signal to convey information, i.e. a signal sent over the UHF citizen band radio can be changed in volume or frequency to represent logic 0 and 1, thus sending digital data over an analogue communication channel. Demodulation is the extraction of digital data from an audio channel. Modulation and demodulation are required to transmit serial data at a higher data rate than achieved by voice.

There are different types of modulation, such as frequency, amplitude and phase modulation. For example, frequency modulation is the altering of the frequency of the signal to represent a serial binary sequence. These modulation techniques may also be combined.

Modulation and demodulation may be achieved using (a) software, where the base station or node is connected to a computer running software to analyse incoming signal; or (b) hardware, consisting of a radio and decoder circuit which may use a modem integrated circuit.

The minimum requirement for each node is to have modulation capabilities, so that each node can transmit sensor data. The need for the node to demodulate as well depends on the network topology and protocol, as will be discussed in Chapter 6. For example, mesh nodes must modulate and demodulate to relay data, while nodes that receive acknowledgement signals must also modulate and demodulate.

5.1 Software-based Decoding at Base Station

Audio tones received by a radio at the base station may be demodulated using custom-written software (see Figure 5.1 for flowchart of node and base). Software in Borland Delphi Version 6 and Matlab Version 7 were written to implement this process.

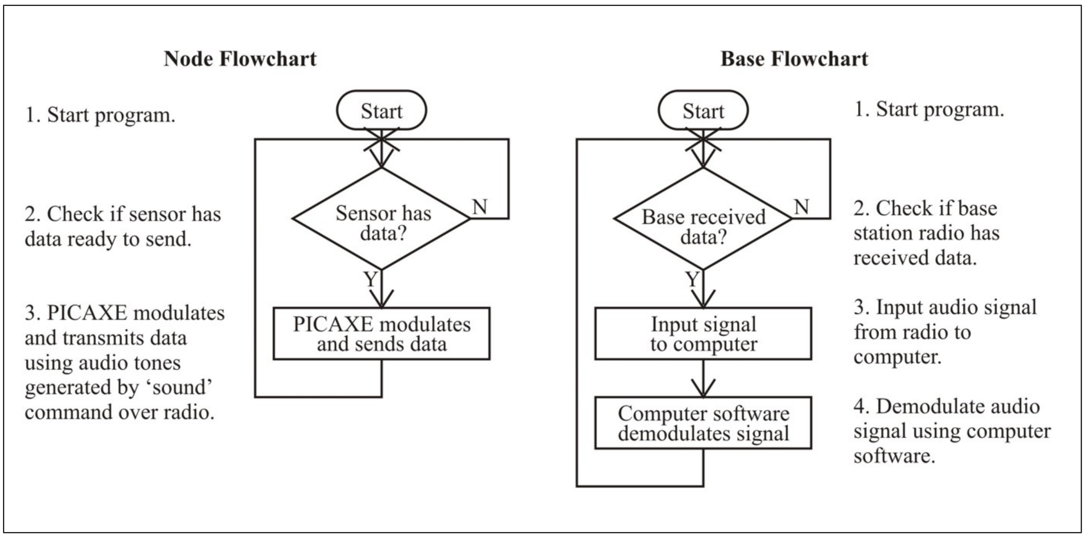


Figure 5.1: Flowchart for node and base with software-based decoding

Software-based decoding at the base station is limited in that nodes could not receive acknowledgements and a dedicated base station computer would be required to continuously read in the data.

The tones used to test the developed software were generated by a Borland Delphi Version 6 component, AudioLab Version 2.2, which is shareware created by Mitov Software (*AudioLab 2.2* 2006). The transmitted signal consisted of five-second blocks of the frequencies ranging from 200 Hz to 12000 Hz in increments of 200 Hz.

Figure 5.2 illustrates the connections required at the base station and each node for a system using software-based decoding. The speaker output of the computer was attached to the microphone input of a UHF citizen band radio, and the sound was transmitted over the radio. The headphone output of another radio was connected to the microphone of another computer which received these tones.

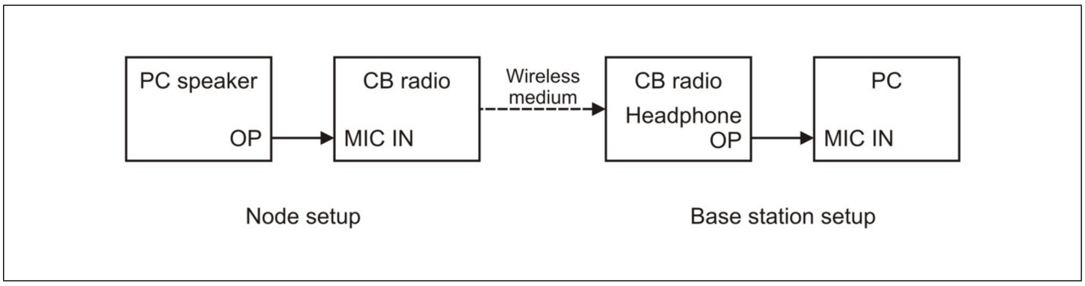


Figure 5.2: Block diagram of node and base station setup for software-based decoding

Limitations of this setup include the requirement of (a) a dedicated base station computer to continuously read in data; and (b) a computer at each node if the node is to receive an acknowledgement.

Commercial software packages that decode audio tones for use with amateur radio are also available, which are able to decode audio tones of different frequencies and baud rates.

5.1.1 Software in Delphi

A Delphi application was written which demodulated incoming audio tones from a radio using the AudioLab Fast Fourier Transform component. The Fast Fourier Transform identifies the continuous spectrum of frequency components of a function. By analysing the microphone input of the computer, the Fast Fourier Transform can decode the signals over the radio in real-time. For example, a dual tone signal consisting of 1209 Hz and 697 Hz was transmitted over a UHF citizen band radio. A Fast Fourier Transform of this signal (as received by another radio) identified the two frequency components, as shown in Figure 5.3 (a screen capture of the written Delphi application).

However, due to a software limitation, the maximum sample rate for the Fast Fourier Transform in AudioLab was 1200 Hz (833 samples a second). By Shannon's sampling theorem a suitable sampling frequency is twice the highest frequency component of the input signal, so the maximum tone of signal that could be accurately decided in

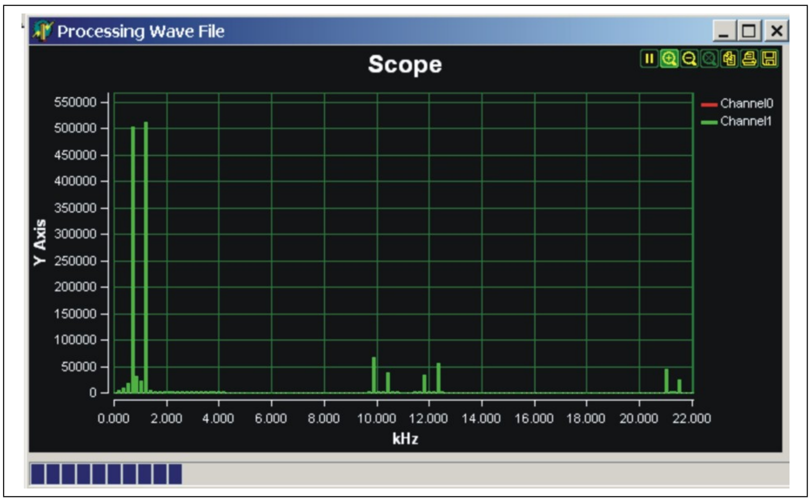


Figure 5.3: Screen capture of Delphi application using AudioLab

AudioLab was 600 Hz. Since the AudioLab component in Delphi restricts the baud rate at which a signal can be decoded, Matlab Version 7 (which allows a greater sampling frequency) was next investigated to implement a software-based decoder.

5.1.2 Software in Matlab

A Matlab Version 7 script was written to decode audio tones using a sampling rate of up to 44 100 Hz. The Matlab code that decoded the tones was based on the Spectrum Scope application written by Scott Hirsch in Matlab Version 7 (Hirsch 2006). This application also used a Fast Fourier Transform to decode the audio signal. However, a disadvantage of this approach was that Matlab was unable to process data from the microphone input in real-time, and the 'wavread' command in Matlab was required to record the audio from radio before processing. This was seen as a major limitation to using Matlab as a software-based decoder.

5.1.3 PICAXE Circuit to Transmit Signals for Software-Based Decoding

The audio signal transmitted over the radio (to be received by the Matlab and Delphi software) was modulated using a PICAXE microcontroller. The PICAXE microcontroller is inexpensive, as it does not require a programmer or eraser. The PICAXE 'sound' command was used to output a series of audible frequencies based on specified inputs, as follows:

```
SOUND pin,(note,duration,note,duration...)
```

The 'sound' command outputs audible sounds on the pin specified by 'pin'. The sound is a series of audio tones or notes, each for a different duration, which are each specified by 'note' and 'duration', respectively. A circuit was built with the PICAXE microcontroller (see Figure 5.4) which:

- Turned on the radio when the node had data to transmit (by connecting battery, pressing on button, then pressing transmit button via relays)
- Transmitted a series of tones through a stereo plug connected to the microphone jack of a UHF citizen band radio

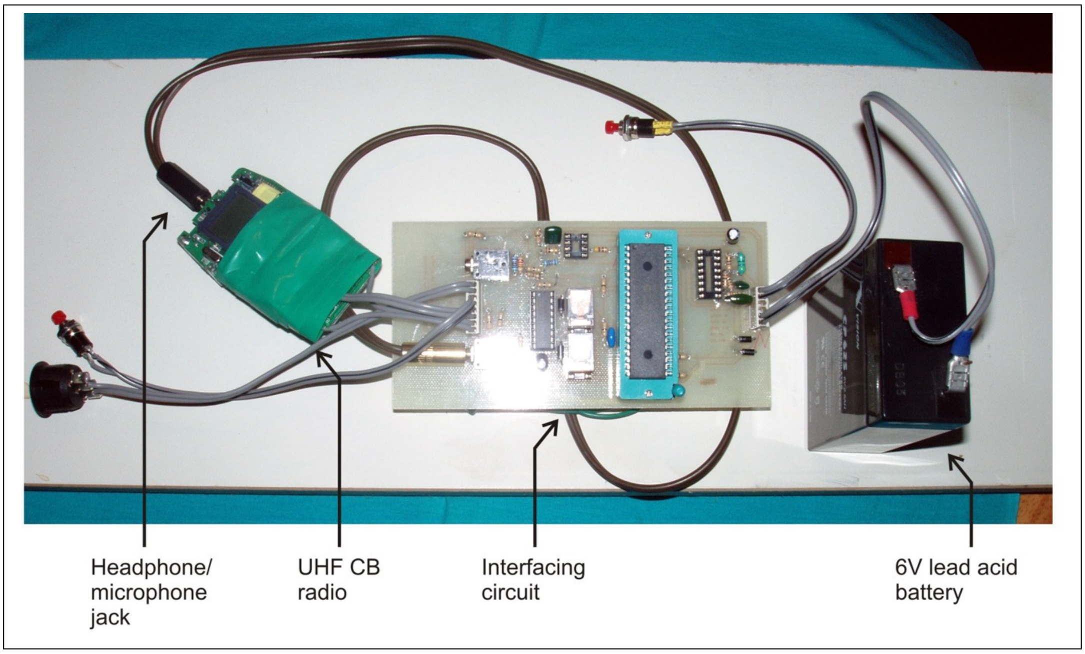


Figure 5.4: Prototype with PICAXE creating audio tones

See Appendix B for the schematic diagram of the interfacing circuit.

5.1.4 Conclusions on Software-based Decoding at Base Station

A software-based decoding base station was initially considered to be a simpler option than a hardware-based decoder; however, computer software limitations and the need for a dedicated computer to interpret the incoming data has made a hardware decoder a viable method. A hardware-based decoder could be implemented without a computer required at the base station or nodes.

5.2 Hardware-based Decoding at Base Station

Audio tones are decoded in hardware using manually built circuits or integrated chips. The method of decoding the signal at the base station depends on how the data is modulated. The two methods of modulating the tones onto the UHF citizen band radio are (a) baseband where data is directly modulated onto the carrier radio wave by modifying the radio; and (b) broadband where encoded audio tones are transmitted over the radio by connecting the microphone of the radio to the output of the modulator without modifying the radio.

Modifications made to UHF citizen band radios must adhere to the Citizen Band Radio Stations Class Licence restrictions. This requires considerable expertise of citizen band radio circuitry, hence was not pursued as an option. Therefore broadband modulation was used. Figure 5.5 illustrates a software flowchart for a network node with a hardware demodulator, developed to test the hardware-based decoding.

5.3 Choosing a Modulation Technique

The chosen modulation method depends on the availability of hardware to modulate and decode signals over an analogue voice communication channel (such as UHF citizen band radio). Audio tones may be encoded with sensor data by varying the amplitude, frequency or phase of an analogue signal to be transmitted over the radio. Figure 5.6 illustrates the modulation of data using various modulation techniques, where each

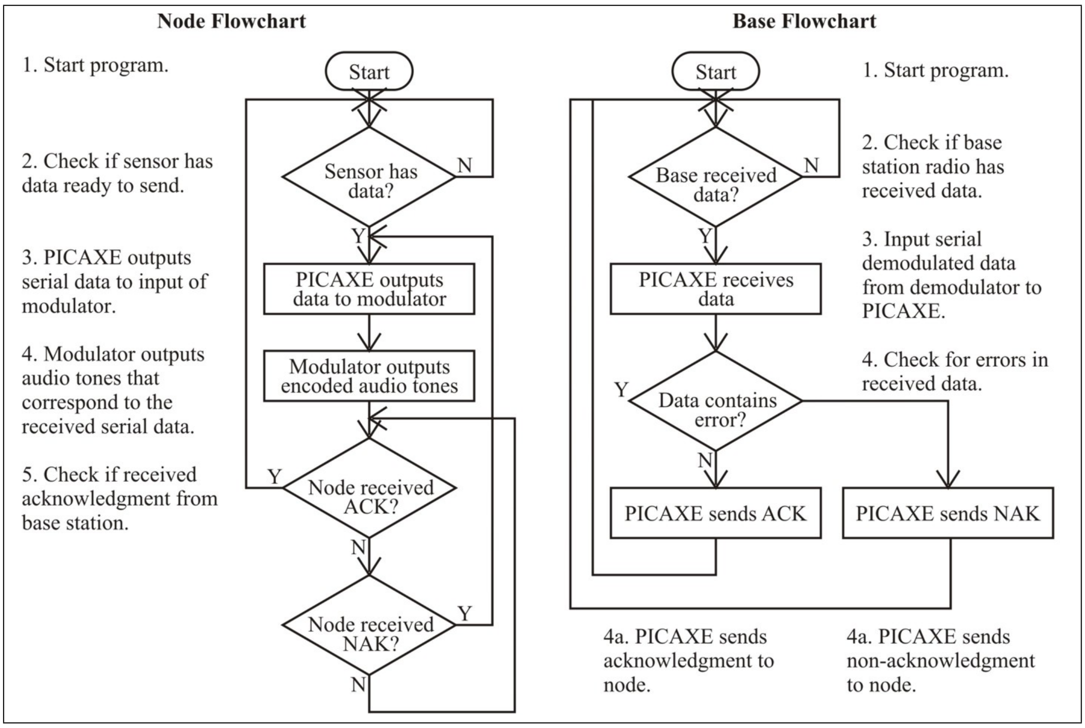


Figure 5.5: Flowchart for node and base with hardware-based decoding

change in the waveform represents one bit of data.

5.3.1 On-off Keying

On-off keying represents the transmitted data using the presence or absence of a signal (see Figure 5.6(b)). Some forms of on-off keying include pulse coded, amplitude, duration and frequency modulation. Examples of on-off keying modulation are impulse coding and Morse code, where impulse coding sends impulses of constant length with variable pulses between them, while Morse code sends a series of impulses of different length.

5.3.2 Amplitude-shift Keying

Amplitude-shift keying changes the amplitude of the transmitted signal to represent a binary sequence. A logic 0 is represented by a low amplitude and a logic 1 is represented by a high amplitude (see Figure 5.6(c)). Amplitude modulation is simple to produce,

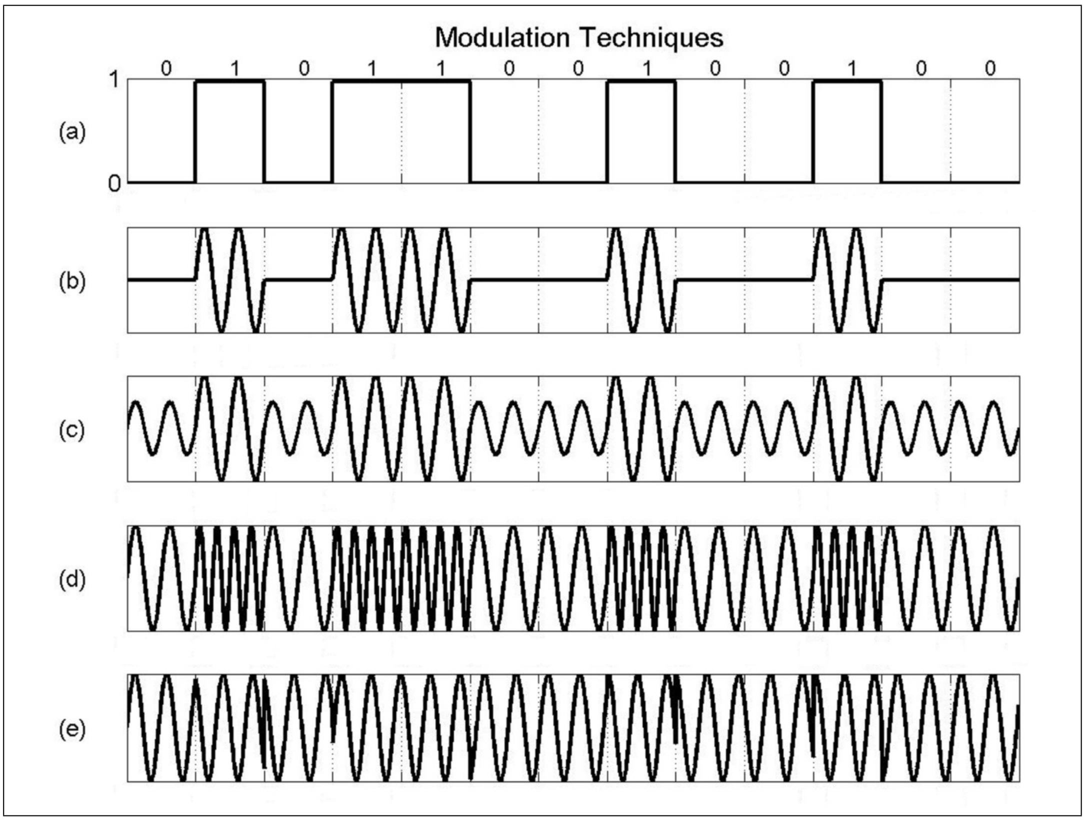


Figure 5.6: Audio modulation techniques: (a) sequence of digital data bits; (b) on-off shift keying; (c) amplitude-shift keying; (d) audio frequency-shift keying; and (e) audio phase-shift keying

and off-the-shelf integrated circuit modems can be purchased such as the TDA5051 and LM1596. These modems produce output voltages proportional to an input signal and demodulate signals by filtering out all of the signal components except for the desired audio. However, amplitude modulation is susceptible to static and other forms of electrical noise.

5.3.3 Audio Phase-shift Keying

Audio phase-shift keying changes the phase of the the signal, with a logic 1 represented by a phase change of 180°, and a logic 0 represented by no phase change (see Figure 5.6(e)). If a phase shift of 90° is used between each time slot, two bits can be transmitted at a time, effectively doubling the potential baud rate. Phase modulation requires more complex decoding hardware than other modulation methods, due to possible ambiguity for decoding signals of phase 0° and 180°.

5.3.4 Audio Frequency-shift Keying

Audio frequency-shift keying (AFSK) is a data modulation method where, for a binary sequence, a logic 1 is represented by a higher frequency and a logic 0 is represented by a lower frequency (see Figure 5.6(d)). Frequency modulation has a lower signal to noise ratio than amplitude modulation, thus sound quality is higher than for amplitude modulation.

Many traditional digital data storage and transmission applications use frequency modulation, such as amateur radio and data storage on audio cassettes. Amateur radio historically uses 1200 Hz and 2200 Hz to represent logic 0 and 1 respectively for 1200 baud transmission, while the Kansas City Standard which is employed to store data on audio cassettes, uses one cycle of 1200 Hz and two cycles of 2400 Hz represent logic 0 and 1, respectively.

Frequency modulation may be implemented using integrated circuit modems available off-the-shelf, such as TCM3105, FX614 and the XR2206 modulator/XR2211 demodulator pair.

5.3.5 Multiplexing

Modulated audio signals may be added together to increase the data rate over the radio channel (eg. multiple frequency-shift keying, sub-audible tones and sub-carriers). Multiple shift-keying outputs two or more frequencies simultaneously, and so can represent two or more bits of data. Sub-audible signalling involves audio tones of frequencies below the human audible range (less than about 300 Hz) being added to the carrier wave of citizen band radios (baseband modulation).

Sub-audible tones may also be used to ensure the privacy of the conversation, where the transmitting radio sends a signal with a sub-audible tone in the background, and the receiver is modified to only receive signals containing that particular sub-audible tone.

5.3.6 Chosen Modulation Technique

The simplicity, availability and widespread use of audio frequency-shift keying equipment for amateur radio indicates that frequency modulation is a reliable and proven method to encode and decode data over an analogue voice communication channel such as UHF citizen band radio, and therefore will be employed in this telemetry system.

5.4 Evaluation of Frequency Shift-keying Modems

Frequency shift-keying can be implemented using off-the-shelf integrated circuits. These chips modulate data with frequencies using a voltage controlled oscillator. Voltage controlled oscillators are built with a transistor or high-frequency operational amplifier and a variable capacitor in the feedback circuit. An amplifier is required to boost the signal from milliWatt to Watt levels. The modulated frequencies can be demodulated in various ways including:

- Zero crossing detector (determines frequency by detecting when received waveform crosses the zero volt level)
- Filtering out frequency components except those expected and use comparator to demodulate
- Phase locked loop
- Frequency to voltage conversion

These modulation and demodulation techniques may be implemented manually using off-the-shelf components, bought in an integrated circuit, or written in PICAXE software. Frequency-shift keying modem integrated circuits that encode data as audio tones based on a binary sequence received by the chip include TCM3105, FX614 and XR2206 FSK encoder. The TCM3105 and FX614 also have decoding capabilities, while the XR2206 encoder may be partnered with the XR2211 decoder for a fully functional modem.

The TCM3105, FX614 and XR2206 and XR2211 pair were evaluated, and the major connections between each of these modems, the PICAXE microcontroller and the UHF citizen band radio are illustrated in Figure 5.7.

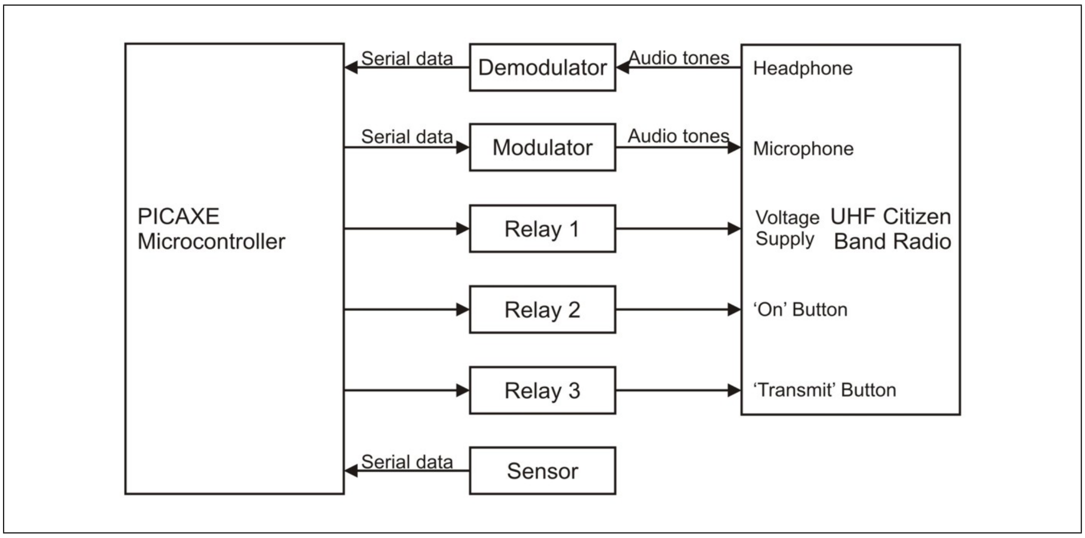


Figure 5.7: Block diagram of interfacing circuit

5.4.1 TCM3105

The TCM3105 modem integrated circuit was first evaluated by connecting the modem as per the data sheet in Appendix G and with other components as per Figure 5.7. The preset modulation standards implemented by the TCM3105 modem include transmission at 600 and 1200 baud using 1200 Hz to encode a logic 0 and 2200 Hz to encode a logic 1. Figure 5.8 displays the circuit board built and used to evaluate the TCM3105.

A series of audio tones corresponding to the PICAXE's serial output was transmitted over a UHF citizen band radio by the TCM3105. A second TCM3105 was connected to a receiving radio to demodulate the signal; however, none of the incoming tones were successfully decoded at 600 or 1200 baud. The TCM3105 would not decode the audio tones even when the radio connection was replaced with a direct connection between the output of the modulator and the input of the demodulator.

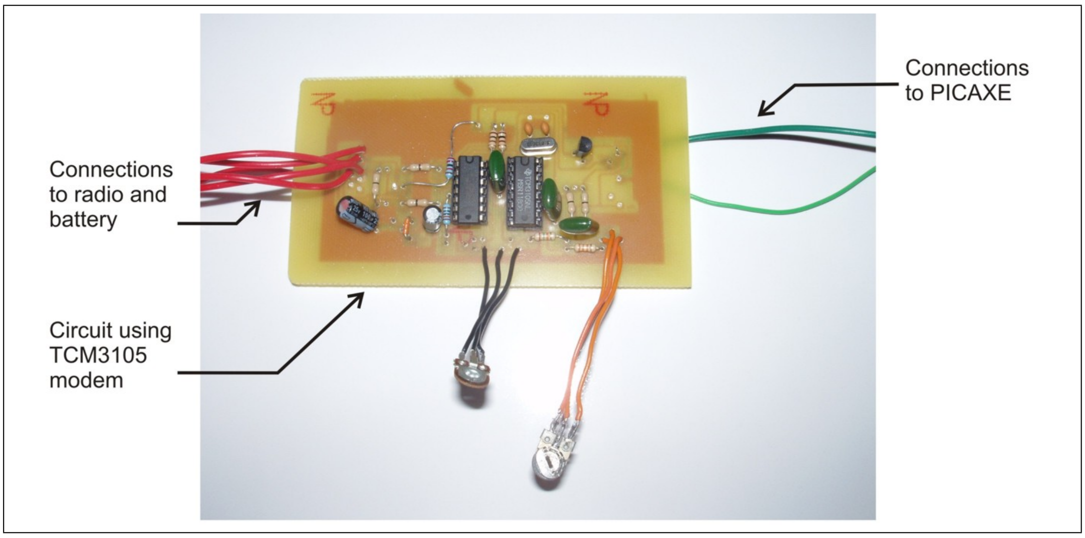


Figure 5.8: Printed circuit board using TCM3105 modem

The actual frequencies of the modulated tones were then tested to ensure they were 1200 Hz and 2200 Hz, as specified in the data sheet (see Appendix G). A Fast Fourier Transform, performed on the output signal of the modem for a logic 0 input, revealed that the major frequency component of the signal was 1160 Hz, and not 1200 Hz (see Figure 5.9). This incorrect audio tone may not be recognised as a logic 0 by the demodulator, and this possibly explains why the demodulator would not decode signals from the modulator.

Further research on the TCM3105 revealed that Texas Instruments acknowledged that the TCM3105 did not actually use the 1200 Hz tone (*Tucson Amateur Packet Radio* 1996). Therefore, attempts to make a working circuit using the TCM3105 modem were abandoned, and another modem integrated circuit was evaluated.

5.4.2 FX614

The FX614 modem was next evaluated using external components as per the data sheet in Appendix G. The FX614 is used in amateur radio modems as a replacement for the now obsolete TCM3105 and, like the TCM3105, has a preset modulation standard of 1200 baud with a logic 0 represented by 1200 Hz and a logic 1 represent by 2200 Hz. This data rate may be increased by changing the crystal required for the timing of the modem, i.e. doubling the crystal frequency doubles baud rate.

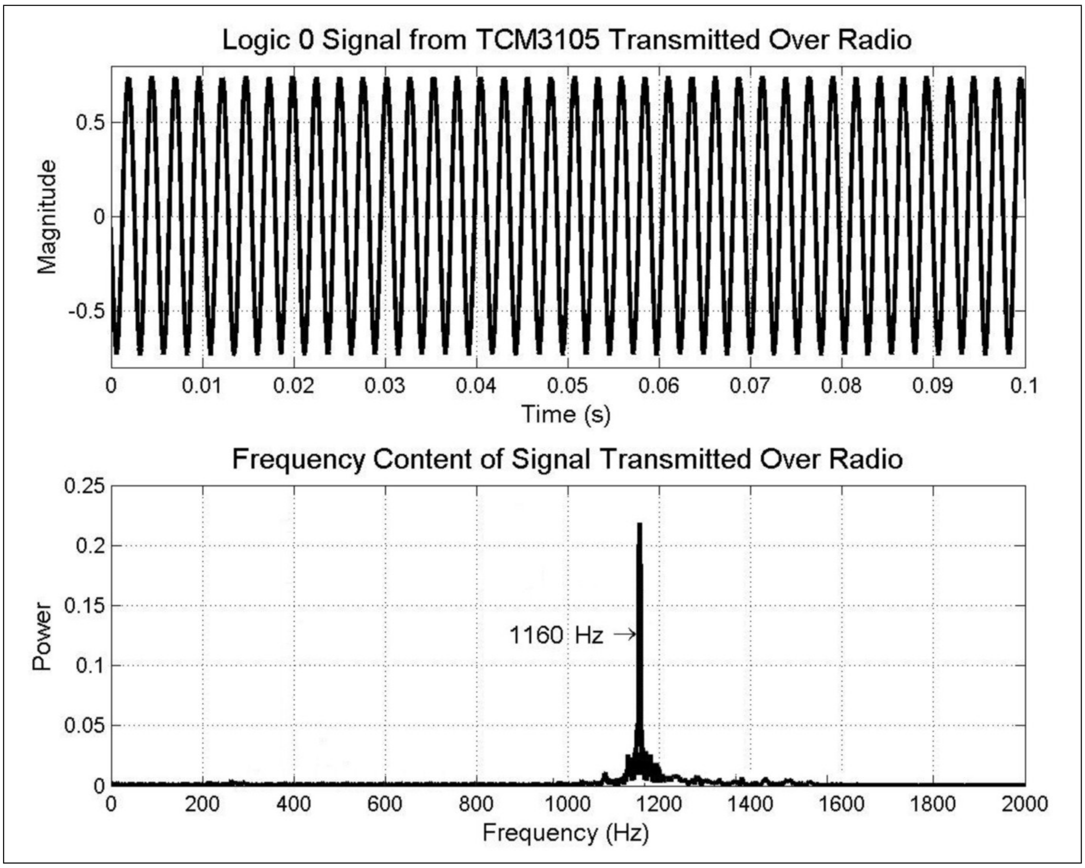


Figure 5.9: Waveform and Fast Fourier Transform of TCM3105 output signal

The circuit constructed using the FX614 modem did not demodulate incoming audio tones when directly wired to the output of the modulator. To avoid spending excessive time on making the built circuit work, this approach was abandoned.

5.4.3 XR2206 and XR2211

The XR2206 (modulator) and XR2211 (demodulator) pair were next attempted and were attractive due to the ability to change the baud rate and the frequencies used to represent a binary data sequence (see Appendix G for data sheets). Therefore, the XR2206 and XR2211 pair were more flexible than the TCM3105 and FX614 modems, as they potentially enable tests on signal reliability to be conducted at different frequencies and baud rates. The XR2206 required at least 10 V to operate, while the XR2211 required 4.5 V to operate.

The XR2206 and XR2211 were tuned to frequencies of 1200 Hz and 2200 Hz (logic 0 and 1, respectively) since these frequencies are used prevalently in amateur radio. Two circuits using this modulator and demodulator pair were built based on the Hamtronics 1200 baud modem (*1200 Baud Data Modems* 2006), a commercial amateur radio modem kit. The XR2211 successfully decoded audio signals transmitted by the XR2206 over the UHF citizen band radio at 600 and 1200 baud in a workshop environment. The baud rate of the audio tones was changed by varying the baud rate of the input serial data, and during this change in baud rates the audio tone frequencies were kept at 1200 Hz and 2200 Hz. Therefore, the XR2206 and XR2211 were chosen to modulate and demodulate data for the telemetry system.

5.5 Conclusions

Audio frequency-shift keying is a reliable method of transmitting data over analogue voice communication channels, as used in amateur radio modems, and therefore was used to encode data in the telemetry system. The XR2206 modulator and XR2211 demodulator successfully modulated and demodulated data over a UHF citizen band radio at 600 and 1200 baud in a workshop environment. These data rates are lower than the data rates achieved by short-range transceivers such as Zigbee, however are comparable to the data rates achieved by existing commercial agricultural telemetry systems such as the GME Telemetry System.

Chapter 6

Design of Network Protocol

With working hardware to implement the chosen modulation technique, the network protocol was investigated. For the design of the UHF citizen band radio network protocol, considerations included the sharing of the citizen band radio channel, the relaying of data through the network, the handling of errors in transmitted messages and the format of the transmitted messages. PICAXE microcontrollers were used at each node to implement these network protocols.

The data received at the base station must be presented such that the status of sensors is visible to the farm manager. Hence, a user interface was developed that graphically displayed the data received at the base station.

6.1 Network Topologies

The network topology is the method used to connect the nodes of the network. Four main network topologies are bus, star, mesh and ring (see Figure 6.1), which may be combined to form configurations such as star-ring and mesh-star networks.

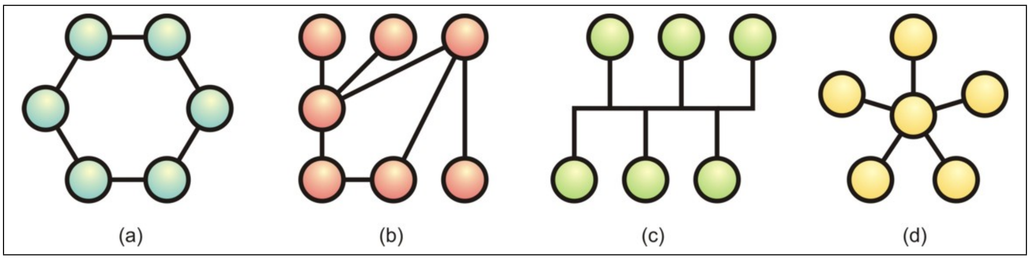


Figure 6.1: Network topologies (a) ring; (b) mesh; (c) bus; and (d) star

6.1.1 Ring, Bus, Mesh and Star Network Topologies

Ring and bus topologies do not have a central node; rather, the nodes of a ring network are linked to form a continuous loop, while the nodes of a bus network are connected via a communications line (Gibilisco 1999). A wireless telemetry system with a ring topology (see Figure 6.1 (a)) would feature the in-field nodes transmitting data directly from node to node in a ring on their own accord. One or a number of nodes would be allocated the task of transferring the data collected in the network to the base station in the farm manager's house.

Most wireless networks have bus topologies, where the communication line (such as UHF citizen band radio) is the radio transmission medium (see Figure 6.1 (c)). Ethernet local area network systems use a bus network topology to share the transmission medium.

A mesh topology features at least two paths to each node of the network (see Figure 6.1 (b)). In peer to peer mesh networking, the nodes act as both data originators and data routers (Akyilidiz et al. 2002). A mesh topology would be unnecessary for a citizen band radio network, since the transmission range of citizen band radios is greater than the required two kilometres and therefore routing radios are not required.

The star topology links all nodes in the network to a single point, the central node (see Figure 6.1(d)). In an agricultural telemetry system example of a star topology, each node would be connected to a sensor and transmit the sensor data directly to the base station, without the need for dedicated data routing radios.

6.1.2 Network Topology Chosen

The star network configuration was the most appropriate topology for a telemetry system using citizen band radios due to the radio range of this transmission medium. For a system requiring transmission distance of two kilometres, actual total coverage of the telemetry system is four kilometres, since the farm manager's house is typically located in the centre of the property.

A radio network with a star topology will comprise:

- a base station in the farm manager's house (to display sensor status)
- a consumer UHF citizen band radio and modem as a peripheral node on the network, with the modem designed to include:
  - data modulation and demodulation
  - a flexible input for connection to sensors or user control
- a network protocol implemented via software which includes:
  - data acknowledgements and error handling
  - data collision prevention

The transfer of sensor data from each node of the star network over a citizen band radio channel would have to be coordinated to avoid data collisions. Therefore, medium access methods applicable to a UHF citizen band radio network were investigated.

6.2 Medium Access Methods

A medium access method must be implemented in the network protocol to allow all the radios in the network to access the channel without colliding with other radio transmissions, due to the half-duplex nature of UHF citizen band radio. Data collisions are costly in terms of throughput because they potentially require both nodes to retransmit later, and there is potential for data to be lost in the network. If a collision occurs, there must be a way to recover from the problem. The two fundamental medium sharing techniques are controlled (or scheduled) and contention (or random) access.

6.2.1 Random Access Methods

Random access methods are characterised by nodes that typically transmit when data is waiting to be sent. Random media access protocols are useful for variable rate and multimedia traffic, where additional overhead and delay would occur using a scheduled access method (Gibilisco 1999). Ethernet LANs commonly use a random media method to transfer data in a network.

Protocols that use random media access methods include ALOHA and carrier sense multiple-access (Carter & Whitehead 2004). For the ALOHA protocol, the node transmits data when ready without monitoring the communications channel first. The central node transmits an acknowledgement if the data was received properly, and schedules a retransmission after a random retransmission delay if the transmitted packet has collided.

Carrier sense multiple-access involves the nodes monitoring the status of the channel before transmitting data. If a ready node senses the channel to be busy, it will wait a random delay and monitor the channel until the channel becomes idle, before transmitting its data. A variation of carrier sense multiple-access is if the channel is detected as being busy, the channel of the radio is changed, and two radios would be setup at the base station to receive. Each node in the network may transmit on the minute, and if the channel is busy, will transmit on another channel every minute.

Schemes that help overcome data collisions in a carrier sense multiple-access network are collision detection and avoidance (Halsall 2005). Collision detection is achieved by a transmitting radio listening to the channel to determine whether its transmission has resulted in a collision. Collision avoidance schemes use an agreed upon scheduling policy to determine whether a transmission is permitted, via a series of ready to send and clear to send (RTS-CTS) messages before transmitting the packet.

6.2.2 Scheduled Access Methods

Scheduled access methods schedule access to the transmission channel in an orderly manner so that collisions among transmissions are guaranteed not to occur. Scheduled access medium sharing techniques include reservation and polling networks.

Reservation systems require that nodes make explicit reservations to gain access to the transmission medium (Fitzgerald & Dennis 2005). Polling occurs when the central node asks each of the nodes if it has data to send to the central node. Two types of polling include roll-call and hub or token polling (Beyda 2000). Roll-call polling involves the central node consecutively interrogating each node until all nodes are polled. Hub polling is characterised by a node starting the poll and passing the poll to the next node on the circuit.

6.2.3 Medium Access Method Chosen for System

Carrier sense multiple-access was initially chosen as the network access method since it is used to communicate between amateur radios, which are very similar to UHF citizen band radio. Carrier sense multiple-access was also a more efficient approach than a scheduled access method since data from in-field sensors such as advance rate sensors may be required only one time during each watering (when frequent polls are unnecessary). However, the opportunity for data collisions to occur increased with a random access method, resulting in the retransmission of data and reducing the throughput of the network. Therefore a scheduled access method such as polling was considered to be a viable approach.

A polling system also allows the base station to control the order in which the nodes are polled, which may be in order of sensor identification or in order of priority as determined by the user interface. Therefore, a polling access method was implemented in the protocol.

6.3 Error Handling

Errors in the data transmission may occur due to noise in the signal caused by interference from electricity, interference from other citizen band radios, or as a result of a weak signal. The detection and correction of errors in the data transmitted will potentially avoid the need to retransmit if an error occurs, and increases the reliability of the telemetry system. One error handling scheme is the redundancy check that involves adding an extra bit to each byte of data sent in the message, such as parity bits, checksums and cyclic redundancy codes.

6.3.1 Acknowledgements and Non-acknowledgements

Acknowledgements (ACK) and non-acknowledgements (NAK) are used in network protocols to indicate whether a message was successfully received, i.e. without containing errors. An acknowledgement system was employed by the sensor network being developed, where if a node sends data without an error, the base station sends an acknowledgement to the node and then the next node is polled. If the data received from a node contains errors, the base station sends a non-acknowledgement to the node and the node retransmits.

6.3.2 Parity Bit Schemes

A parity bit scheme adds a parity bit to each byte of data transmitted, and is odd if the number of ones in the transmission is odd, or even if the number of ones in the transmission is even. However, if there are errors in two bits of the byte sent, or if errors have occurred in both the byte and the parity bit, the error will not be detected. Therefore, transmitting one parity bit with each byte will detect up to half of the errors that have occurred.

The use of more parity bits would increase the number of errors detected; however, the throughput would be reduced and the error detection would still not detect all the errors.

6.3.3 Checksum

A more sophisticated error checking scheme is the checksum. Checksums determine the integrity of data by storing the result of adding the bits in the corresponding rows or columns of each byte in the message (Leis 2006). The checksum transmitted in the message is compared to the checksum on the data received. However, the checksum cannot detect errors such as errors occurring in the checksum bit, the reordering of bytes in the message and multiple errors which sum to zero.

The checksum scheme is simple to calculate, and the throughput and performance are significantly increased on the parity bit error checking scheme (one for each byte or each bit of the bytes, depending on the type of checksum scheme used).

6.3.4 Cyclic Redundancy Schemes

A cyclic redundancy scheme is an error detection scheme which is simple to calculate and has good capabilities as an error detection scheme. The check bits are computed as follows and as detailed in Leis (2006):

1. Define a polynomial (called a generator) of N bits (where N is the number of bits in the data frame and is typically 16 or 32).
2. Treating the data frame as a number, divide this number by the generator polynomial.
3. Transmit the frame and then the remainder.
4. Perform the same calculation on the received data and compare the remainder.

Cyclic redundancy schemes are able to detect single-bit and double-bit errors, and this is an improvement on the parity-bit and checksum methods. The cyclic redundancy code in the received message and the code calculated using the received data are then compared.

6.3.5 Repetition Codes

Error detecting schemes would require at least two additional transmissions per error detected. These extra transmissions will potentially reduce the throughput of the base station, which has a transmission time restriction of three seconds in every hour under current citizen band licence restrictions.

Error correcting schemes such as repetition codes were investigated. A repetition code repeats bits of data in messages transmitted (Mackay 2003) to reduce the probability of errors. While the repetition code potentially reduces the number of retransmissions required since errors may be corrected, the extra bits to be transmitted in every message may reduce the throughput of the network. To repeat one data byte four times in a message at 600 baud, there is an additional transmission time of 53.33 milliseconds.

An improvement on the repetition code is the block code where a sequence of bits of length K are converted into a transmitted sequence of length N bits.

6.3.6 Implementation of Error Handling Scheme

The cyclic redundancy code and repetition code emerged as promising error handling schemes for further investigation, because of their error detecting capability and simplicity, respectively. The performance of the network protocol using cyclic redundancy code and repetition code was evaluated in the field (see Chapter 8 for results). More complex error correcting schemes were not attempted as they are outside the scope of this project.

An acknowledgement scheme was chosen to feature in the network protocol for reliable data transfer.

6.4 Asynchronous/Synchronous Transmission

The transmission and receipt of data in the network may be asynchronous or synchronous. Asynchronous transmissions transmit data intermittently, usually with a start and stop bit at the beginning and end of each set of data. Synchronous transmission use characters at the beginning of the data blocks to synchronise the data transmission (Karl & Willig 2005). Synchronous transmission is advantageous over asynchronous transmission when large blocks of data are being sent, since time is not wasted sending a start and stop bit for each character. The PICAXE 'serin' command was used to read the data serially from the demodulator and is of the following format:

```
SERIN pin,baudmode,(qualifier,qualifier...),#variable,#variable...
```

The qualifiers are optional variables which must be received in the stated order before the subsequent bytes will be received. Therefore the PICAXE code would use synchronous data transmission, with a series of bits before the data to enable the data to be received correctly.

6.5 Format of Data

Formatting the data to be transmitted over the network involved choosing a message format, as well as considering data compression techniques.

6.5.1 Data Compression

Data compression is a method of increasing the baud rate of the transmission, which would be advantageous for citizen band radio with the restricted transmission duty cycle of the transmission medium. Relatively small amounts of data will be transmitted in each packet for this agricultural telemetry system as shown in Tables 6.1, 6.2 and 6.3, so no data compression techniques were implemented. However, data heavy sensors such as cameras would require compression techniques, in conjunction with a higher data transmission, to transmit data over UHF citizen band radio.

6.5.2 Designed Communications Protocol for Telemetry System

The developed communications protocol for transmissions from the base station to the in-field nodes of the UHF citizen band radio telemetry system is shown in Table 6.1, and the protocol for transmissions from the node to the base station is shown in Tables 6.2 and 6.3 using a cyclic redundancy code and a repetition code (error handling schemes), respectively. Fixed-length messages were used in the communications protocol.

Both the cyclic redundancy code and repetition code were evaluated in the field (with results in Chapter 8).

Table 6.1: Packet format for transmissions from base station to in-field node

Bytes	*Field Name*	*Purpose*
Bytes 1-5	Preamble byte	Activates PICAXE to start receiving
Byte 6	Start of message byte	Denotes start of data transmission
Byte 7	Transmitter code	Unique identifier for base station
Byte 8	Receiver code	Unique identifier for in-field node to poll
Byte 9	Poll	$FF for poll, $00 for no poll
Byte 10	Receiver code	Unique identifier for in-field node to acknowledge
Byte 11	Acknowledgement	$FF for ACK, $00 for NAK
Byte 12	End of message byte	Denotes end of data transmission

Table 6.2: Packet format for data transmissions using cyclic redundancy code

Bytes	*Field Name*	*Purpose*
Bytes 1-5	Preamble byte	Activates PICAXE to start receiving
Byte 6	Start of message byte	Denotes start of data transmission
Byte 7	Transmitter code	Unique identifier for in-field node
Byte 8	Receiver code	Unique identifier for base station
Byte 9	Data byte	Data byte
Byte 10	Cyclic redundancy code	Transmits cyclic redundancy code
Byte 11	End of message byte	Denotes end of data transmission

Table 6.3: Packet format for data transmissions using repetition code

Bytes	*Field Name*	*Purpose*
Bytes 1-5	Preamble byte	Activates PICAXE to start receiving
Byte 6	Start of message byte	Denotes start of data transmission
Byte 7	Transmitter code	Unique identifier for in-field node
Byte 8	Receiver code	Unique identifier for base station
Byte 9	Data byte	Data byte
Byte 10-13	Data byte	Transmits data byte four more times
Byte 14	End of message byte	Denotes end of data transmission

The preamble bytes used were bytes of alternating zeros and ones. If data from an in-field node was not received by the base station, the poll was sent to the same node again, while if data was received then an acknowledgement for that node and the poll for the next node are transmitted in the same message. Transmitting the poll and acknowledgement together effectively halves the number of transmissions by the base station.

6.6 Implementation of Network Protocol

The protocols were implemented in PICAXE software for each node in the network, using hardware illustrated in the block diagram in Figure 6.2 for the in-field node and Figure 6.3 for the base station. The master PICAXE is a PICAXE-40X and the buffer PICAXE is a PICAXE-08M. See Appendix C for reasons why these sizes were chosen.

6.6.1 Design of Circuit to Implement Network Protocol

As shown in Figures 6.2 and 6.3, the major components of each interfacing circuit are the two PICAXE microcontrollers, UHF citizen band radio and modem. The data modulated and demodulated in the interfacing circuit was controlled by the two PICAXE microcontrollers.

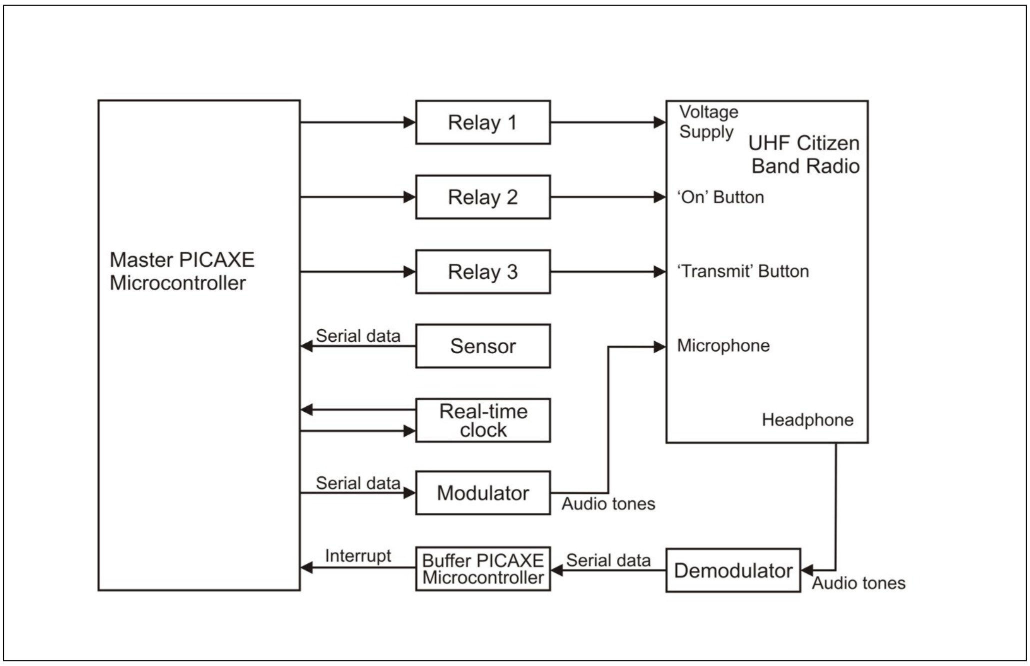


Figure 6.2: Block diagram of in-field node interfacing circuit to implement network protocol

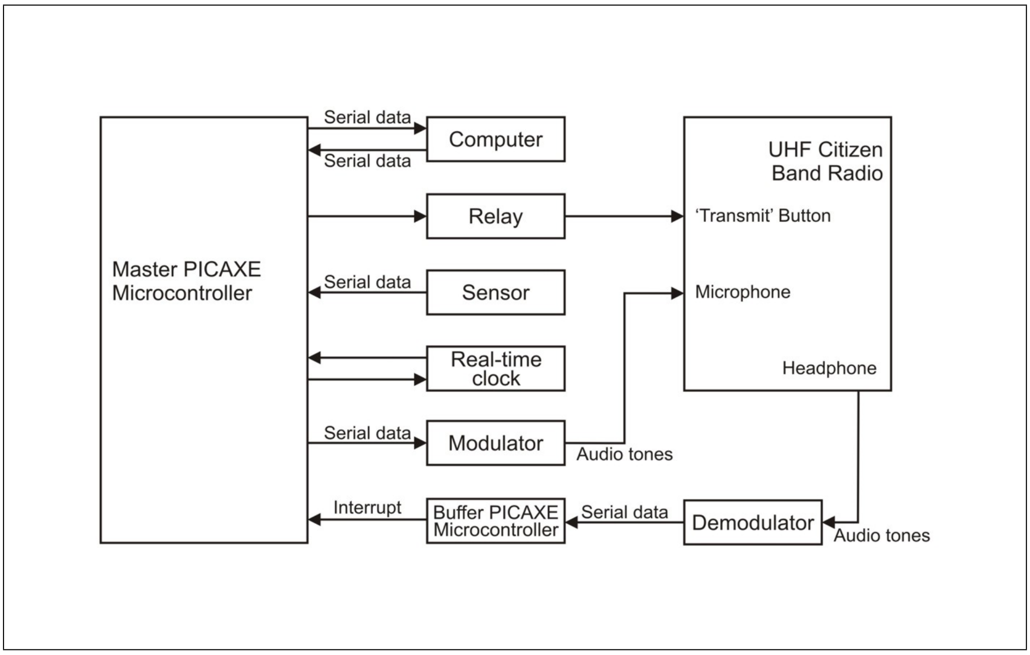


Figure 6.3: Block diagram of base station interfacing circuit to implement network protocol

The PICAXE command used to output serial data to the modulator is 'serout' , while the command that inputs serial data is 'serin'. These commands have the following format:

```
SERIN pin,baudmode,(qualifier,qualifier...),#variable,#variable...
```

```
SEROUT pin,baudmode,(#data,#data...)
```

For both of these commands, ‘pin’ represents the input or output pin to use, while ‘baudmode’ specifies the speed at which the serial data is to be transmitted or received. PICAXE data rates include 600, 1200, 2400 and 4800 baud, however these could potentially be increased by overclocking the PICAXE or by using a different microcontroller.

For a polling access method, the base station sends a poll to the node to request data. Therefore, the node must run a ‘serin’ command and wait for the poll. Since the ‘serin’ command cannot be interrupted, the node cannot perform other operations until data is received. Similarly, the base station must wait to receive data after a poll is sent, but if it does not receive data as expected, then the base station microcontroller would fall into an endless loop waiting for data.

Two methods of overcoming this limitation were (a) adding a series of high bits to the start of each message sent to activate an interrupt on the master PICAXE; or (b) using a buffer PICAXE microcontroller in the interfacing circuit (as in Figure 6.2) that waits for incoming data and then transfers the data to the master PICAXE. The buffer PICAXE approach (method (b)) was taken, since method (a) would increase the transmission time of every message sent in the network, and hence reduce the throughput of the network.

Using the buffer PICAXE microcontroller approach, the buffer PICAXE would flag an output pin to indicate that data was received. When the state of this output pin changed, the master PICAXE (which has an input connected to the flag output of the buffer PICAXE) would activate an interrupt to transfer data from the buffer PICAXE to the master PICAXE. The master PICAXE would then interpret the received data and send a message if appropriate. A single interrupt is employed by the PICAXE and is set by using `setint %00000010,%00000010` for an interrupt routine that activates when pin 1 is high.

The real-time clock (RTC) in the block diagram enables the node to only turn on during the period in which it expects to be polled, if the polling interval is known.

This also reduces the power consumption of the circuit, since the radio will only be on when expecting a poll. A polling interval of 15 minutes was arbitrarily chosen for the telemetry system.

From Figure 6.3, the base station circuit requires a serial connection between the master PICAXE and a computer so that data received by the base station can be transferred to the computer and displayed on a graphical user interface. The developed graphical user interface is discussed in Section 6.9.

6.6.2 Overview of Base Station and In-field Node PICAXE Software

The protocol implemented at the base station is illustrated in Figure 6.4 for both the master and buffer PICAXES. The process is started when the interrogation interval, specified by a preset variable, is reached (as checked using the real time clock). Each node in the network is polled in order of sensor identification number. A potential variation of this protocol includes the polling order of the sensor nodes being changed according to each sensor's priority level.

The in-field node only transmits data as requested by the base station. The base station's polling process is illustrated in Figure 6.4 and how the node responds to a poll is illustrated in Figure 6.5.

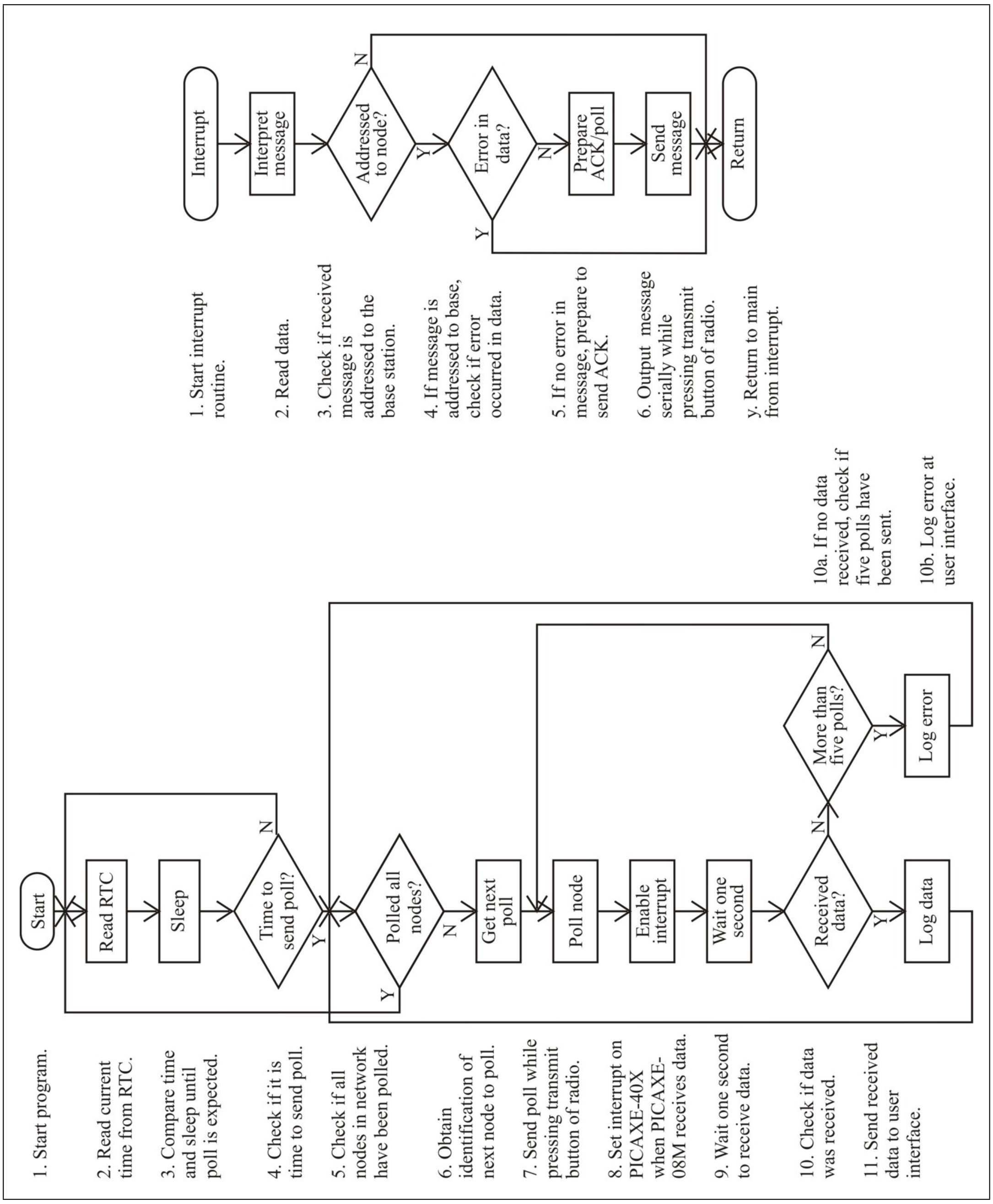


Figure 6.4: Protocol for base station

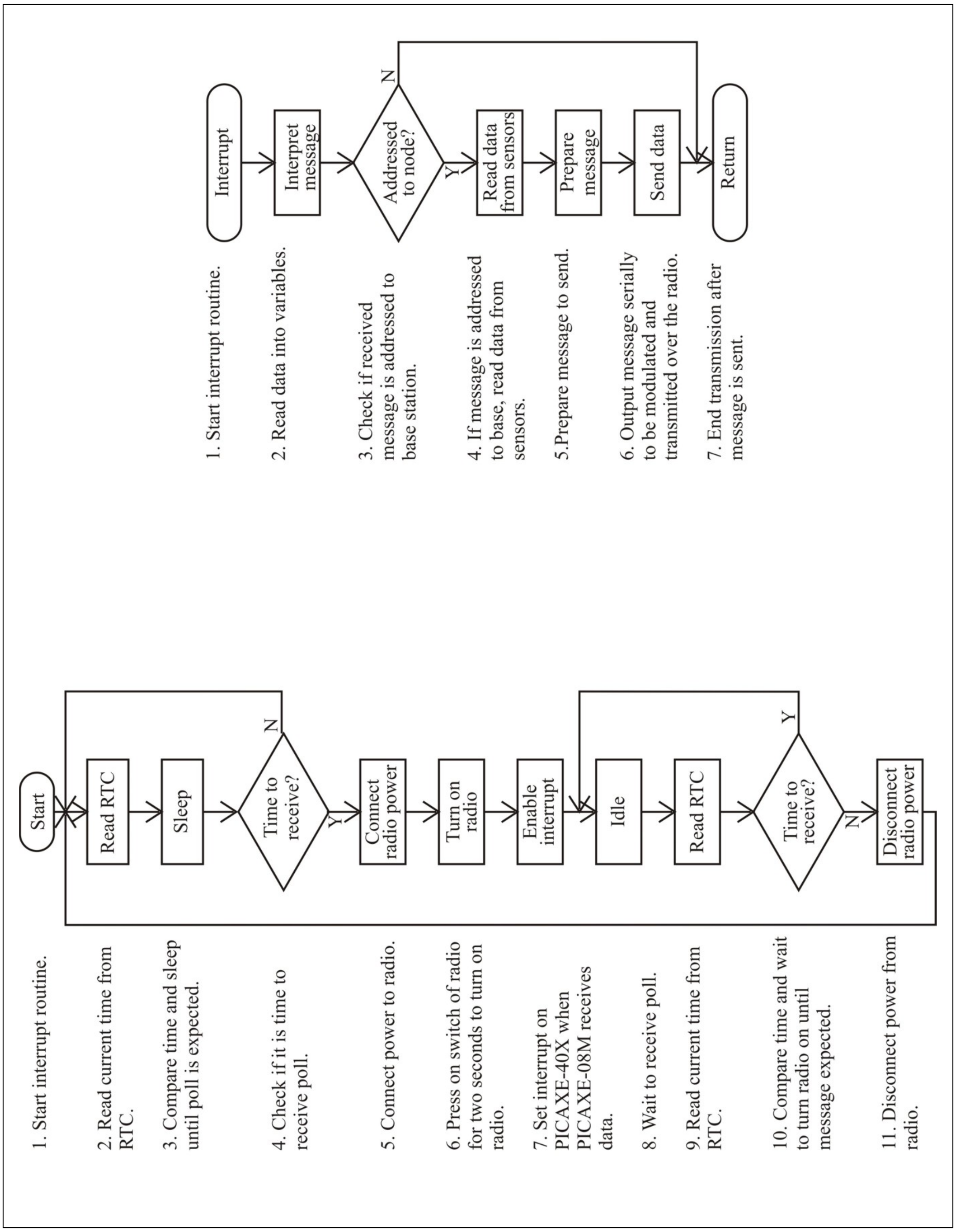


Figure 6.5: Protocol for in-field node

6.7 Construction of Interfacing Circuit

The interfacing circuit (that encodes and decodes data using audio tones over the radio) was built to implement the network protocol designed in Chapter 6. The components used for the circuit are listed with prices in Appendix C. The major components used in the interfacing circuit include two microcontrollers, modulator, demodulator and real-time clock. An additional component used is a step up regulator that is required if the circuit operates from a 6 V supply, since the modulator (XR2206) requires 10 V to operate (as mentioned in Chapter 5). Therefore, a MAX642 12 V step up regulator was used in the base station and node circuits.

The base station and in-field nodes require the same circuitry, except for a serial connection on the base station circuit between the master PICAXE and a computer to transfer data to be displayed on a user interface.

6.7.1 PICAXE Microcontrollers Chosen

Two PICAXES were used at each node, with the requirements of each detailed in Appendix C. Based on these requirements, a PICAXE-40X was chosen to complete the main tasks, while a PICAXE-08M was chosen to buffer the data between the radio and the master PICAXE-40X. A PICAXE-18X satisfied the input and output requirements of the master PICAXE in the system, but due to the developmental nature of the work, the more extensive PICAXE-40X was used.

6.7.2 Circuit Schematic of Interfacing Circuit

With connections as per the block diagram in Figure 6.2, an interfacing circuit was built (see Figure 6.6). See Appendix C for the schematic diagram of the interfacing circuit and Appendix F for the calculations of external component values.

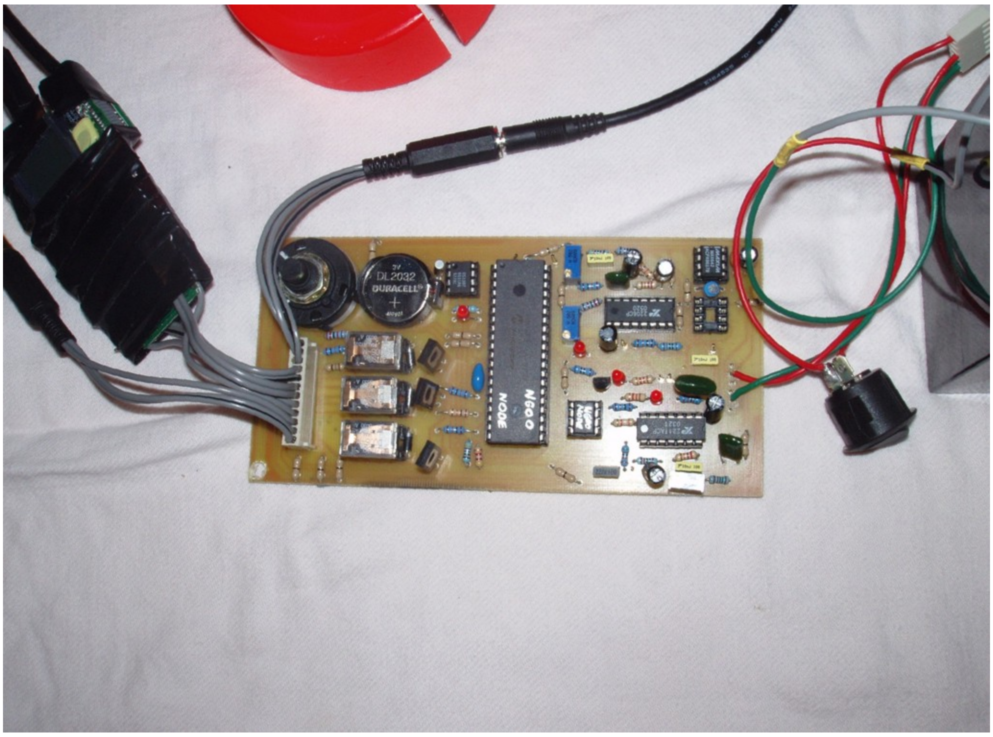


Figure 6.6: Interfacing circuit to transmit and receive data using audio tones

6.8 Workshop Evaluation of Interfacing Circuit

Two printed circuit boards of the interfacing circuit were made for a workshop evaluation of the protocol. The workshop evaluation aimed to test the robustness of the protocol, such as how the network responded when messages were not received or transmitted properly.

6.8.1 Testing the Network Protocol

The steps taken to test the protocol are displayed in Table 6.4. The protocol was able to overcome interruptions on the channel and unresponsive nodes (as reported in Table 6.4). The base station retransmitted the polls up to five times, and when data was not not received the node was interpreted as being offline, despite the nodes not actually receiving the polls. Therefore, it would be beneficial to distinguish between collisions that block the poll from the base station, and a non-responsive node. Before the base station polls the nodes, voices or tones on the channel may be detected using an energy detector to ensure the channel is not busy.

Table 6.4: Results of workshop evaluation of network protocol

Protocol Test	Implemented By	Designed Base Station Action	Actual Base Station Action
Node not receiving	Switching off node	Retransmitted poll	As desired
Node not receiving after five polls	Switching off node	Identified next network node to poll	As desired
Node not transmitting	Switching off node	Retransmitted poll	As desired
Interrupted transmission	Simultaneous or colliding transmissions	Retransmitted poll	As desired

Another three printed circuit boards of the interfacing circuit were then made, so that the network protocol could be tested in the field using five network nodes. See Figure 8.3 for five nodes in field. The field evaluation of the protocol gave identical results to the workshop evaluation (in Table 6.4).

See Appendix D for the the final PICAXE code listing.

6.9 Development of Graphical User Interface

A desktop graphical user interface was developed in Borland Delphi Version 6 to (a) display the status of in-field sensors to the farm manager; and (b) potentially enable the farm manager to obtain data on demand by selecting the next sensor to poll. Screen captures of the various functions of the user interface follow.

The base station interfacing circuit must be connected to the computer running the user interface software to obtain up-to-date in-field data. After opening the user interface application, the serial port that is connected to the base station must be selected and the *Open/close port* button as illustrated in Figure 6.7 must be pressed.

After the serial connection for the user interface is setup, the status of each sensor on the selected field can be displayed when the user passes the mouse pointer over that sensor (see Figure 6.7). The yellow sensors are the advance rate sensors, while the red sensors are the inflow and outflow meters. The field that is displayed can be changed by clicking the drop down box in the top left hand corner and selecting another field number.

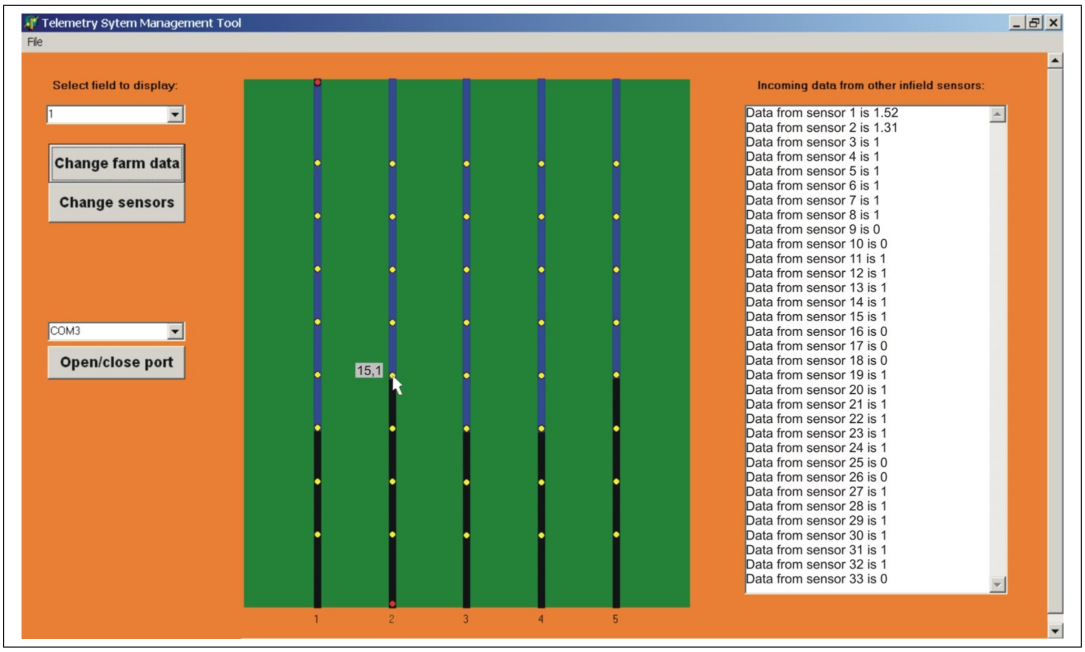


Figure 6.7: User interface displaying status of sensors

The user may edit the database information about the farm by clicking *Change farm data* (see Figure 6.8). This will display all the information in the farm database, including the number of fields on the farm and the dimensions and number of furrows on each farm. Clicking *Add* allows the details of another field to be changed, while clicking *Edit* allows the user to edit the details of existing fields.

The user is also able to change the details of the in-field sensors by clicking *Change sensors* on the main page. This will display the database of in-field sensors details, such as the location of each sensor on each field with respect to the upper left hand corner of the field, as well as the type of sensor. Details of sensors may be added by pressing *Add*, or edited by pressing *Edit* (see Figure 6.9).

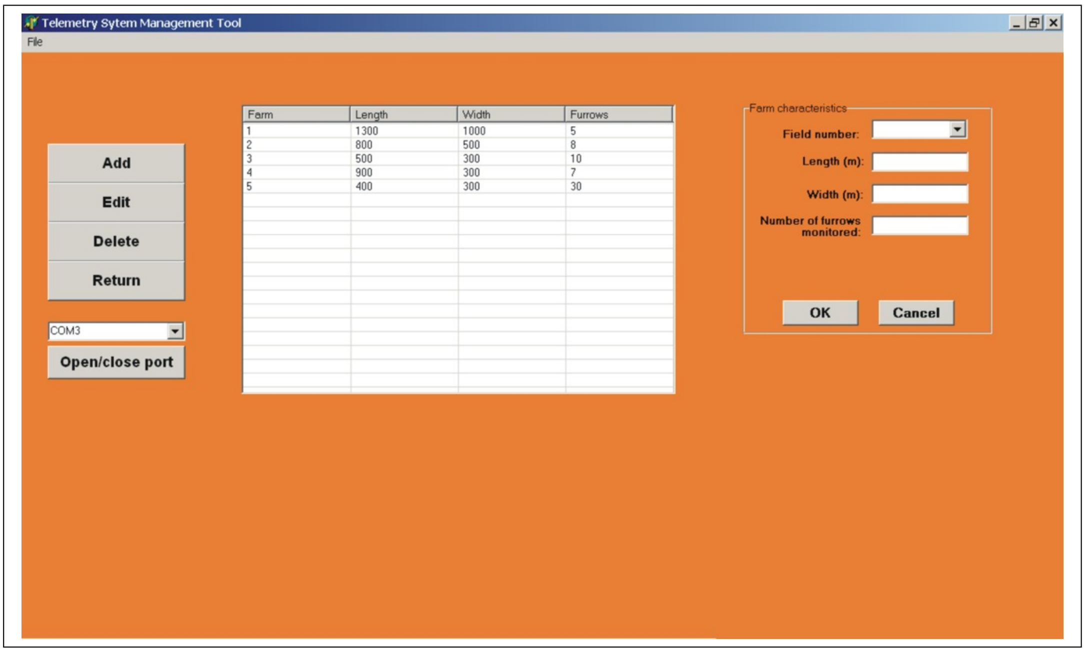


Figure 6.8: Changing farm information via the user interface

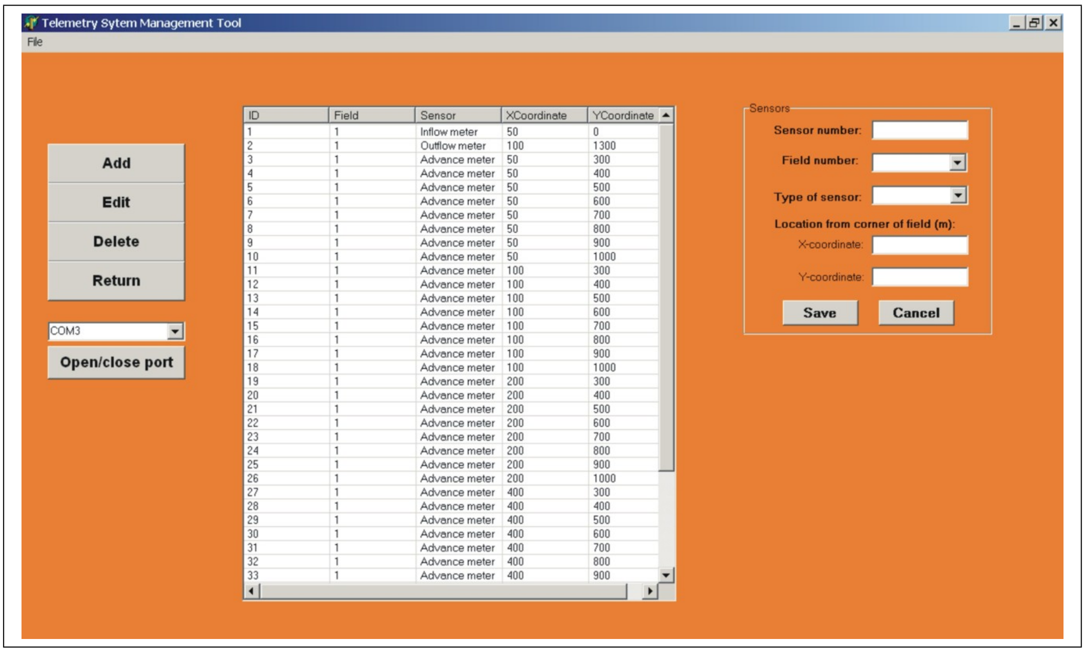


Figure 6.9: Changing in-field sensor information via the user interface

6.10 Comparison of Designed Protocol with OSI Model

The designed network protocol was then compared to the protocols of the Open Systems Interconnection (OSI) network model, which is a set of protocols used to organise and partition the communication process in a network. The model specifies how every layer of the architecture (the seven layers in the OSI model) is to function. These seven layers are the physical, data link, network, transport, session, presentation and application layers as described in Table 6.5. The layer descriptions in the table are from Wall & King (2005) while the third column explains how the citizen band wireless network developed in this project addresses each layer.

Table 6.5: Summary of telemetry system OSI network model layers

Layer	**Description**	**Relevant Component of Developed Network**
Layer 7 (Application)	Provides access to the program requiring the communications data	User interface developed
Layer 6 (Presentation)	Provides independence to process from different data representations	Demodulator decodes audio tones, and modulator encodes serial data
Layer 5 (Session)	Provides communications between applications and managers connections	In-field nodes are polled every 15 minutes
Layer 4 (Transport)	Provides reliable transfer of data between points	Cyclic redundancy and repetition codes were investigated
Layer 3 (Network)	Responsible for routing information through the network	Network has star topology and uses polling method to access sensor nodes
Layer 2 (Data Link)	Converts data into form suitable for transmission over network	A network protocol was designed featuring acknowledgements and sensor identification bytes
Layer 1 (Physical)	Physical medium that connects network nodes	UHF citizen band radio

6.11 Conclusions

The UHF citizen band radio transmission medium enabled a flexible network protocol to be implemented. The developed wireless network featured a polling medium access method, star topology, error handling using repetition and cyclic redundancy codes and robust protocol.

A designed network protocol was able to overcome collisions on the citizen band radio channel since the base station would poll the node up to five times before polling the next node in the network. The network protocol used was also energy efficient, since the UHF citizen band radios only turned on when they were scheduled to be polled.

The protocols were implemented using two PICAXE microcontrollers at each node, and the modulation hardware investigated in Chapter 5. The data received by the base station were transferred to a computer running a graphical user interface, so that the status of in-field sensors could be displayed to the farm manager.

Chapter 7

Construction of Nodes for Field Evaluations

Each network node consists of a modem, microcontroller, UHF citizen band radio, antenna, batteries and other external components. Additional circuitry may also be added to the circuits of these nodes to allow more flexible operation, such as an interface with transceivers from another wireless network.

7.1 UHF Citizen Band Radios

The UHF citizen band radios to be used in the telemetry system were chosen based on the the capabilities of the radios and the results of the in-field evaluation (see Chapter 8). UHF citizen band radios with power output from 0.5 W to 5 W (full legal power output) are available, in both handheld and mobile packages. A mobile radio is physically larger, more expensive and generally has a larger transmission range than a handheld radio.

The radio used at each of the in-field nodes should be physically small and inexpensive (since many would be required in the field), and therefore a handheld radio was chosen for the in-field nodes. A mobile radio is more appropriate than a handheld radio at the base station to ensure maximum transmission distance and to ensure only one base station radio is required per network. Specifications of the handheld and mobile radios used in the evaluation are in Table 7.1.

Table 7.1: Comparison of UHF citizen band radios to use in system

Brand	**Model**	**Package**	**Power Output**	**Voltage Supply**	**Line-of-sight Range**	**Cost**
Dick Smith Electronics	D1800	Handheld	0.5 W	6 V	3 km	$25
Digitech	DC-1040	Handheld	1.5 W	6 V	8 km	$70
Omdi UHF	OM-477CB	Handheld	1 W/5 W	6 V	5 km/15 km	$145
Uniden	UH012sx	Mobile	5 W	12 V	15 km	$279
GME Electrophone	TX4400	Mobile	5 W	12 V	15 km	$429

The three handheld radios tested are shown in Figure 7.1, while the two mobile radios tested are shown in Figure 7.2 and Figure 7.3. The node radio would be chosen based on the results of the in-field evaluation. The base station radio used in early tests when the base station did not send acknowledgements, was the Uniden UH012sx mobile radio. However, the UH012sx would not transmit on channels 22 or 23, which are the only legal citizen band telemetry channels. Therefore, the GME Electrophone TX4400 which was able to transmit on the telemetry channels was the mobile radio chosen to use in the telemetry system.

For a commercial telemetry system, the UHF citizen band radio and interfacing circuit would be made on a single circuit. The transceiver section of the circuit would only have the features required, i.e. without a screen or voice operated transmission. The radios tested for the telemetry system were controlled by and attached to the constructed interfacing circuit by dismantling each radio and connecting wires directly to the voltage supply, ground, on switch and transmit switch of the radio (see Figure C.5 and Figure C.6 in Appendix C).

Figure 7.1: The handheld radios used at the in-field nodes, in order from left to right, Dick Smith Electronics, Digitech and OMDI UHF radios

Figure 7.2: Uniden UH012sx base station radio

Figure 7.3: GME Electrophone TX4400 base station radio

7.1.1 Extending Communication Range

A full-duplex setting may be used on UHF citizen band radios which allows the radios to be used as repeaters (Rad 2006). For example, if a radio set on the full-duplex setting receives signals on channel 31, it will simultaneously re-transmit the signals on channel one. However, the only citizen band channels that data can legally be transmitted on are 22 and 23 and this duplex setting is not useful. A radio may be used as a repeater if the radio receives a message that commands it to retransmit the received data on the same channel.

7.1.2 Interference from Other UHF Citizen Band Users

UHF citizen band radio is an open channel, and transmissions from radios within about 10 km of the in-field telemetry system may collide with data transmissions of the telemetry system. These collisions would cause the error handling scheme to repeatedly send data, which potentially reduces the throughput of the system and increases the number of transmissions and therefore power consumed by the nodes.

Telemetry systems of nearby farms that also use UHF citizen band radios may cause the data to be received by the base station of other farms, or transmitted to the nodes of other farms. Therefore, a method is required to avoid interference from other citizen band users and to provide data privacy so that multiple networks can co-exist within the same area.

Existing telemetry systems such as the Terracept NICTOR™ ensure data privacy through data encryption and network security protocols (Ter 2006). For a UHF citizen band radio, interference may be reduced by (a) using radios that are able to scramble the data in the transmission; (b) using radios that feature a Continuous Tone Coded Squelch System (CTCSS); or (c) adding extra data in the transmission that is unique to the network the node is in.

UHF citizen band radios with a scrambler function (such as the Digitech DC-1028) alter the audio signal to be incomprehensible if received by a radio without the scrambler function, but is received without distortion by a radio with the same scrambler function. The scrambler function would be useful for data privacy from other telemetry systems using the same channel, where all the radios in each telemetry system are set to a different scrambler setting. The demodulator would be unable to interpret the signals from a radio with another scrambler setting.

CTCSS is similar in purpose to the scrambler function; however, it divides each of the 40 channels of the UHF citizen band into 38 channels. A radio set to a CTCSS code (or sub-channel) will only receive transmissions from radios on the same channel and with the same CTCSS code selected. Therefore, all the radios in a network would be set to the same CTCSS code, and the radio telemetry systems of nearby farms would be set to a different CTCSS code.

The scrambler and CTCSS features of citizen band radios would both effectively block out unwanted transmissions from everyday UHF citizen band users. Further data privacy would be provided if an extra byte of data were transmitted in the message to identify the telemetry system network that the node is part of. Each UHF citizen band network would be assigned a different identification number, and nodes of the network would only receive messages containing the same network identification.

7.2 Antenna

The handheld radios evaluated were supplied with no gain antennae, and so external antennae were used. An omnidirectional UHF Mopole antenna from RFIndustries (product number CD51-68-50), with a two signal gain of 4 dB, was tested with the radios to maximise the transmission distance. The antenna was tuned to transmit and receive signals in the 477 MHz band, which is the band centre of channels 22 (476.950 Hz) and 23 (476.975 Hz), by trimming the top to a length of 269 mm (see Figure 7.4).

Antennae were also installed on the base station UHF citizen band radio (GME TX4400) and the 0.5 W sensor node radio (Dick Smith Electronics), since the radios were supplied with no gain antennae. The antennae used were omnidirectional, and so transmit

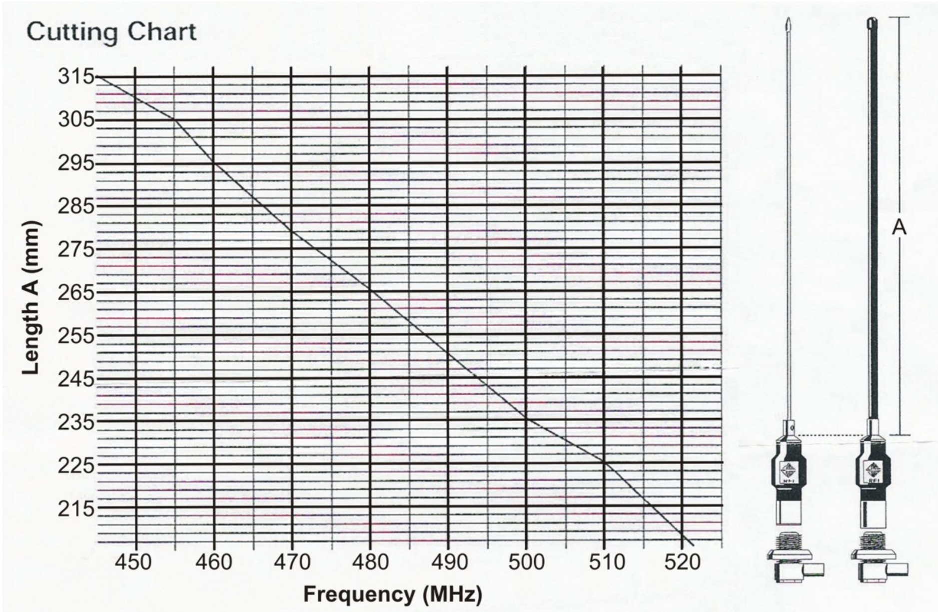


Figure 7.4: Tuning the CD51-68-51 RFIndustries antenna

Source: Ant (1995)

and receive at a power that radiates uniformly in all directions. If the antennae of nodes and base station were positioned carefully, directional antennae that transmit and receive at maximum power in a particular direction may be used instead.

7.3 Integrating Advance Rate Sensors with Nodes

Testing of transmissions can be achieved using artificially synthesised sensor data. However, the use of actual agricultural sensors in the evaluation would enable an interface between the sensors and the node hardware to be tested. Advance rate sensors, similar to those used in the Irrimate^{TM} system, would be the simplest sensors to develop for the system evaluation as they consist of only a pair of wire ends to ascertain the presence of water.

In the advance rate sensors of the Irrimate^{TM} system, a pair of gold plated wire ends form a short circuit when water connects the gold tips. Water can also be detected using a change in capacitance of a pair of wire ends, a method used by Turnell et al. (1997) to determine the advance rate of a field. A short circuit was a less complex method of water detection than a change in capacitance, so was the method of choice. Preliminary tests were conducted with a pair of wire ends connected to the input of a PICAXE as illustrated in Figure 7.5.

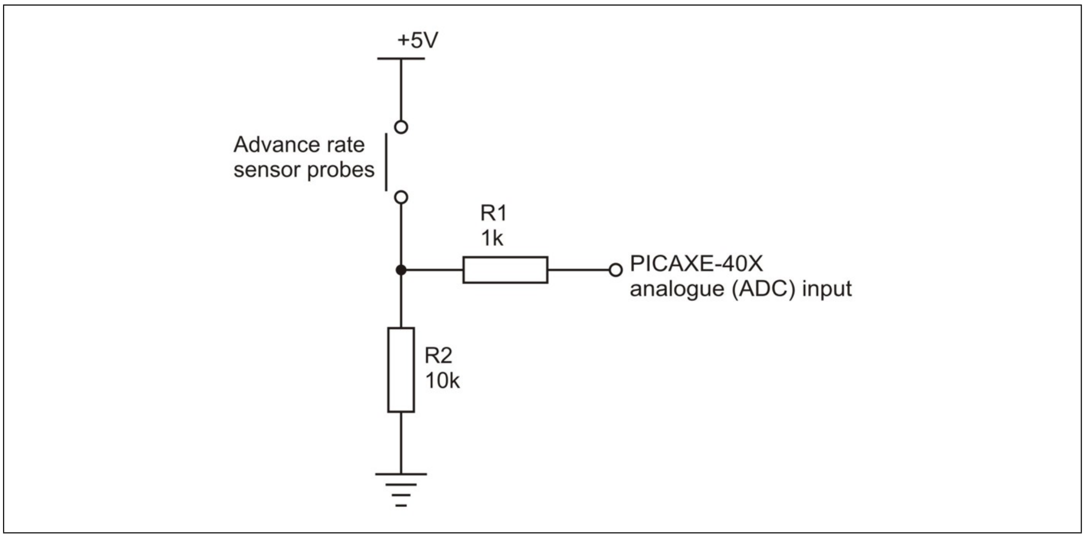


Figure 7.5: Schematic of advance rate sensor probes connected to PICAXE input

In these preliminary tests, a pair of wire ends was placed in a cup of water, and a short circuit was formed upon both wire ends touching the water. The short circuit was detected by an analogue input of the PICAXE instead of a digital input so the sensitivity of the input could be adjusted. However, the wire ends corroded and ceased to form a short circuit within five occurrences of wetting the ends. A pair of steel nails was next used to detect the water (see Figure 7.6), but corroded after about ten tests.

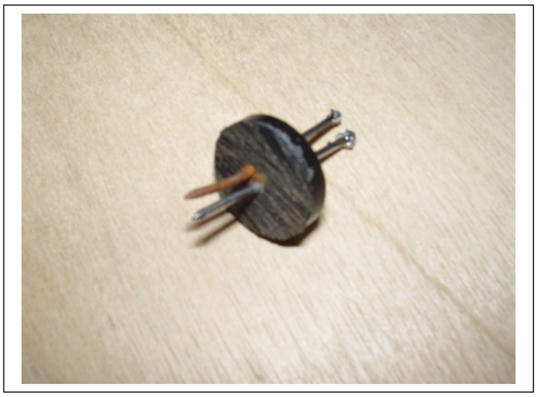


Figure 7.6: Water detector using nails

Figure 7.7: Water detector using PCB pattern

A water detector was then made using a printed circuit board that was etched to feature two closely-woven, but not touching, copper track patterns (see Figure 7.7). This etched pattern had not corroded during the testing period of about fifty tests. Therefore, the etched pattern water detector was used as the water detector in the advance rate sensor for the evaluation of the telemetry system.

7.4 Container for In-field Hardware

The hardware required at each sensor node must be housed in a rugged container that is able to withstand damage from on-farm machinery and weather (such as rain, extreme heat and sun exposure). The desirable characteristics of the container are (a) rugged (if driven over by farm machinery such as tractors); (b) weatherproof (extreme hot or cold, rain); and (c) small footprint.

The container used during the testing of the system was made from plumbing pipes and fittings that were painted red so that the container was visible in the field and could be easily found (see Figure 7.8).

The pipes in use perhaps would not be able to withstand being driven over by a tractor, so an alternative design could be a fibreglass container mounted on a flexible arm that deflects if driven over by a tractor. However, for an actual telemetry system, the sensors would not supply any useful information before sowing or after harvesting, and may therefore be removed during these times of the season.

7.5 Economic Comparison

The price to construct the node was about \$119 retail (see Appendix C for details). In comparison, the XBee Zigbee Transceiver costs about \$90 retail, without the micro-processor, container or other external components of the circuit; a node of the GME Telemetry System costs \$1000; and a Bluetooth device with a bipole antenna has a retail cost of about \$144.

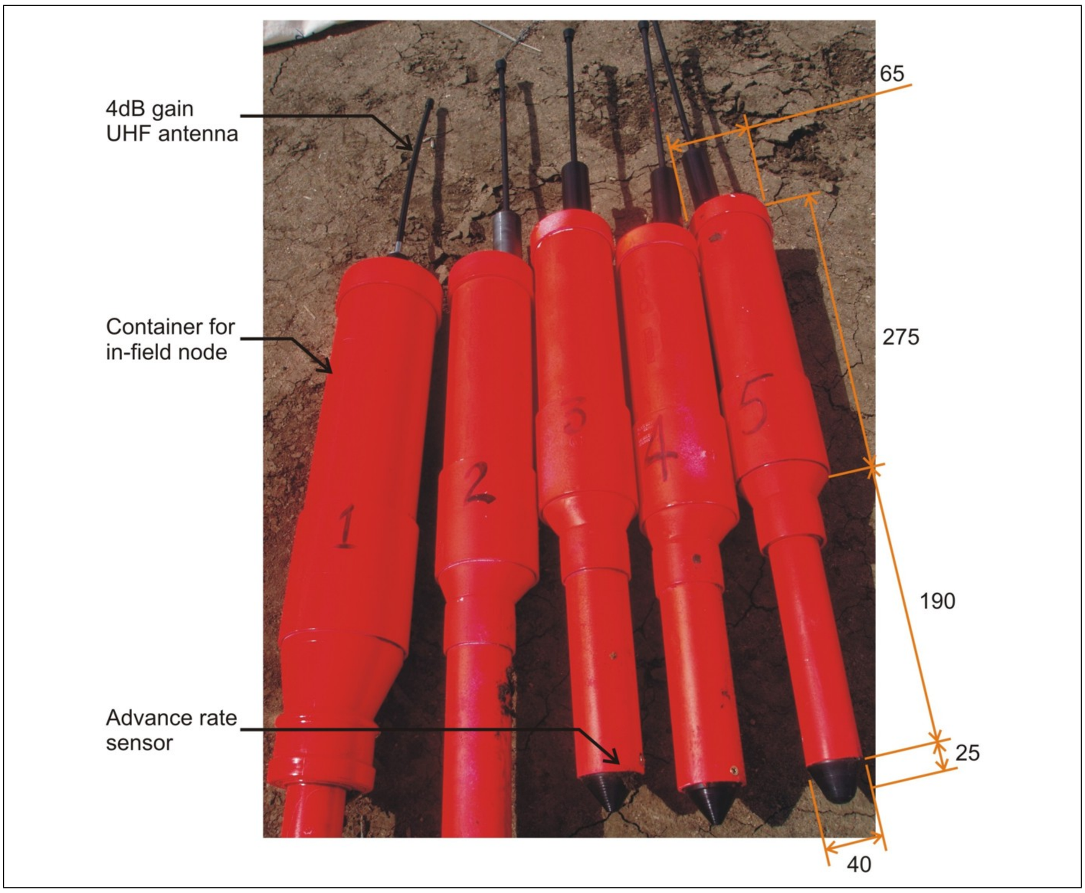


Figure 7.8: Dimensions of container for in-field node

7.6 Circuit Voltage Requirements and Electrical Current Usage

The management of power in a wireless telemetry system is crucial to maximise the life of the sensor nodes in the field, i.e. the telemetry system being developed should have a field life of up to three years. There has been considerable research into increasing the energy efficiency of sensor nodes (Mainwaring et al. (2002), Szewczyk, Mainwaring & Culler (2004) and Ye, Heidemann & Estrin (2004)), such as the development of medium access methods that prevent collisions from occurring, and thus reduce the power wasted on retransmitting data. However, literature research has also acknowledged the difficulty of obtaining a practical solution.

The voltage requirements and electrical current usage of the node hardware were determined to evaluate the energy efficiency of the hardware, as well as to investigate the battery recharging requirements during the lifetime of the node.

7.6.1 Voltage Requirements of Node Hardware

The minimum voltage required by the node hardware to function was investigated. The node hardware consisted of the interfacing circuit and the 0.5 W Dick Smith Electronics UHF citizen band radio. The hardware was powered by four 2500 mAh Nickel Metal-Hydride (NiMH) batteries which typically supplied 4.8 V, but supplied 5.52 V when fully charged. A step up regulator was used in the circuit to supply the XR2206 modulator with 12 V (requires 10 V to operate from the data sheet in Appendix G). Under load, the step up regulator converted the supplied voltage to 7 V and not 10 V, which was still sufficient to operate the modulator.

The UHF citizen band radio was turned on and was not receiving or transmitting as the node hardware was left running with the Nickel Metal-Hydride batteries connected. The first component to stop operating was the UHF citizen band radio, when the battery voltage was 4.97 V (see Figure 7.9).

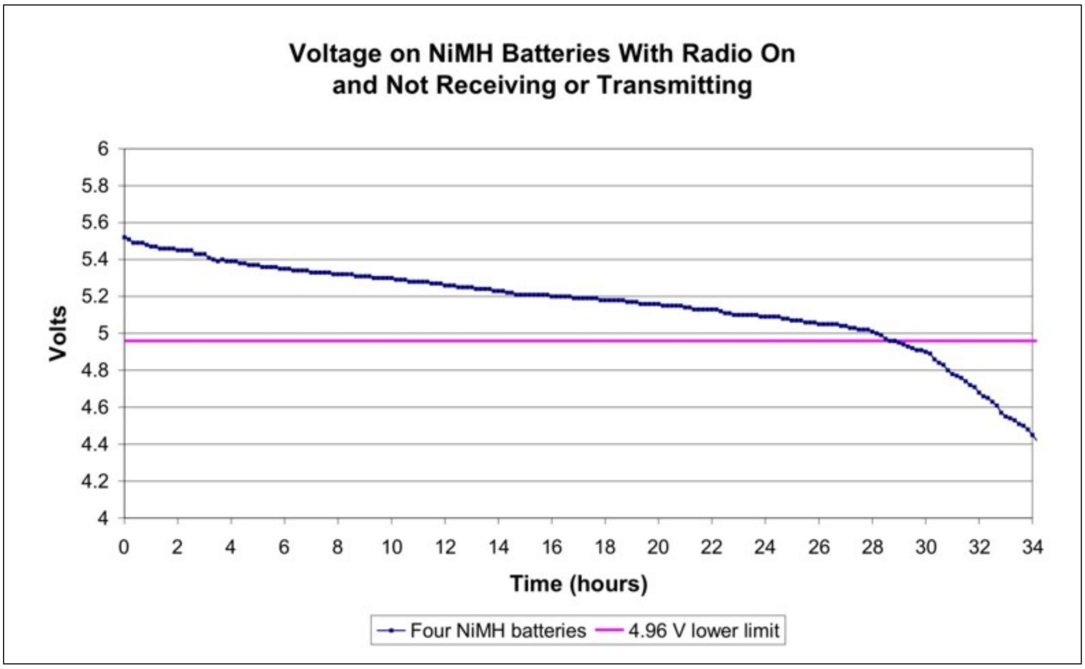


Figure 7.9: Lifetime of four 2500 mAh Nickel Metal-Hydride batteries connected to circuit with radio on

Therefore the battery voltage must be greater than 5 V for the node hardware to transmit or receive data over the UHF citizen band radio. The electrical current usage of the node hardware was next investigated.

7.6.2 Electrical Current Usage of Node Hardware

The electrical current consumption of the node hardware was calculated using the approximate times that the radio would be on and off, where the node only turns on when it is scheduled to be polled (see Table 7.2).

Table 7.2: Electrical current requirements of node hardware using 0.5 W radio during a period of one hour

Activity	**Current Usage**	**Time/hour**
Radio off	74 mA	50 minutes
Radio on and idle	122 mA	∼5 minutes
Radio on and receiving	164 mA	∼5 minutes
Radio on and transmitting	300 mA	300 ms

The calculation of electrical current usage for the node was based on equations from *Technical Information for Choosing Solar Module* (2006). From the calculations in Appendix F the node hardware draws 85.5 mA of current, which is high for a sensor node compared to Maxstream's XBee Zigbee 2.4 GHz module which draws about 45 mA when transmitting and 50 mA when receiving (Zig 2006). Therefore methods to reduce the power consumption of the circuit were investigated.

The highest power consuming component of the node hardware was the step up regulator which was used to step the voltage supply to 12 V for the XR2206 modulator to operate. To reduce the electrical current consumption of the node hardware, the step up regulator was removed from the circuit and the Nickel Metal-Hydride batteries were replaced with a 12 V lead acid battery with a current rating of 1.3 Ah. The lead acid battery was more expensive than the Nickel-Metal Hydride batteries (which cost $15.70 and $13.00, respectively). The new circuit drew 25.8 mA which is 30.2% of the current usage with the step up regulator, and therefore the battery would last approximately

50 hours (see Appendix F for calculations).

The lead acid battery would be recharged in the field by a solar panel to enable long term operation in the field. Solar panels such as the PowerFilm Flexible Thin Film Solar Modules would be appropriate for recharging the batteries in a telemetry system, since they are thin, flexible and are easily attached to the casing of the node. From calculations in Appendix F, an appropriate solar panel is a 12 V that can charge the batteries at 100 mA costing about $50.

The sun exposure of the solar panel affects the rate at which the batteries charge. Therefore, to maximise the charge rate, the solar panel would be placed on top of the hardware container. A solar panel regulator that manages the charging of the batteries would be required with the solar panel.

The current consumption for the node would potentially be increased for 1.5 W and 5 W radios.

7.7 Additional Circuitry for Enhanced Performance

This section discusses additional components that could be added to enhance the performance of the system, including a circuit to stop a continuous transmission over the radio after three seconds; a solar panel and regulator; and a detachable keypad used to input the sensor identification number.

7.7.1 Radio Shutdown Device

Citizen band radios transmitting on either of the data channels (22 or 23) must be fitted with a device that shuts the transmitter down after three minutes of continuous operation, according to restriction (h) of the Australian Media and Communication Authority's class licence for citizen band radio stations. Rather than allowing a continuous transmission of three minutes, a maximum continuous transmission time of three seconds was chosen in accordance with the Citizen Band Radio Stations Class Licence restriction of a three second transmission duty cycle. Appendix E shows a possible circuit design to be added to the existing interfacing circuit that shuts down the radio after three seconds of transmission (see Figure E.1), and the corresponding PICAXE code.

7.7.2 Interface with Other Wireless Technologies

Interfacing a network of UHF citizen band radios with short-range transceivers would be advantageous for a network of close sensors (within 200 m), as suggested by Mark Phythian and discussed in Chapter 4, and as implemented by the commercial Electrosense CB Monitor telemetry system (see Chapter 3). A short-range transceiver currently being implemented by the National Centre for Engineering in Agriculture uses the Maxstream XBee Zigbee Pro modules which operate at 2.4 GHz and transmit at up to 250 kbps with an outdoor range of up to 1 km.

A small network of XBee Zigbee devices could be interfaced to report to a citizen band radio node which then transmits data back to the base station. This would require the data to be received by a Zigbee transmitter, saved by a PICAXE, and then transmitted over the radio back to the base station. See Figure E.2 in Appendix E for a circuit schematic that interfaces PICAXE to Zigbee and the corresponding PICAXE code.

7.7.3 Sensor Identification

The in-field setup of the nodes would potentially be simpler if a switch or keypad was used to specify the sensor identification numbers of the nodes during the setup. This would prevent the need for hard coding the node identifications, and would allow the location and identification number of each node to be noted, and then recorded in a database at the base station. See Appendix E for a possible schematic using a DIP switch (Figure E.6) and the corresponding PICAXE code to read the identification number.

A more user friendly sensor identification input may consist of a detachable keypad and liquid crystal display (LCD) that plugs into each node as the nodes are placed in the field. The LCD would display the numbers entered into the keypad to reduce the keying errors which may otherwise be unnoticed.

Ideally no manual configuration of the node would be required during the setup so that nodes may be added and removed as required. Rather, a scheme for the base station to assign identifiers to the nodes may potentially be implemented.

7.7.4 Sensor Interface

The UHF citizen band radio sensor nodes may be interfaced with sensors other than advance rate sensors using a universal asynchronous receiver transmitter (UART) connection. The master PICAXE of the citizen band radio node would communicate with the data logger of the sensor, in accordance with the communications protocol of the sensors, to obtain the sensor data.

7.7.5 Web Server

Logging the information from the base station directly onto an IP address to be displayed on a website would enable the farm manager to monitor the status of the sensors, from any location via the Internet.

7.8 Hardware and Software Setup of Current Apparatus for Farmer Use

The interfacing board, batteries and radios (hardware) must be setup as follows:

1. Connect the battery and radio to the interfacing circuit of each node.
2. Position sensor nodes in the field.
3. Connect the base station interfacing board to the computer's serial port.

To transfer the sensor data from the PICAXE on the base station circuit to the user interface application on a computer, the computer must have a serial port or have a USB-to-serial adaptor driver installed. The user interface software is setup as follows.

1. Run graphical user interface.
2. Enter details (position and identification number) of network nodes.
3. Enter details of fields (number, size).

7.9 Conclusions

The hardware at each in-field node of the telemetry system required for a field evaluation was developed, and consisted of a UHF citizen band radio, two gain antenna, advance rate sensor and rugged container.

UHF citizen band radios with different output power were trialled at the in-field node, while a 5 W radio was used at the base station for the evaluation. The antenna increases the transmission range of the radio signals, while the development of a water detector for an advance rate sensor allows the interface between the sensors and the node hardware to be tested. A container for the in-field hardware was constructed using plumbing pipes which enabled the radio antenna to be mounted on top of it, and the water detector for the advance rate sensor to be attached at the bottom of the container (to be placed in the ground). Additional circuitry investigated included a radio shutdown device and an interface with other wireless technologies.

Chapter 8

Evaluation of Telemetry System

In-field tests were conducted on the system using circuits that were developed based on the network protocols detailed in Chapter 6 and construction details in Chapter 7. The objective of the field testing was to determine the optimal radio height, radio power output (0.5 W to 5 W), baud rate and audio tone frequencies to maximise radio coverage range and data integrity. In addition, the battery life and the effect on the transmission distance in the presence of water, crop canopy and other environmental conditions were tested.

8.1 Measurement Setup

Each node in the network consisted of a UHF citizen band radio, circuit board and 6 V battery. The 6 V battery used on all the wireless nodes was a regulated 6 V lead acid battery. The UHF citizen band radio used at the base station was the GME Electrophone TX4400 5 W UHF Transceiver (see Figure 8.1 for the setup of the base station in Jondaryan). The setup at each in-field node used the apparatus shown in Figure 8.2. Both the base station and in-field node radios used 4 dB (two gain) UHF antennae.

In both the bare soil and crop canopy environments, samples of approximately 50 transmissions between the two nodes were taken. The base station was situated on the

Figure 8.1: Setup at base station

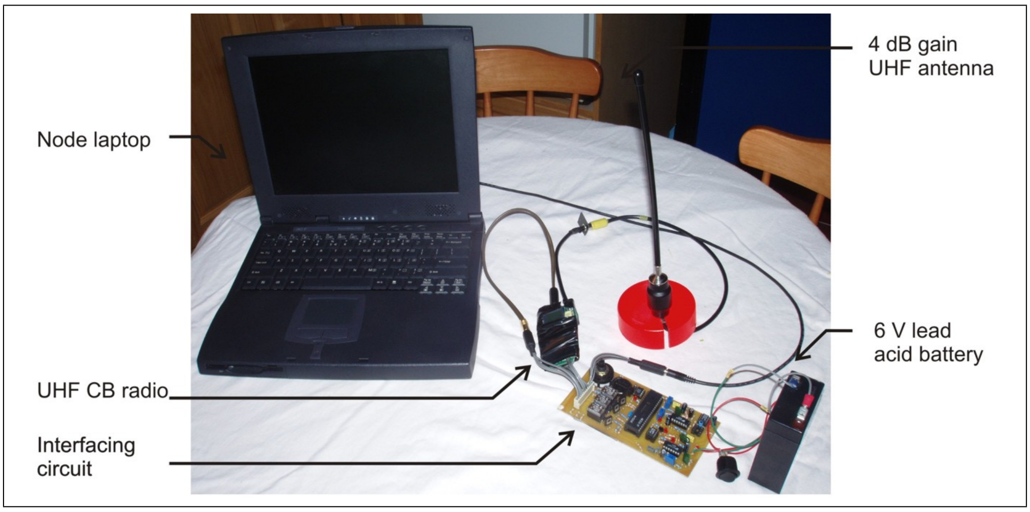


Figure 8.2: Apparatus used for transmission testing of in-field node

ground at the edge of the field, while the in-field node was moved further from the base station down a furrow. The fields on the trial sites were only about 1 km long, so the tests were conducted across more than one field for transmission distances greater than 1 km.

The distance between the in-field node and base station was increased until the signal from the base station was not detected by the radio of the in-field node. At each of the distance intervals, one set of data samples was recorded with the radio antenna tested at different heights above the ground.

The data rate initially used in the bare soil and crop canopy environment tests was 600 baud. This data rate was chosen so that any reliability issues at the lower baud rate could be identified. See Section 8.5 for discussion of upgrading to 1200 baud and results of 1200 baud tests.

The data sent in the in-field tests were random numbers between 0 and 255. Each message transmitted and received was also given a unique identifier to allow direct comparisons between messages transmitted and received.

8.2 Evaluation of System in Bare Soil Environment

The reliability of the system was first evaluated on bare fields with stubble on Jondaryan and Dalby farms (see Figure 8.3 for Dalby test site).

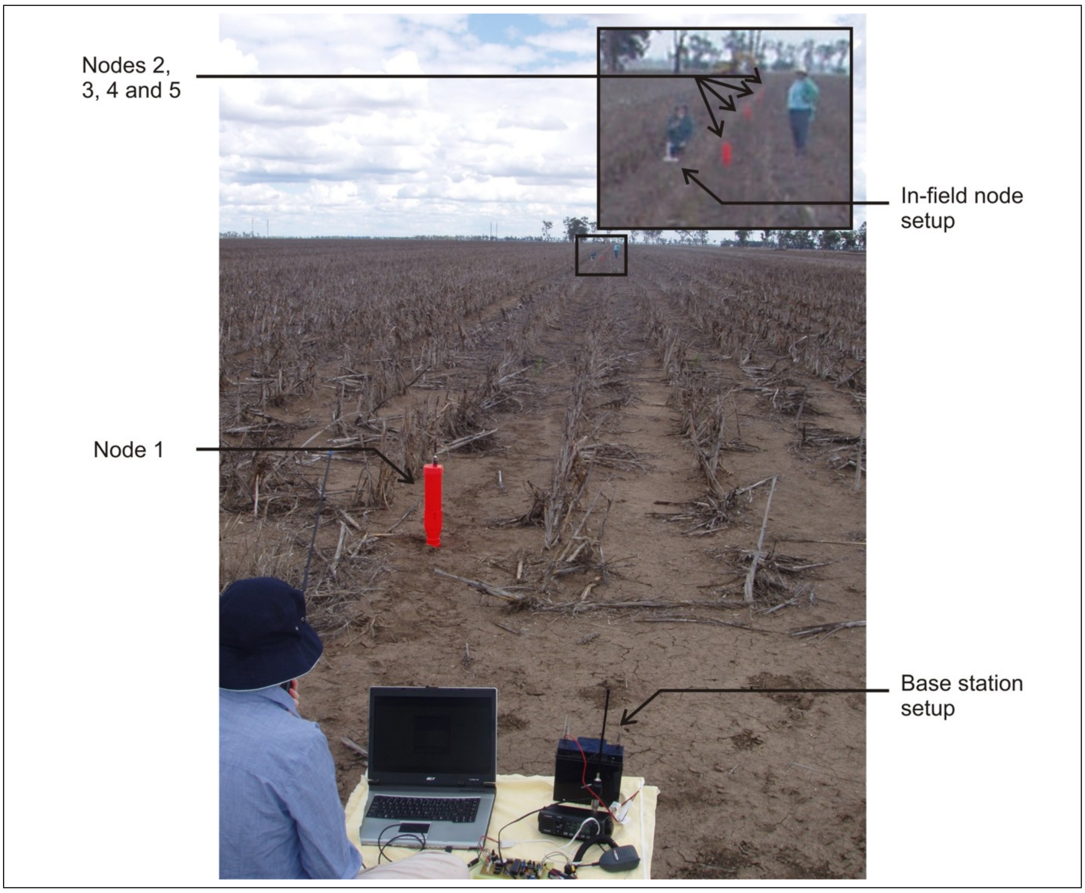


Figure 8.3: Bare soil environment (with stubble) test site at Dalby

The base of the base station radio antenna was on the ground, and the top of the antenna was 0.5 m above the ground. The in-field containers were placed in the ground along the furrows, with the top of the radio antenna 0.5 m above ground level. The bare soil environment was about 20 m away from the farm's water storage (see Figure 8.4), which would potentially attenuate and weaken the radio signals from the UHF citizen band radios.

Figure 8.4: Farm's water storage near stubble test site at Dalby

8.2.1 Message and Data Delivery Rates for 0.5 W Radio in Bare Soil Environment

The 0.5 W UHF citizen band radio (Dick Smith Electronics radio) was set up with the in-field node, and the number of messages transmitted and received by each node was compared (see Figure 8.5). The message delivery rate is the percentage of messages received out of the messages transmitted, and has been classified according to transmission from in-field node to base station, or from base station to in-field node. The data delivery rate is the percentage of sensor data that was received by the base station, disregarding the actual number of transmissions required.

From Figure 8.5, the data from the node was delivered at 600 baud with a 100% success rate for transmission distances of up to 1.5 km, after which the message delivery rate reduced rapidly. The message delivery rate for the node to base transmissions was lower than that for the base to node transmissions due to the lower output power of the in-

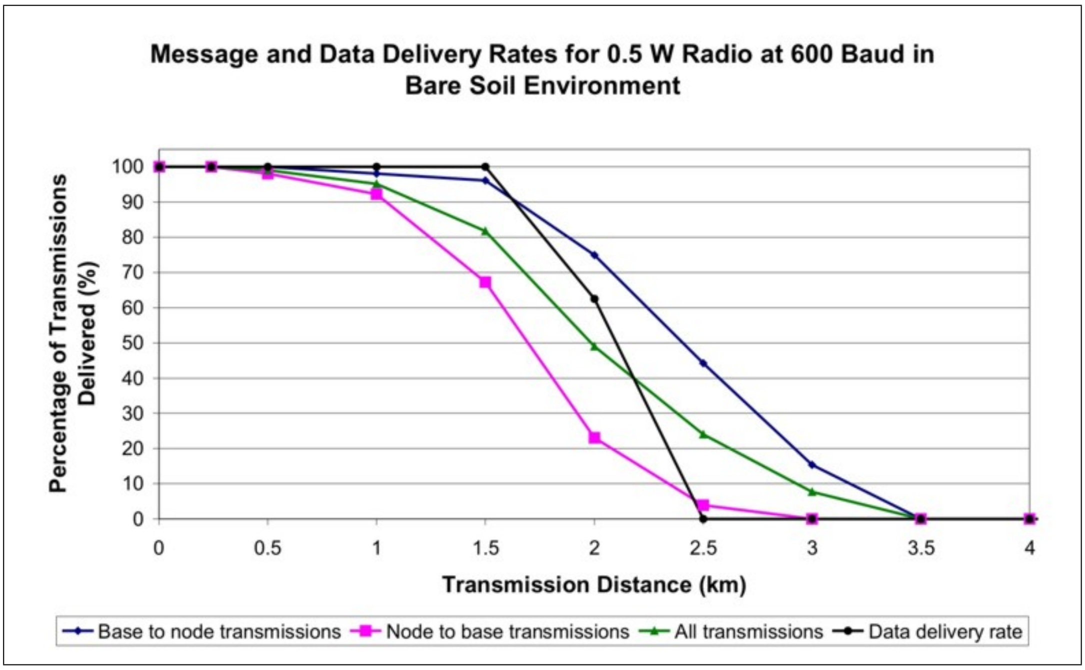


Figure 8.5: Percentage of messages and data delivered in bare soil environment at 600 baud for 0.5 W radio

field radios (0.5 W compared to the base station output power of 5 W). Therefore, to transmit data for 1 km from the node to the base station, the message delivery rate from the node to base must be increased. This may be achieved by using higher power output radios at the in-field nodes.

8.2.2 Message and Data Delivery Rates for Different Output Power Radios

Additional in-field tests were conducted in a bare soil environment using a 1.5 W output radio (Digitech) and a 5 W output radio (Omdi UHF). The antennae used for these two in-field radios were standard no gain UHF antennae (that came with the radio), while the base station and 0.5 W radio used a 4 dB (two gain) antenna. Figure 8.6 compares the successful transmission rate of node to base transmissions for different transmission distances and each of the three in-field radios (0.5, 1.5 and 5 W output). The message delivery rate for only node to base transmissions was graphed due to its poorer performance compared to the message delivery rate of base to node transmissions in the 0.5 W radio tests.

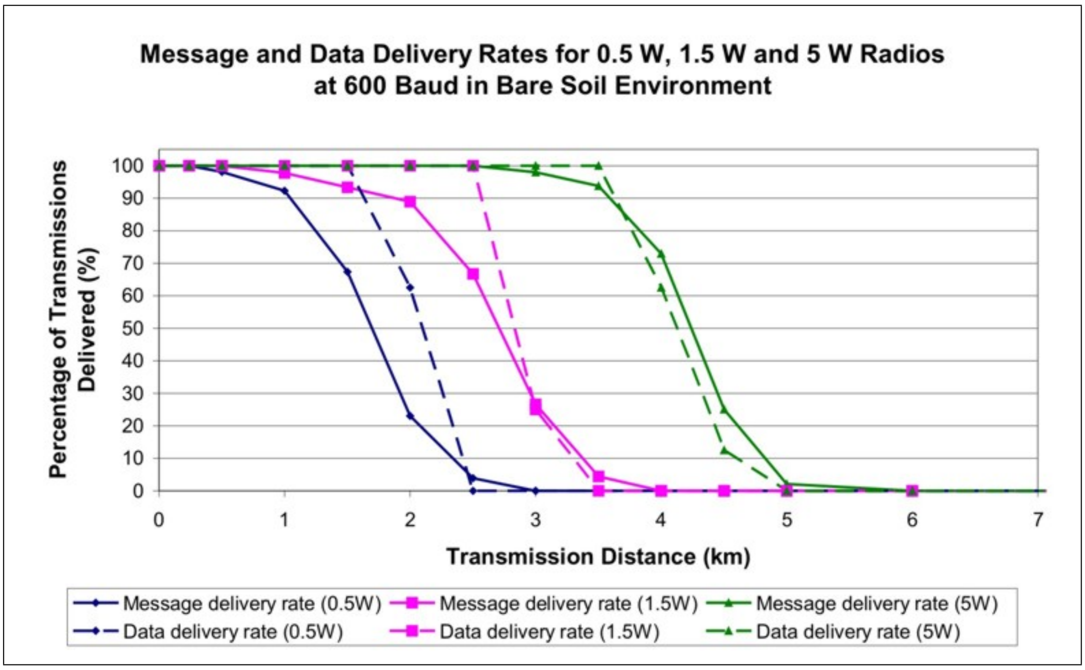


Figure 8.6: Percentage of messages and data delivered by base at 600 baud in a bare soil environment for different output power radios

From Figure 8.6, transmitting at 600 baud using the 1.5 W radio provided a 100% data delivery rate for transmission distances up to 2.5 km. The data delivery rate was greatest for the 5 W radio, with 100% of data being transmitted for up to 3.5 km from the in-field node.

8.2.3 Conclusions of Evaluation in Bare Soil Environment

The evaluation of the system in a bare soil environment, using a 0.5 W radio at the in-field node, identified the node to base transmissions as the limiting factor in the transmission distance achieved since a lower power output radio was used at the node than the base station. The reliable transmission distances achieved by the three output power radios at 600 baud are displayed in Table 8.1. From these results, a 1.5 W radio would be selected as the in-field node radio for transmission at 600 baud since the transmission distance exceeded 2 km.

Table 8.1: Summary of telemetry system performance in bare soil environment

Power Output of Radio	Data Rate	Reliable Transmission Distance
0.5 W	600 baud	1.5 km
1.5 W	600 baud	2.5 km
5 W	600 baud	3.5 km

8.3 Evaluation of System in Crop Canopy Environment

The system was then tested in poor conditions such as in the presence of crop canopy, irrigation water and overhead power lines, to determine the effect that potentially signal-attenuating conditions have on the reliability of the radios. A barley farm in Jondaryan which was located near overhead power lines and underground telephone cables was used as the test site (see Figure 8.7).

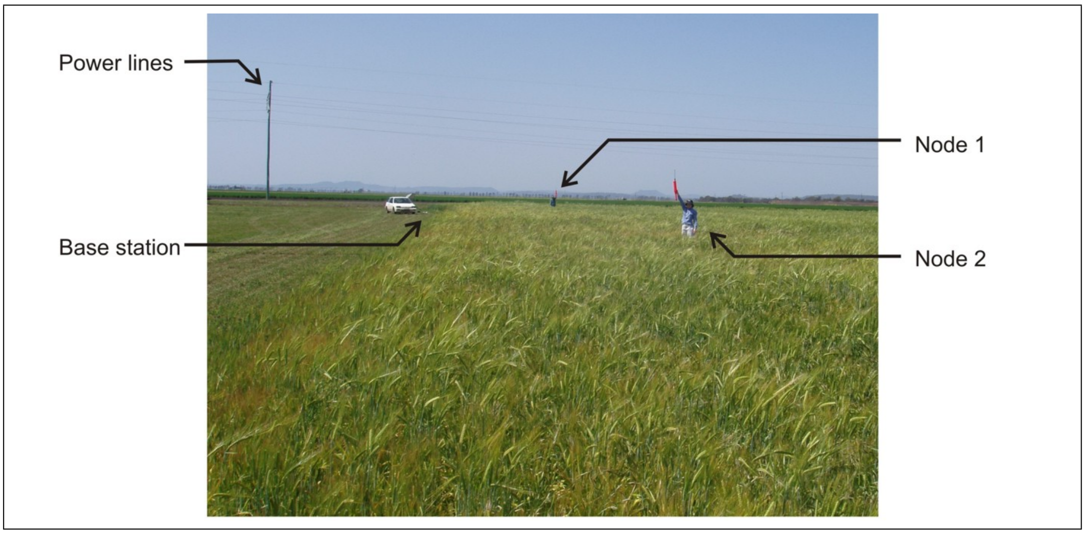


Figure 8.7: Crop canopy environment on Jondaryan barley farm with first two nodes 200 m apart and held above canopy for visual clarity in the photograph

8.3.1 Comparison of Delivery Rates for 0.5 W Radio Through Crop Canopy and Bare Soil Environment

The performance of the 0.5 W radio transmitting at 600 baud in a crop canopy environment was first tested. The base station was placed on the ground at the edge of the field almost directly underneath the overhead power lines and above underground telephone cables (see Figure 8.7 for position of the first two nodes and base station in the field). The in-field node containers were placed in the ground throughout the field, with the top of the antenna 0.5 m from ground level.

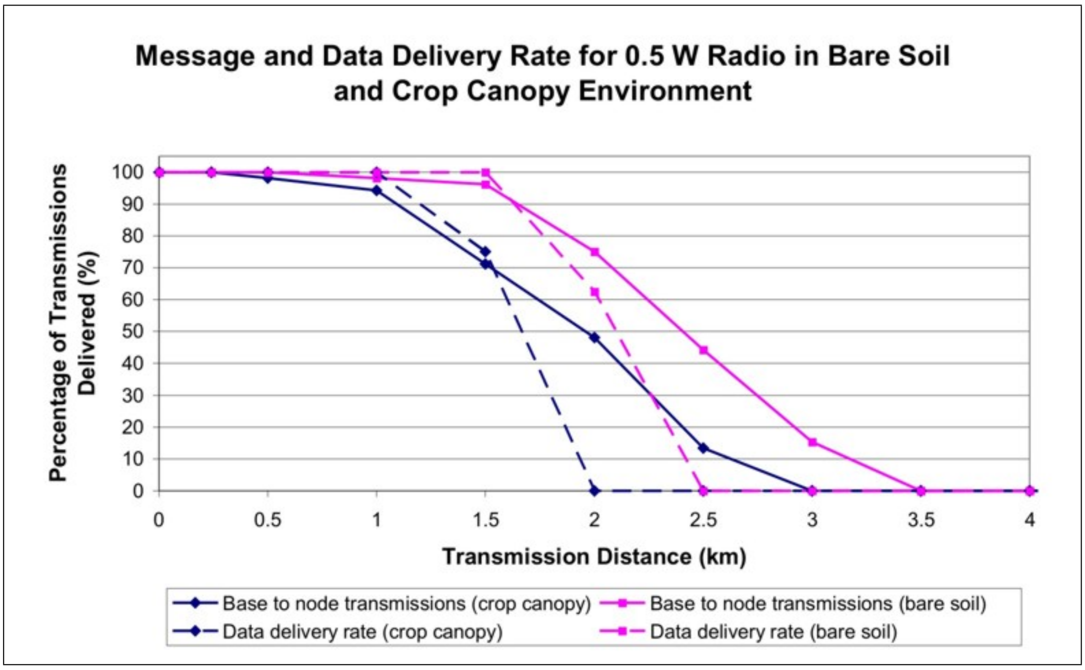


Figure 8.8: Percentage of messages and data delivered for 0.5 W radio at 600 baud in bare soil and crop canopy environment

From Figure 8.8, the presence of crop canopy reduced the transmission distance of the in-field radios by approximately 500 m. Data was received reliably for transmissions at 600 baud from the 0.5 W radio for distances up to 1 km through crop canopy, while the base station received data from the 0.5 W radio at distances up to 1.5 km in a bare soil environment. Therefore, different output power radios were tested through crop canopy to increase the transmission distance to 2 km.

8.3.2 Comparison of Delivery Rates for Different Output Power Radios Through Crop Canopy and Bare Soil Environment

The in-field 0.5 W radio was replaced with 1.5 W and 5 W radios, and subsequently 100% reliability was achieved for a 2 km transmission distance at 600 baud, as shown in Figure 8.9. Data was reliably transmitted 1 km from a 0.5 W radio, 2 km from a

1.5 W radio and 3.5 km from a 5 W radio, at 600 baud through crop canopy. The transmission range of all of the radios was approximately 500 m greater than in a bare soil environment. Therefore, in a crop canopy environmnet a 1.5 W citizen band radio would be sufficient to use.

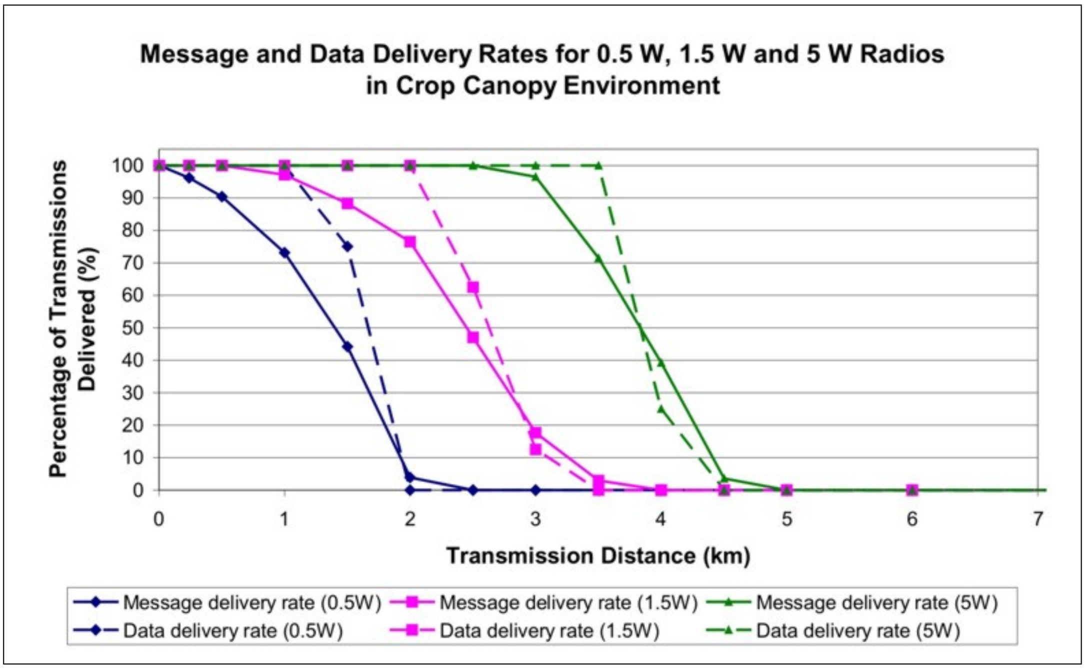


Figure 8.9: Percentage of messages and data delivered for different output power radios at 600 baud in crop canopy environments

8.3.3 Conclusions of Evaluation in Crop Canopy Environment

The data delivery rates for transmissions between different output power radios and the base station at various transmission distances through crop canopy was tested and compared to those in a bare soil environment. The results of the evaluation in crop canopy at 600 baud is shown in Table 8.2. These results indicate that a 1.5 W radio would be selected as the in-field node radio since the transmission distance exceeds 2 km when transmitting at 600 baud. The transmission range in a bare soil environment was approximately 500 m greater than through crop canopy.

Table 8.2: Summary of telemetry system performance in crop canopy environment

Power Output of Radio	Data Rate	Reliable Transmission Distance
0.5 W	600 baud	1 km
1.5 W	600 baud	2 km
5 W	600 baud	3.5 km

8.4 Evaluation of System with Elevated Base Station Antenna

The base station antenna of an actual agricultural telemetry system would not be placed on the ground; rather, it would be mounted on the roof of the farm manager's house or shed to maximise transmission range. Therefore, tests were conducted at 600 and 1200 baud with the base station on top of the farm's water storage, at a height of 4 m above ground level (see Figure 8.10).

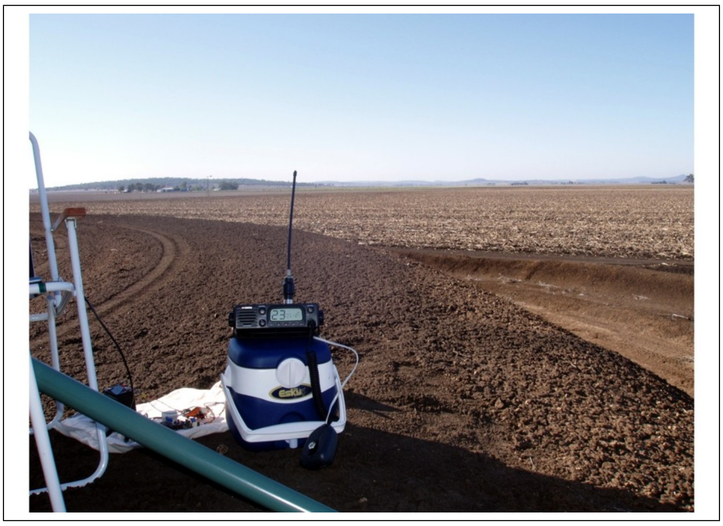


Figure 8.10: Base station with antenna mounted 4 m above fields

8.4.1 Comparison of Delivery Rates for Different Output Power Radios with Base Station Elevated

The message delivery rates for base to node, node to base, and all transmissions for each of the radios when the base station antenna is elevated are compared in Figure 8.11.

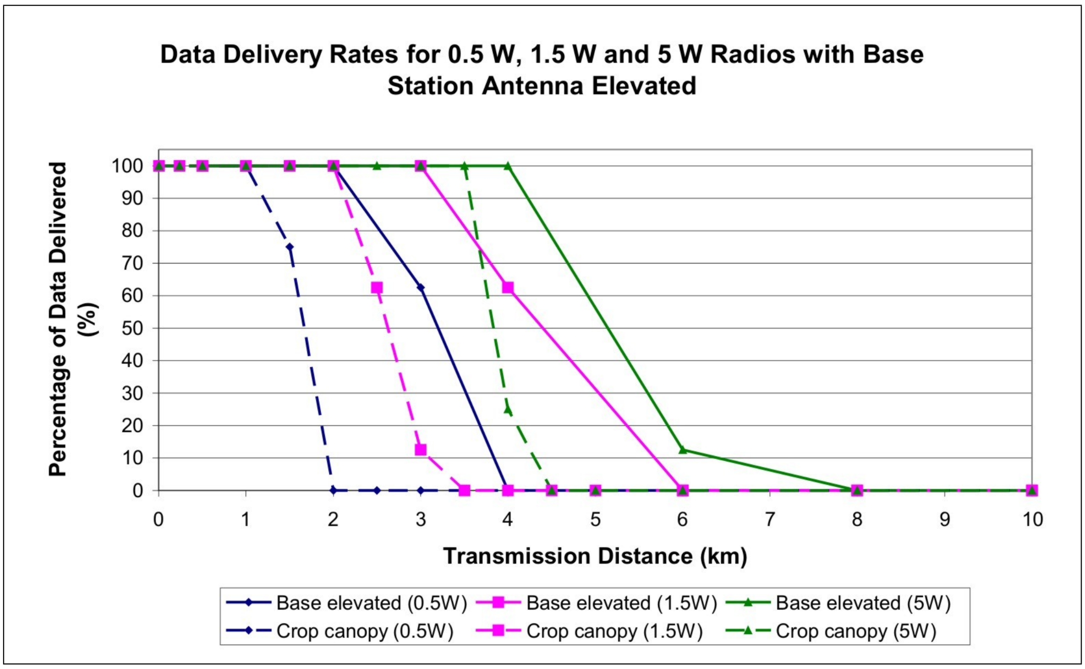


Figure 8.11: Percentage of data delivered for different output power radios through crop canopy and with base station elevated

The plot in Figure 8.11 compares the delivery rates for when the base station antennae are mounted high and low and the in-field nodes are in a field with crop canopy. A summary of the results is displayed in Table 8.3. From these results, a 0.5 W radio was able to transmit with a reliability of 100% at 600 baud and is thus the new recommended UHF citizen band radio to use as the in-field node radios.

8.4.2 Data Delivery Rate for Increased Data Rate in Bare Soil Environment

In-field tests were conducted on the telemetry system using a data rate of 1200 baud with the base station elevated. The new baud rate was programmed in the system by changing a variable in the PICAXE software that defined the baud rate. The frequencies of the tones used to modulate the data were 1200 Hz and 2200 Hz for a logic 0 and 1, respectively. Since the same frequencies were used no other circuit changes were required.

Transmitting data at 1200 baud resulted in low delivery rates of data, with less than two percent of transmissions being successfully received between a node using the 0.5 W radio and the base station 100 m away (as illustrated in Figure 8.12). This low delivery rate also resulted in no successful transmissions of data even after five retransmissions from the node to the base. Therefore, changes must be made to the circuit to increase the data rate, as will be discussed in Section 8.5.

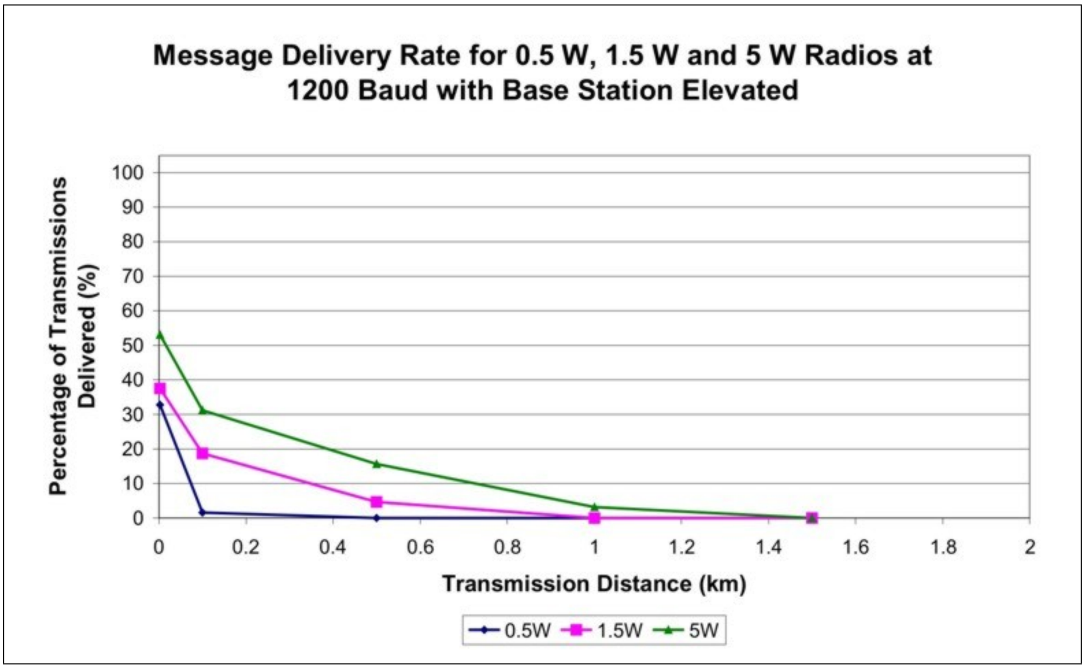


Figure 8.12: Percentage of messages transmitted between node and base in bare soil environment at 1200 baud

8.4.3 Conclusions of Evaluation with Base Station Elevated

The reliability of node to base transmissions and data delivery was increased when the base station antenna was mounted high, and the reliability was greater at larger transmission distances. Further work is required to modify the node hardware for reliable data transmission at 1200 baud, since there were no successful data transmissions using the current hardware in the field evaluation.

Table 8.3: Summary of telemetry system performance with base station elevated

Power Output of Radio	Data Rate	Reliable Transmission Distance
0.5 W	600 baud	2.5 km
1.5 W	600 baud	3.5 km
5 W	600 baud	4 km
1 W, 1.5 W, 5 W	1200 baud	0 km

8.5 Increasing Data Rate

The reduced reliability at 1200 baud can be explained using Shannon's sampling theorem, since the 1200 Hz and 2200 Hz frequencies were not represented accurately when the baud rate was increased from 600 to 1200 baud (and the sampling frequency halved). One method to overcome this is to increase the audio tone frequencies.

Amateur radio modem techniques were investigated to increase the reliability of data transfer over citizen band radio at 1200 baud. Amateur radio modems that operate at baud rates greater than 1200 baud often use audio tones greater than 1200 Hz and 2200 Hz to modulate the data, for example an amateur radio modem circuit using the TCM3105 transmits at 2400 baud using 2165 Hz to represent a logic 0 and 3970 Hz to represent a logic 1. However, Texas Instruments reported a fault in the TCM3105 so this integrated circuit could not be used. The XR2206 modulator and XR2211 demodulator may potentially allow the transfer of data at a greater data rate if frequencies of 2165 Hz and 3970 Hz are used.

Other frequencies for the audio tones may be chosen arbitrarily, but the ability of UHF citizen band radios to transmit higher frequencies is unknown due to no available documentation. Hence, the bandwidth of a UHF citizen band radio was tested to determine the maximum frequency able to be transmitted and decoded successfully over the radio.

8.5.1 Testing Bandwidth of UHF Citizen Band Radio

The UHF citizen band radio was expected to have a bandwidth similar to a telephone line (3000 Hz) since these are both analogue voice communication channels. To test the bandwidth, a range of frequencies was created in Matlab using a sweep from 0 Hz to 20 kHz over a period of five seconds.

The sweep was transmitted using 0.5 W and 5 W citizen band radios located at various distances down a furrow from a fixed base station in a bare soil environment on a Jondaryan farm. The audio volume level of the transmission was identical to the audio volume level of the previous in-field evaluations. The signals were received and recorded by a computer connected to the speaker of the base station radio at a sampling rate of 44.1 kHz. By Shannon's sampling theorem, only frequencies up to half (or practically, up to one quarter) of the sampling frequency can be accurately represented. Therefore only the response for the frequencies between 0 kHz and 10 kHz were considered. The root mean squared of the signal was analysed in software in Matlab to give the plots in Figure 8.13.

There was a d.c. offset of -2.3 mV rms in the received signal (see Figure 8.13(a)), which was most likely caused by the biasing connection of the amplifier stage at the receiver (which was a.c. coupled). The noise in the signal also made interpreting the bandwidth difficult, so a Butterworth filter was applied to smooth the signal. The signal that was offset and filtered is shown in Figure 8.13(b).

The bandwidth of the UHF citizen band radios was determined using the the voltage required by the input of the XR2211 demodulator to cause limiting, which is 2 mV rms to 10 mV rms (from XR2211 data sheet in Appendix G). These voltage levels intercept with the root mean squared-value curve in Figure 8.13(b) at frequencies of 1000 Hz to 4200 Hz. Therefore, data encoded with audio tones up to 4200 Hz could be tested over the radio. Neither the distance between the two radios nor the power output of the radios affected on the bandwidth of the UHF citizen band radios.

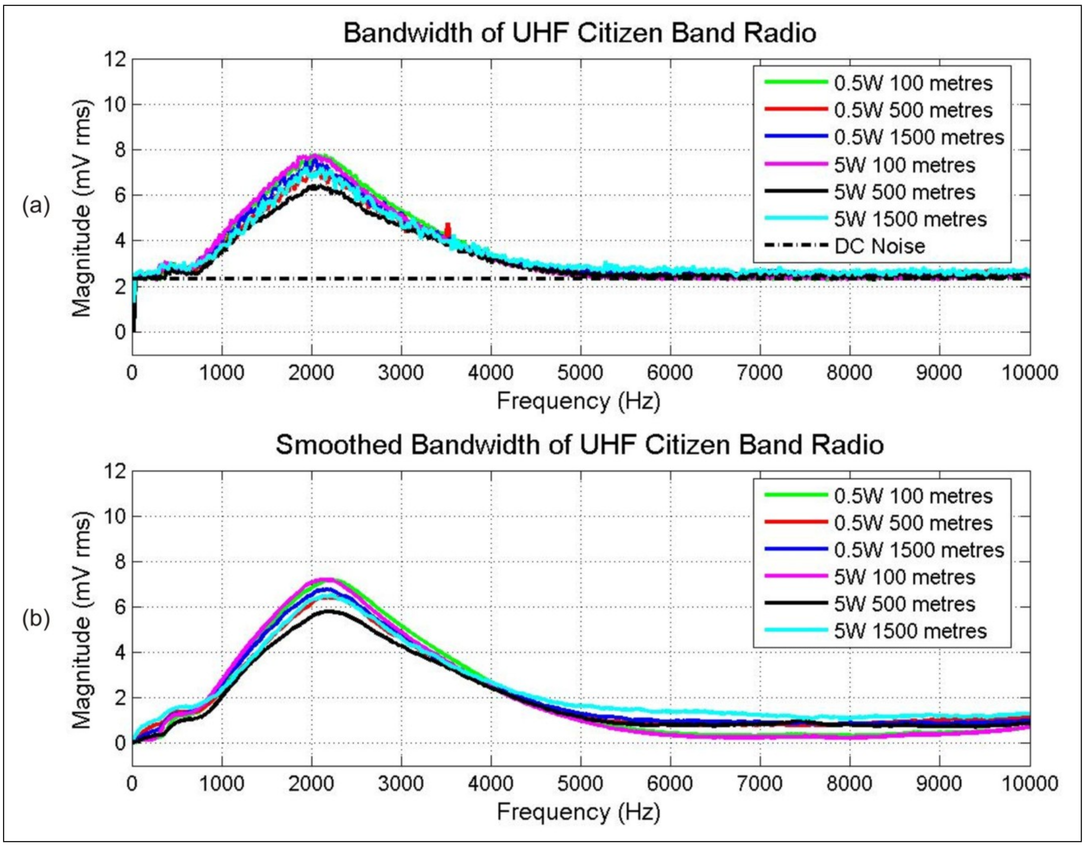


Figure 8.13: Waveform for audio tones swept from 0 Hz to 20 kHz that (a) contains d.c. noise; and (b) has been smoothed and offset

8.5.2 Signal-to-noise Ratio of UHF Citizen Band Radio

Background noise over the radio channel may limit the bandwidth of the radio channel or cause errors during the transmissions. The signal-to-noise (S/N) ratio of the channel is a measure of the amount of noise in the signal by comparing the transmitted signal to the background noise of the channel. Therefore the signal-to-noise ratio was determined for transmissions in the audio tone frequencies used to modulate the data over the radio channel. Data from the bandwidth tests were used as follows:

$$\text{Ratio} = \left(\frac{A_{signal}}{A_{noise}}\right)^2 \qquad \text{and} \qquad \text{Ratio (dB)} \quad = \quad 20\log_{10}\left(\frac{A_{signal}}{A_{noise}}\right)$$

See Table 8.4 for the test results, using $A_{noise} = 0.7 \times 10^{-3}$ for the amplitude of the noise from Figure 8.13(a).

The signal-to-noise ratio for an audio cassette is typically about 60 dB, while for a

Table 8.4: Signal-to-noise ratio of transmitted citizen band radio signal at various frequencies

Audio tone frequency	$\mathbf{A}_{signal}$	**S/N ratio**	**S/N ratio (dB)**
1200 Hz	3.9×10^{-3}	31.04	14.92
2200 Hz	6.5×10^{-3}	86.22	19.36
2600 Hz	6.0×10^{-3}	73.47	18.66
3600 Hz	3.2×10^{-3}	20.90	13.20

CD is almost 100 dB. The signal-to-noise ratio for the UHF citizen band radios was lower than these ratios (between 15 and 20 dB). Further signal-to-noise ratio tests may measure the ratio of useful to false (erroneous) data and the throughput of the network.

The frequency of the audio tones was decided to be 2600 Hz for a logic 0 and 3600 Hz for a logic 1 since these are close in value to the frequencies suggested for 2400 baud transmission using the TCM3105. To change the frequencies of the modulated audio tones, new values of external timing components of the XR2206 modulator and XR2211 demodulator were calculated (see Appendix F for calculations).

8.5.3 Evaluation of System with New Audio Tones with Base Station Elevated

Data transmission at 1200 and 2400 baud were attempted using audio tones at 2600 Hz and 3600 Hz to represent logic 0 and 1, respectively, with the base station elevated. The results of transmission distance tests are shown in Figure 8.14.

From Figure 8.14, the delivery rate of transmissions at 1200 baud were very similar to those at 600 baud. The reliable transmission distance using the 0.5 W radio was 1.5 km, and using the 1.5 W radio was 2 km.

The baud rate was then increased to 2400 Hz using the modulation frequencies as used at 600 baud (2600 Hz and 3600 Hz). No data was successfully delivered when transmitting at 2400 baud. This was because the audio tone frequencies were too small for the demodulator to accurately decode at 2400 baud. Increasing the data rate above 1200 baud was not pursued further due to time constraints.

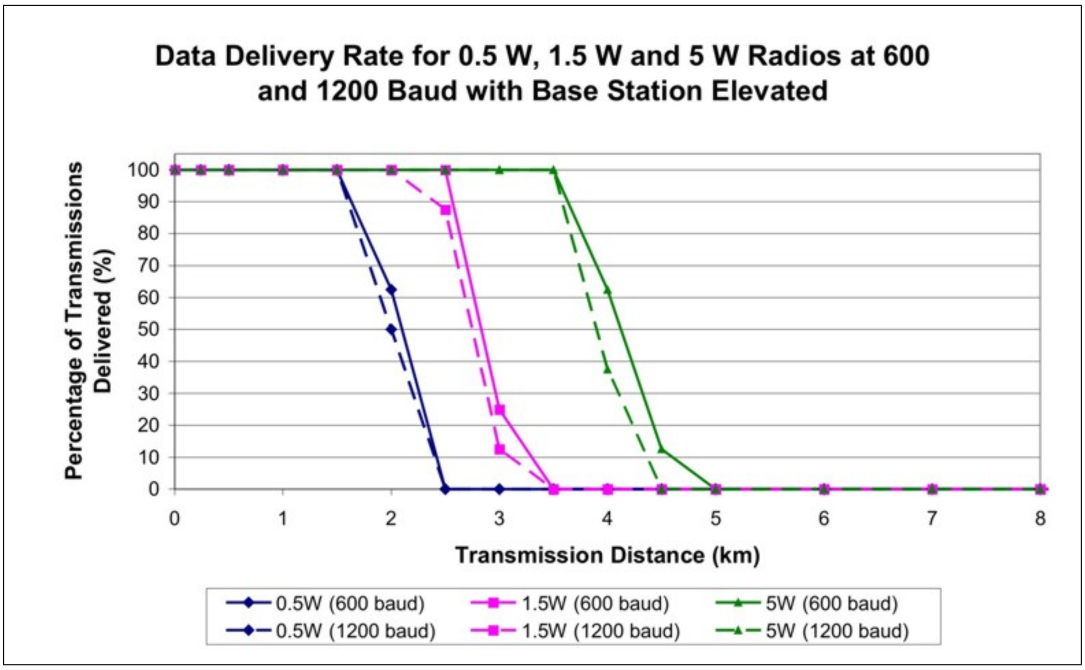


Figure 8.14: Percentage of data successfully delivered at 600 and 1200 baud with base station elevated

8.5.4 Conclusions of Increasing the Data Rate

The technique chosen to increase the data rate to 1200 baud over the UHF citizen band radio was to increase the audio tone frequencies that represent the data. However, the highest frequency that can be used is limited by the bandwidth of the UHF citizen band radio. Hence, bandwidth tests were conducted and found the bandwidth of the citizen band transmission medium to be 1000 Hz to 4200 Hz. Therefore the frequencies tested in bare soil environment evaluation were chosen to be 2600 Hz and 3600 Hz. A summary of the evaluation results for transmitting data at 1200 baud is shown in Table 8.5.

Table 8.5: Summary of telemetry system performance with base station elevated at 1200 baud

Power Output of Radio	Data Rate	Reliable Transmission Distance
0.5 W	1200 baud	1.5 km
1.5 W	1200 baud	2 km
5 W	1200 baud	3.5 km

Extensive tests for transmission at 1200 baud as for the tests at 600 baud were not conducted due to time constraints; however, the tests conducted were through crop canopy and hence in the worst case scenario. The transmission results obtained at 1200 baud were promising, and indicate that transmission at 1200 baud may be used routinely for the agricultural telemetry system.

8.6 Error Analysis

The error rates of the transmissions were analysed for increasing distances between the node and the base station. These errors were transmitted data bits that were received incorrectly. Figure 8.15 shows the error rates of all transmissions between the in-field radio and the base station.

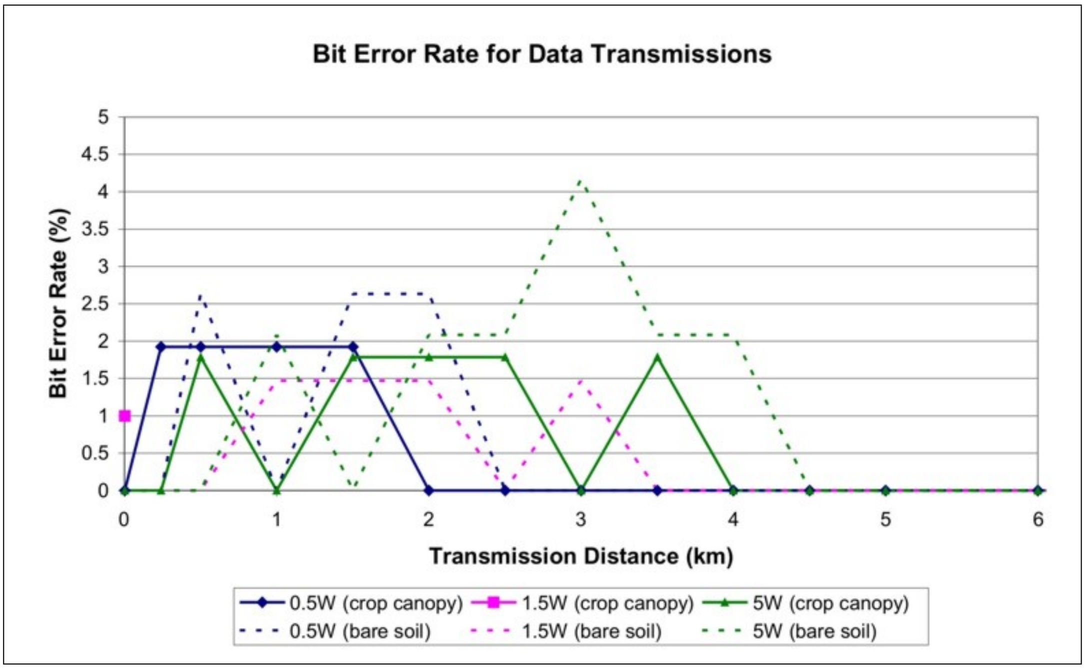


Figure 8.15: Error rate for in-field transmissions

From Figure 8.15 there was no observable correlation between power output of in-field radio, the transmission distance, and the error rate. The error rate was between 0% and 4.2%, but becomes zero when the radios move out of transmission range, and no signal was transmitted successfully. All of the errors in Figure 8.15 were in transmissions from the node to the base station. No errors occurred in transmissions from the base to the in-field nodes due to the two gain antenna of the base station, while the in-field nodes used no gain antennae. The errors were due to the bits of the received byte being shifted, for example the data byte 234 (%11101010) was sent and the byte 245 (%11110101) was received, or bits were received incorrectly, such as the byte 238 (%11101110) being sent and the byte 174 (%10101110) being received.

Errors in transmitted data included incorrect/erroneous data bytes and errors in receiver and transmitter identification numbers that were transmitted from the node to the base. An error in the receiver identification would result in a message being addressed to the wrong node. The network protocol was designed such that the base station will identify this error and ignore the message, then poll the correct node again. Therefore, an error handling scheme was required in the telemetry system protocol.

8.6.1 Error Detection

The ability of the cyclic redundancy code and repetition code to handle errors was evaluated in the field. The errors in the message transmitted were detected by the cyclic redundancy code, and the erroneous byte was corrected by transmitting the identification and data byte five times.

8.6.2 Sources of Errors

Sources of errors include background noise on the radio channel which reduces signal-to-noise ratio of the radio channel, and strings of repetitive bits which potentially cause the demodulator to lose synchronisation.

During the evaluation, the UHF citizen band radios became hot from transmitting frequently and may have overheated, potentially contributing to the error rates. Each radio transmitted data over 100 times in a period of five minutes during the testing to simulate the transmissions of a number of nodes. In an actual telemetry system, each of these radios would only transmit about once every 15 minutes.

For the error tests, the nodes were moved along the field at increments between 500 m and 2 km, so the in-field evaluation was only accurate to between 500 m and 2 km.

8.7 Other Field Results

The system was tested under real life agricultural conditions such as in the sun on a hot day, near power lines and telephone cables, and with interruptions on the radio channel.

8.7.1 Interference from Power Lines and Other Radio Signals

Research by Latimer & Reddell (1990) found that power lines caused noise in low power radio signals, thereby reducing the reliability of the network. The farms visited in Jondaryan and Dalby were often under high tension power lines (see Figure 8.16), so the effects of power lines on the telemetry system must be considered. Farm managers at the barley farm test site in Jondaryan commented on the interference and noise they frequently hear on their mobile telephones from the overhead power lines near their farm.

Figure 8.16: High tension power lines are near many farms in Jondaryan

There were no unexplained lapses in reliability when tests were conducted near overhead power lines or above underground telephone cables. The high power output of the UHF citizen band radios (0.5 W to 5 W) avoided any reduced reliability typically caused by power lines on low power radio transceivers.

8.7.2 Weather Conditions

The field evaluation in the crop canopy was conducted on hot days, while the other tests were conducted on cool days. At higher temperatures, the air is thinner and therefore the propagation of radio waves is reduced. Although tests were not specifically carried out using weather as the variable factor, no significant difference in signal attenuation was observed in results carried out on hot and cool days.

8.7.3 Evaluation of Protocol Robustness

No data collisions occurred due to the robust polling access method of the network, i.e. nodes only transmitted data when requested by the base station. The interruptions on the channel did not cause any data to be lost.

8.7.4 Location of Base Station on Farm

The farm manager's house is often located at the centre of the farm property to minimise the time taken to travel to the boundaries of the property. This effectively implies that a telemetry system that is able to propagate radio signals 3 km, actually has a coverage area of 6 km if the base station is located at the centre of the farm.

8.8 Network Timing Analysis

A network timing analysis is useful to ensure that the transmission time by the base station does not exceed three seconds in every hour, in accordance with the Citizen Band Radio Stations Class Licence. A maximum of five attempts to poll each node are made and at least two transmissions in total are required to obtain data from one node. Therefore, a maximum of ten transmissions can be made by each node. If each message is 12 bytes, for a forty sensor node network the base station will send:

12 bytes of data $\times 8$ bits per byte $\times$ 40 nodes $\times 4$ polls per hour $= 15360$ bits

This transmission will take 12.8 seconds at 1200 baud if the nodes are polled every 15 minutes and data does not have to be retransmitted, which is above the three second duty cycle per hour allowed by the citizen band class licence.

Methods of reducing this telemetry system's transmission time include (a) using a network topology where in-field radios poll some nearby nodes so the base station interrogates fewer radios for data; (b) increasing data rate; (c) reducing polling interval; and (d) using fewer repetition codes.

8.9 Field Evaluation Conclusions

A series of tests enabled the performance of the UHF citizen band radio system to be evaluated. No intermittent reliability of radio transmissions was experienced during the field tests at 600 and 1200 baud. The position of the base station antenna influenced the UHF citizen band radio chosen for use at the in-field nodes. When the base station antenna was 0.5 m from the ground, the 1.5 W radio reliably transmitted data (with 100% data delivery rate) at 600 baud for 2.5 km in both a bare soil and crop canopy environment. When the base station antenna was mounted 4 m from the ground, the 0.5 W radio reliably transmitted data at 600 baud for 2 km. No messages were delivered successfully at 1200 baud using 1200 Hz and 2200 Hz to represent the logic 0 and 1, respectively, due to Shannon's Sampling Theorem.

The bandwidth of the UHF citizen band radio was found to be between 1000 Hz and 4200 Hz, and higher tone frequencies (2600 Hz and 3600 Hz) were tested to increase the data rate to 1200 baud. The transmission range at 1200 baud was 2 km for the 0.5 W radio, and hence remained the UHF citizen band radio of choice for the in-field nodes.

Overhead power lines, underground telephone cables and hot conditions had no observed effect on the transmission reliability. The protocol was implemented successfully in the field.

Chapter 9

Conclusions

A telemetry system of UHF citizen band radios for farm use has been developed. Benefits of the telemetry system developed include potentially improving the level and quantity of in-field crop data being collected, thereby improving the profitability of farms by increasing water application efficiency, reducing labour and maximising yields.

9.1 Achievement of Objectives

The aim of this project was to develop a wireless communication link between in-field surface irrigation advance rate sensors to a base station, up to 2 km apart, to enable close to real-time monitoring of in-field irrigation application and other on-farm operations.

Agricultural sensors, such as those that measure surface irrigation advance rate, and their methods of data transmission were investigated. Existing advance rate sensors utilise an infra-red link to transfer data from the sensor to a handheld computer, or had no wireless transmission capabilities.

Existing on-farm sensor networks, both commercial and academic research prototypes, use short-range and high data rate wireless technologies such as Zigbee and Bluetooth, or long range and low data rate transceivers such as UHF citizen band radios. UHF citizen band radio was envisaged to be a reliable and long distance method of wireless transmission, while some short-range transceivers experience poor performance in available research results of in-field tests.

A wireless network was designed with: a polling medium access method; star topology; error handling using repetition and cyclic redundancy codes; and robust protocol. Both software- and hardware-based decoding were investigated, but demodulating hardware using modem integrated circuits such as XR2211 was the most viable approach. PICAXE microcontrollers were used due to the simplicity of their programming. The base station hardware required a computer to display the information received by the central node. The in-field hardware was enclosed in a rugged container.

The telemetry system was tested in the field, in both bare soil and crop canopy environments and with the base station positioned at ground level and mounted several metres above the ground. A 0.5 W radio transmitted from the in-field nodes to the base for 2 km with 100% reliability at 1200 baud, satisfying all the telemetry system requirements. No interference or reduced reliability of the wireless communications from nearby irrigation water or overhead power lines were observed during the in-field evaluation.

A user interface was developed to allow the farm manager to monitor the status of the in-field sensors, as well as to be alerted if an in-field node was offline.

The UHF citizen band radio may potentially be interfaced with other wireless networks using a different wireless technology such as Zigbee by using an interfacing circuit.

Wireless image transmission to monitor on-farm operations was considered with potential areas of further research to increase the data rate over UHF citizen band radio including data modulation, compression and multiplexing.

9.2 Comparison of Specifications with Other Wireless Technologies

The telemetry system developed had the following characteristics:

- Greater transmission distance than some academic research telemetry systems (which were as low as 18 m).
- No unexplained intermittent reliability as reported by literature. Data was reliably delivered from the sensor nodes at 1200 baud for 2 km through crop canopy, while some wireless networks in research papers reported reliability of 63.8% and were prone to frequent failures (Shih, Cho, Ickes, Min, Sinha, Wang & Chandrakasan 2001).
- Cheaper cost per node than GME telemetry system ($1000 per node), with the total price to make the node being $118 (price for single item quantities), and comparable to the XBee Zigbee Pro Wireless Module which costs $87 each.
- Flexible sensor connection options.
- No fees since the transmission medium uses a licence-free radio frequency.
- Restrictions on citizen band include three second transmission duty cycle.
- Lower data rates (1200 baud) than short-range transceivers such as Zigbee which have a data rate of up to 250 kbps.

9.3 Other Applications of Developed Telemetry System

The telemetry system developed has credentials both inside and outside the agricultural industry. The telemetry system may be the medium used to remotely control on-farm operations such as irrigation control, closing fences or turning off a pump. Potential areas for applying the telemetry system developed that lie outside the agricultural industry include environmental monitoring, where nodes are deployed to measure in the open, and applications in the manufacturing and construction industries.

9.4 Further Work

Further work that could be completed include the investigation of increasing the data rate over UHF citizen band radio, which would enable the timely transmission of data from data heavy sensors, such as images from cameras.

Data modulation, compression and multiplexing are methods identified to be investigated to increase the data rate of transmissions over UHF citizen band radios. The modem devised for the UHF citizen band radio system may be a new design or may incorporate an existing technology. For example, a computer modem implementing pulse coded modulation (such as the V.90 standard) has a data transmission rate of 56 kbps, and may potentially be a worthwile model to modulate and demodulate data for UHF citizen band radio.

Multiplexing is another method to increase the data rate of signals transmitted over UHF citizen band radios. Examples of multiplexing methods are sub-audible and sub-carrier signalling. Compressing the data to be transmitted would also potentially increase the data rate. An example of compression is the identification and encoding of repeating characters into smaller characters. *Audio communication band image transceiver* (1996) details a design that is able to transmit images faster than standard SSTV methods over an audio communication channel by a digital signal processing method, rather than by separating the image signal into red/green/blue components. This method may offer direction for image compression for citizen band radio.

With further research the UHF citizen band radio may potentially allow higher data rates and rival other system's capacities.

References

1200 Baud Data Modems (2006), Hamtronics, Inc.
http://www.hamtronics.com/data_1200.htm.

Adcon Remote Telemetry Units (2006), Adcon.
http://www.adcon.at/english/produkte_rtus_en.html.

Agrilink Communications (2006), Agrilink Holdings Pty Ltd.
http://www.agrilink-int.com/.

Akyilidiz, I. F., Su, W., Sankarasubramaniam, Y. & Cayirci, E. (2002), 'Wireless sensor networks: a survey', *Computer Networks* **38**, 393–422.

Ant (1995), *Installation of CD50-68-50 and CD51-68-50 Series Mopole^{TM} Antennae*. Data sheet from RFIndustries Pty Ltd.

Audio communication band image transceiver (1996), United States Patent and Trademark Office.
http://patft.uspto.gov/.

AudioLab 2.2 (2006), Mitov Software.
http://www.mitov.com/html/audiolab.html.

Australian radiofrequency spectrum allocations chart (2006), Australian Communications Authority, Australian Government.
http://www.acma.gov.au/.

Beyda, W. J. (2000), *Data Communications*, Prentice-Hall Inc, New Jersey.

Cardell-Oliver, R., Smettem, K., Kranz, M. & Mayer, K. (2004), 'Field testing a wireless sensor network for reactive environmental monitoring', *Proceedings of the 2004 Intelligent Sensors, Sensor Networks and Information Processing Conference* pp. 7–12.

Carter, T. W. & Whitehead, P. (2004), *Wireless Networking*, Wiley Publishing, Inc., Indiana.

CB Monitor Sensor and Telemetry System (2006), Electrosense Technologies.
`http://www.electrosense.com.au/`.

Citizen band radio (2006), Australian Communications Authority, Australian Government.
`http://www.acma.gov.au/ACMAINTER.65640:STANDARD:799048457:pc=PC_1688`.

Dalton, P. A. (2000), 'Watertight, whole farm water use efficiency, determining your own water security', *In Proceedings of the 2000 Australian Cotton Conference* .

Fitzgerald, J. & Dennis, A. (2005), *Business Data Communications and Networking*, John Wiley and Sons Inc, USA.

Gibilisco, S. (1999), *Handbook of Radio and Wireless Technology*, McGraw-Hill Companies Inc, New York.

GME Telemetry System (2005), GME.
`http://www.gme.net.au/commercial/telemetry.php`.

Gokhale, A. A. (2001), *Introduction to Telecommunications*, Delmar Thomson Learning, USA.

Gralla, P. (2002), *How Wireless Works*, Que Publishing, Indiana.

Greenfield, T. (2002), *Research Methods for Postgraduates*, Arnold Publishers, London.

Halsall, F. (2005), *Computer Networking and the Internet*, Pearson Education Limited, England.

Hirsch, S. (2006), *Spectrum Scope*, Spectrum Scope.
`http://www.mathworks.com`.

Karl, H. & Willig, A. (2005), *Protocols and Architectures for Wireless Sensor Networks*, John Wiley and Sons Inc, USA.

King, K. D. (2002), 'Wireless technologies: An overview', *OJNI* **6**(1).

Latimer, E. A. & Reddell, D. L. (1990), 'Components for an advance rate feedback system', *Transactions of the ASAE* **81**, 1162–1170.

Leis, J. (2002), *Digital Signal Processing - A Matlab-Based Tutorial Approach*, Research Studies Press Ltd, England.

Leis, J. (2006), *ELE3305 Computer Systems and Communications Protocols Study Book*, University of Southern Queensland, Toowoomba.

Leon-Garcia, A. & Widjaja, I. (2000), *Communication Networks*, McGraw-Hill Companies Inc, Canberra.

Long, B. (2006), *Speech Synthesis and Speech Recognition Using SAPI 5.1*, Brian Long Consultancy and Training Services.
`http://www.blong.com/Conferences/DCon2002/Speech/SAPI51/SAPI51.htm`.

Mackay, D. J. C. (2003), *Information Theory, Inference and Learning Algorithms*, Cambridge University Press, Cambridge, UK.

Mainwaring, A., Polastre, J., Szewczyk, R., Culler, D. & Anderson, J. (2002), 'Wireless sensor networks for habitat monitoring', *Proceedings of First ACM International Workshop on Wireless Sensor Networks and Applications* . Atlanta, Georgia, USA.

Morais, R., Valenta, A. & Serhatodio, C. (2005), 'A wireless sensor network for smart irrigation and environmental monitoring: A position article', *2005 EFIT/WCCA Joint Congress on IT in Agriculture* . Brisbane.

PIC (2006), *PICAXE Manual 1: Getting Started.* Instruction manual supplied with PICAXE programming editor.

PICAXE Connect (AXE210) (2006), Revolution Education Ltd.
`http://www.rev-ed.co.uk/picaxe/`.

Purcell, J. (2004), 'Doing better with what we have: The art of surface irrigation', *Australian Cottongrower* **24**(7), 36.

Rad (2006), *GME Electrophone TX4400 40 Channel UHF Transceiver.* Instruction manual.

Radiocommunications (Citizen Band Radio Stations) Class Licence (2002), Australian Communications Authority, Australian Government.
`http://www.acma.gov.au/ACMAINTER.65640:STANDARD:799048457:pc=PC_287`.

Raine, S. R. & Dalton, P. (2003), 'Final report: An investigation of in-field irrigation management practices to improve the efficiency of furrow irrigated cotton production systems', *National Centre of Engineering in Agriculture Publication* . Paper No. #1000006/2, NCEA, Toowoomba.

Raine, S. R., Purcell, J. & Schmidt, E. (2005), 'Improving whole farm and in-field irrigation efficiencies using irrimate tools', *Irrigation 2005: Restoring the Balance* .

Robinson, C. (2006), *Irrigation*, West Texas A & M University, USA.
`http://www.wtamu.edu/~crobinson/Irrigation/cotrow.jpg`.

Shih, E., Cho, S., Ickes, N., Min, R., Sinha, A., Wang, A. & Chandrakasan, A. (2001), 'Physical layer driven protocol and algorithm design for energy-efficient wireless sensor networks', *Proceedings of ACM MobiCom'01* pp. 272–286.

Szewczyk, R., Mainwaring, J. P. A. & Culler, D. (2004), 'Lessons from a sensor network expedition', *Proceedings of the First European Workshop on Wireless Sensor Networks* . Berlin, Germany.

TCM3105 1200 Baud Modem IC (2006), Technicothica.
`http://www.klm-tech.com/`.

Technical Information for Choosing Solar Module (2006), PowerFilm Inc.
`http://www.powerfilmsolar.com/products/oem_components/technical.htm`.

Ter (2006), *Terracept NICTOR^{TM} Telemetry Units.* Brochure.

Tucson Amateur Packet Radio (1996), Tapr-TNC.
`http://ftp.iasi.roedu.net`.

Turnell, D. J., Deep, G. S. & Freire, R. C. S. (1997), 'A low-cost water detection method for furrow irrigation control', *IEEE Transactions on Instrumentation and Measurement* **46**(4), 986–990.

Wall, R. W. & King, B. A. (2005), 'Networking protocol for multi-media ad hoc data communications', *Applied Engineering in Agriculture* **21**(2), 285–293.

Wang, N., Zhang, N. & Wang, M. (2006), 'Wireless sensors in agriculture and food industry - recent development and future perspective', *Computers and Electronics in Agriculture* **50**(1), 1–14.

Water Use on Australian Farms (2003), Australian Bureau of Statistics, Canberra.

WeatherTRAK ET Irrigation Controller (2006), Aquacraft.
`http://www.aquacraft.com/Products/weathertrakAbout.htm`.

Wireless LANs - what and how (2006), Australian Communications Authority, Australian Government.
`http://www.acma.gov.au/ACMAINTER.65640:STANDARD:799048457:pc=PC_1739`.

Ye, W., Heidemann, J. & Estrin, D. (2004), 'Medium access control with coordinated adaptive sleeping for wireless sensor networks', *IEEE/ACM Transactions on Networking* **12**, 493–506.

Zhang, Z. (2004), 'Investigation of wireless sensor networks for precision agriculture', *2004 ASAE/CSAE Annual International Meeting* . Paper No. #041154, ASAE, St. Joseph, Mich.

Zig (2006), *XBee and XBee-PRO 2.4 GHz OEM RF Modules.* Data sheet.

Appendix A

Project Specification

University of Southern Queensland

FACULTY OF ENGINEERING AND SURVEYING

ENG 4111/2 Research Project
PROJECT SPECIFICATION

FOR:	**Alison McCARTHY**
TOPIC:	TELEMETRY AND COMMUNICATION SYSTEMS FOR IRRIGATION MANAGEMENT
SUPERVISORS:	Mark Phythian Erik Schmidt
PROJECT AIM:	This project aims to provide a wireless communication link from infield surface irrigation advance rate sensors and other sensors to a base station, to enable close to real-time monitoring of in-field irrigation application and other on-farm operations.
PROGRAMME:	**Issue B, 25 May 2006**

1. Research sensors that measure surface irrigation advance rate and their methods of data transmission.
2. Critically evaluate existing on-farm sensor networks and wireless link technologies that may be employed on-farm, comparing performance characteristics such as data rate, transmission range and networking configurations.
3. Design a rugged system, with a usage life of 3 years, that transmits wireless data up to 2000 metres from infield advance sensors to a base station, thus enabling close to real-time evaluation of farm operations.
4. Evaluate the performance of the designed system under field conditions such as the presence of crops and irrigation water, and consider the effect of interference from other wireless devices.
5. Design an interface that enables farm managers to monitor the status of sensors.
6. Investigate techniques to enable interfacing of the designed wireless network with other existing on-farm networks using a different wireless technology.

As time permits:

7. Investigate and implement techniques for wireless image transmission to monitor on-farm operations.

AGREED:

_______ (Student) _______ , _______ (Supervisors)

25 / 5 / 06 25 / 5 / 06 . 25 / 5 / 06

Appendix B

Voice Modulation Hardware Development

Hardware was built to test the transmission of voice over the UHF citizen band radio to indicate when an advance rate sensor detected water. The node undergoes the following process:

- Turn on the radio when water was detected by the sensor
- Play and transmit a prerecorded sound over the radio that was unique for each node from a sound module or MP3 player
- Turn off the radio

The timing of the circuit to turn on the radio was controlled by NE555 timers through relays and was initiated by a short circuit on the water detector input. When the base station UHF citizen band radio received a particular tune or sound, it could identify the node at which water was detected.

B.1 Sound Modules

An alternate method of transmitting voice over UHF citizen band radio was to record a unique sound or tune for each node to devices such as a recordable sound module from a greeting card, a recordable module from a voice memo (see Figure B.1) and an MP3 player (see Figure B.2). The recordable sound module stored 10 seconds of sound, the recordable voice memo module stored 20 seconds of sound and the MP3 stored hours of sound.

A spoken number or tune was recorded on each of these modules and transmitted over the radio. The recordable module taken from the voice memo had the poorest quality signal and a large amount of noise was present in the signal. The greeting card module was then tested and had improved quality over the voice memo module; but, the noise present was still significant. Therefore, an MP3 player was used and had superior quality over the recordable modules.

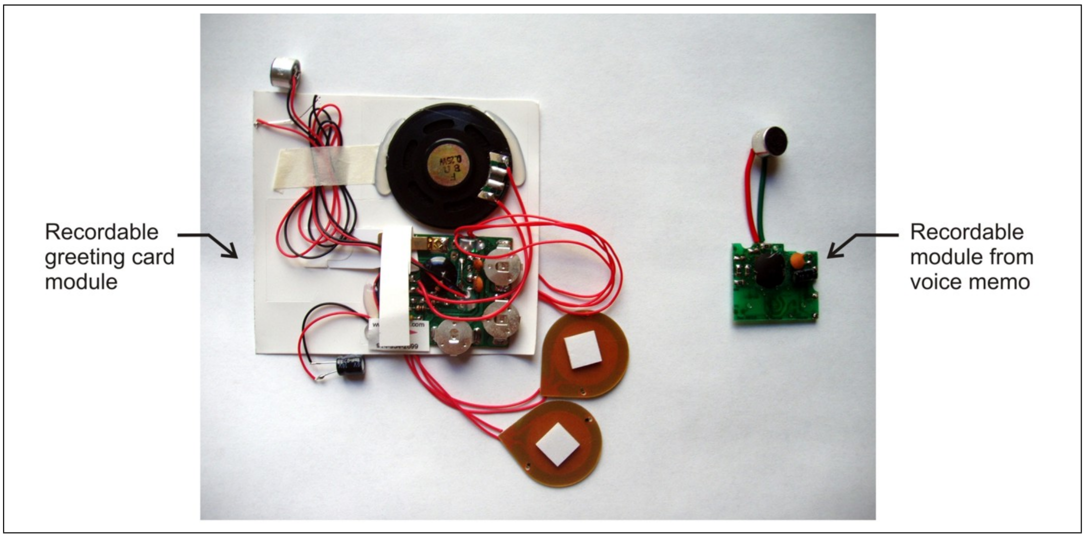


Figure B.1: Recordable sound modules tested

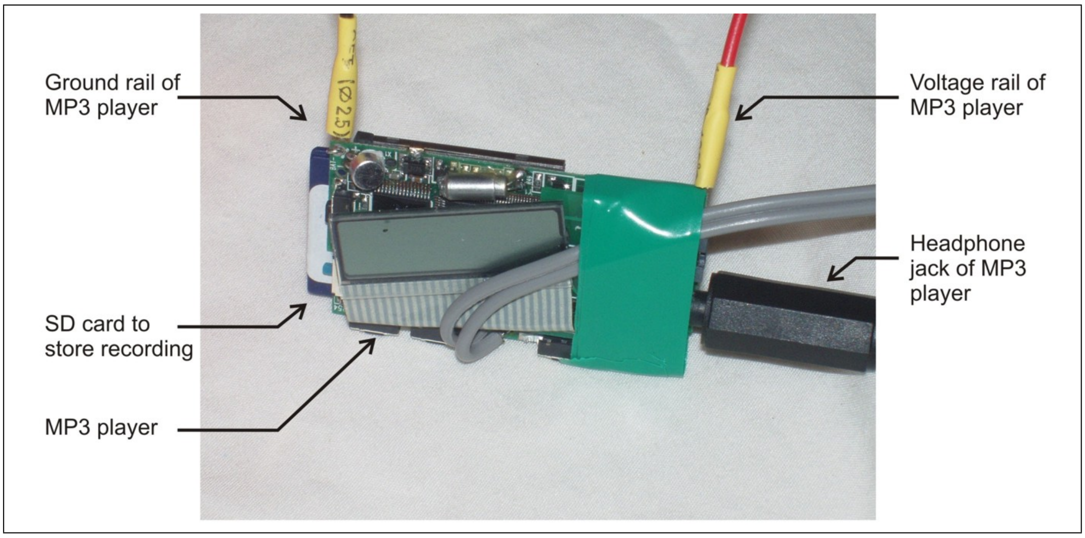


Figure B.2: MP3 player tested

B.2 Circuit Built

Hardware that played and transmitted a track from the MP3 player over a UHF citizen band radio was developed, as per Figures B.3 and B.4.

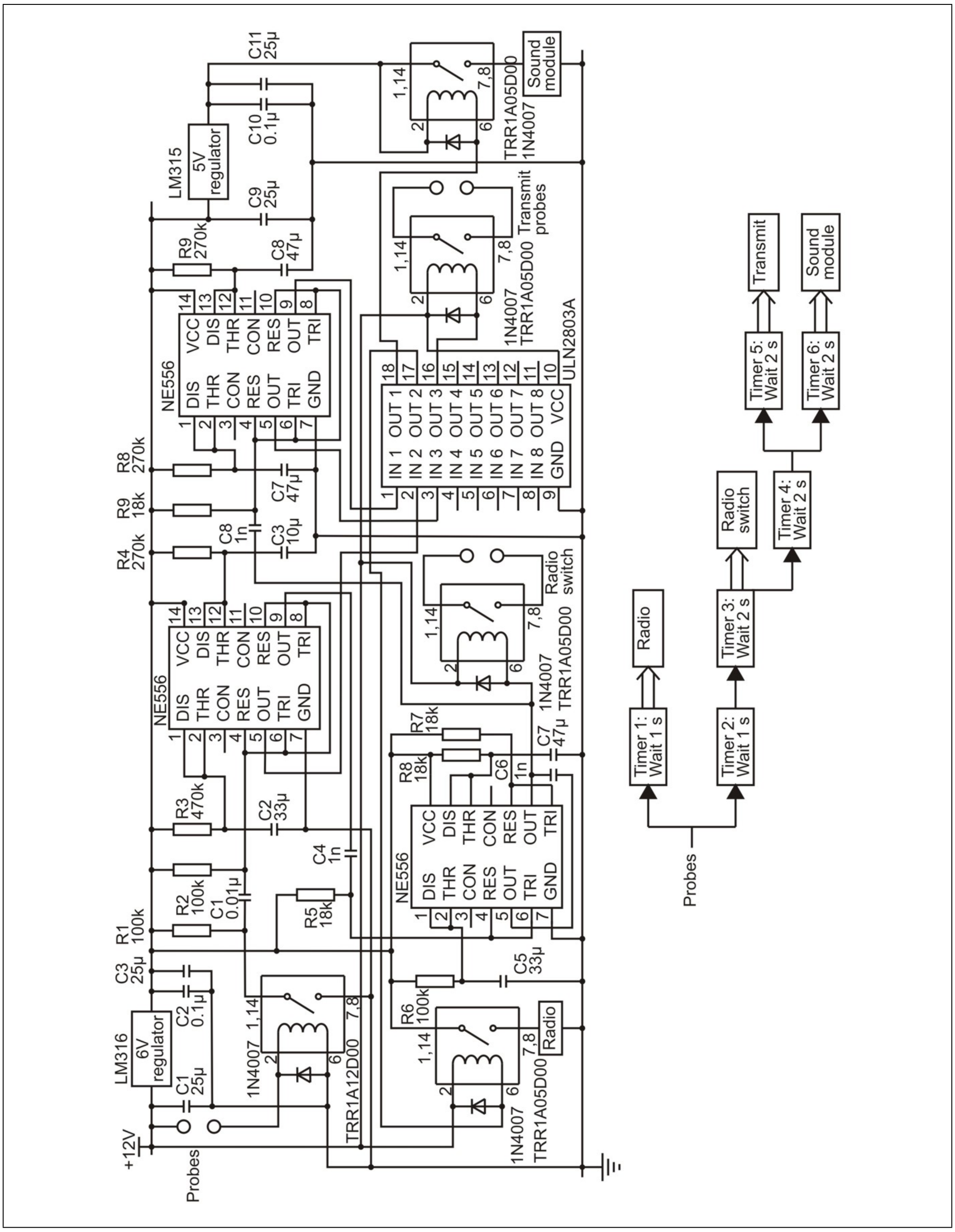


Figure B.3: Interfacing circuit that transmits recording from MP3 player over radio

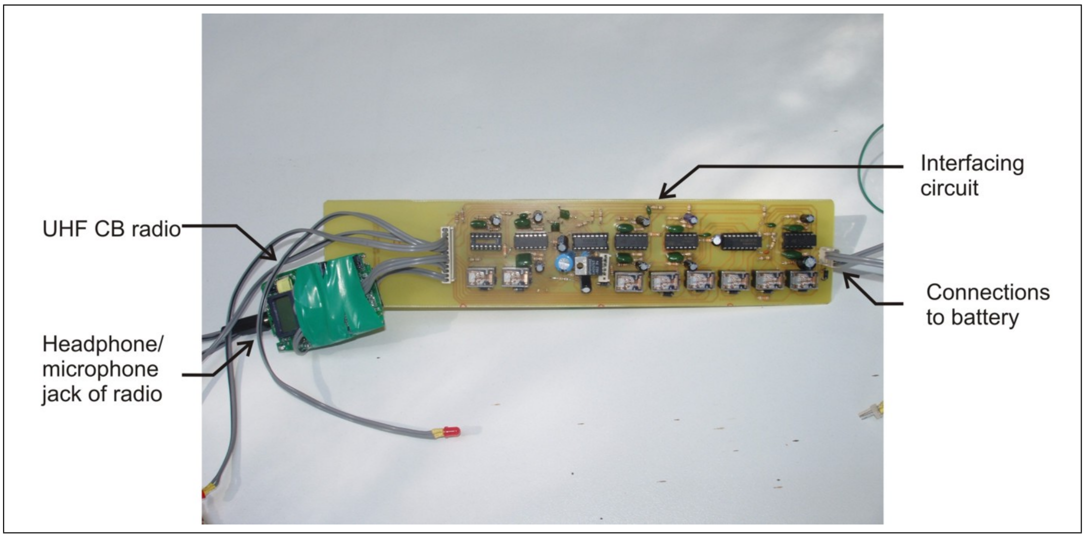


Figure B.4: Printed circuit board that transmits recording from MP3 player over radio

Appendix C

Interfacing Circuit Development

C.1 Interfacing Circuit Using TCM3105 Modem

The TCM3105 modem, initially used as the interfacing circuit modem, was connected as per Figure C.1, based on the recommended circuit in the data sheet (see Appendix G). The circuit connections on the TRS, TXR1 and TXR2 pins were set according to the data sheet so that the modem would transmit and receive at 1200 baud using audio tones frequencies 1200 Hz and 2200 Hz to represent logic 0 and 1, respectively.

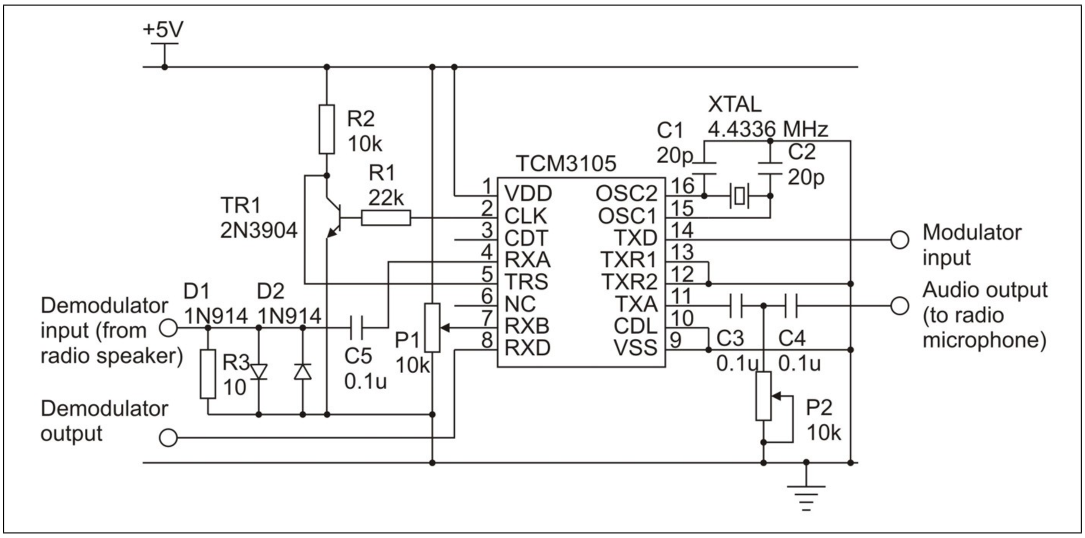


Figure C.1: Schematic of interfacing circuit using TCM3105 modem

C.2 Interfacing Circuit Using FX614 Modem

The schematic diagram of the circuit used to test the FX614 modem is illustrated in Figure C.2, based on a circuit developed for amateur radio use as per *TCM3105 1200 Baud Modem IC* (2006).

C.3 Interfacing Circuit Using XR2206 and XR2211

An interfacing circuit using the XR2206 modulator and XR2211 demodulator was designed, but first the I/O requirements of the circuit microcontrollers were determined. Table C.1 outlines the tasks completed by the microcontrollers in the interfacing circuit.

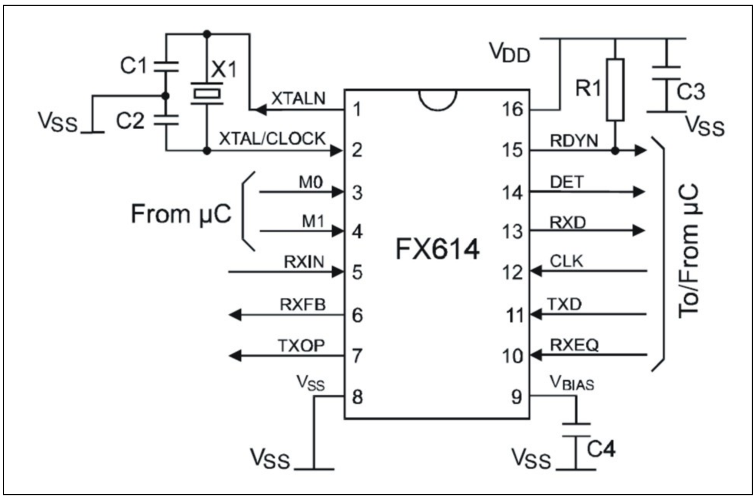


Figure C.2: Schematic of interfacing circuit using TCM3105 modem

C.3.1 PICAXE I/O Requirements for Interfacing Circuit

From Table C.1 the input and output requirements of each PICAXE in the interfacing circuit were determined. Table C.2 summarises the requirements and displays the minimum PICAXE that could be used.

The PICAXE-08M was the smallest of the PICAXE variants to satisfy the I/O requirements for the buffer PICAXE of both the node and the base station, with serial communications capabilities (PIC 2006). Therefore a PICAXE-08M was chosen as the buffer PICAXE.

The PICAXE-18X satisfies the I/O requirements of the master PICAXE at the node and base, as it provides up to eight digital outputs, three analogue inputs, and with serial communications, I2C and interrupt capabilities. However, a PICAXE-40X was used as the master PICAXE due to the developmental nature of the work.

C.3.2 Schematic Diagram of Interfacing Circuit

Using a PICAXE-08M for the buffer PICAXE and a PICAXE-40X for the master PICAXE, the interfacing circuit was designed. The connections for the XR2206 and XR2211 were based on a commercial amateur radio modem kit using these integrated circuits from *1200 Baud Data Modems* (2006).

Table C.1: Requirements of microcontrollers in interfacing circuit

Base/Node and Master/Buffer	**Tasks**	**I/O Required**	**Notes**
Base and node master	Output serial data to modulator	1 serial output	Output baud rate is baud rate of transmitted audio signal
Base and node master	Read time and date from real-time clock	1 SDA input, 1 SCL output	SDA and SCL connections are required for communications with devices such as real-time clocks and EEPROMs
Base and node master	Control power and transmission switch of radio	3 digital outputs	Connects the voltage rail to the radio, and presses 'on' and 'transmit' buttons
Base and node master	Read data from online sensor	1 serial input, 1 analogue input	Analogue input for advance probe sensors data, serial input for analogue sensor values
Base and node master	Receive data from buffer PICAXE	1 serial input	Interrupt is required to indicate that buffer PICAXE has data ready to send
Base master	Transfer sensor to computer	1 serial input, 1 serial output	Serial connections to base station computer to display sensor data on graphical user interface
Base and node buffer	Receive serial data from demodulator	1 serial input	Input baud rate is baud rate of transmitted audio signal
Base and node buffer	Transfer data from buffer PICAXE to master PICAXE	1 serial output	Output data serially to master PICAXE on same line as interrupt is set
Base and node, master and buffer	Implement network protocol	Nil	PICAXE has limited code listing length

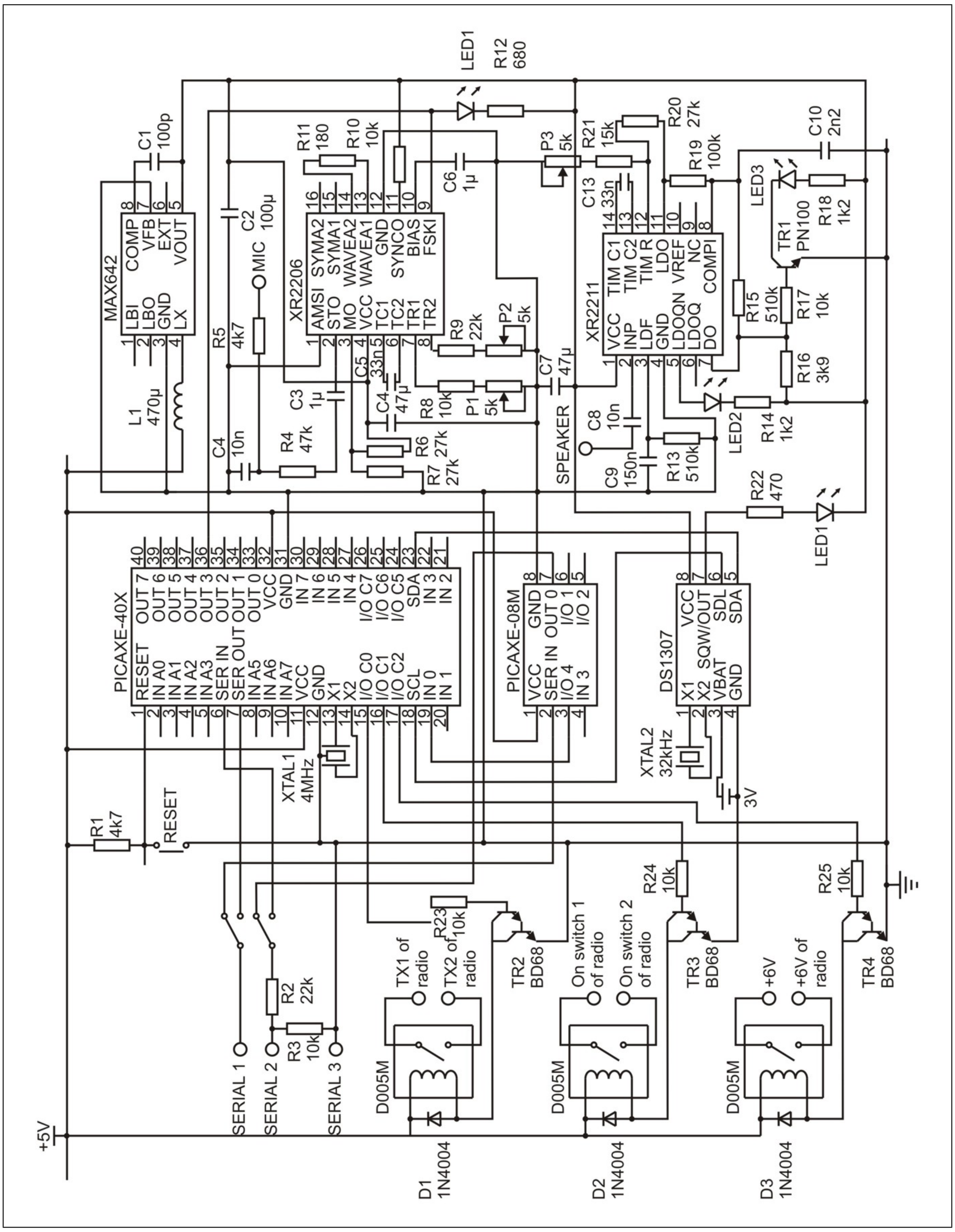


Figure C.3: Schematic of circuit to transmit and receive data using audio tones

Table C.2: PICAXE I/O requirements for interfacing circuit

	Serial IP	**Serial OP**	**Digital IP**	**Digital OP**	**Anal. OP**	**Other**
Base master	2	2	0	3	1	SDA/SCL, interrupt
Base buffer	1	1	0	0	0	Nil
Node master	1	1	0	3	1	SDA/SCL, interrupt
Node buffer	1	1	0	0	0	Nil

The final printed circuit board designed and made for each node is illustrated in Figure C.4.

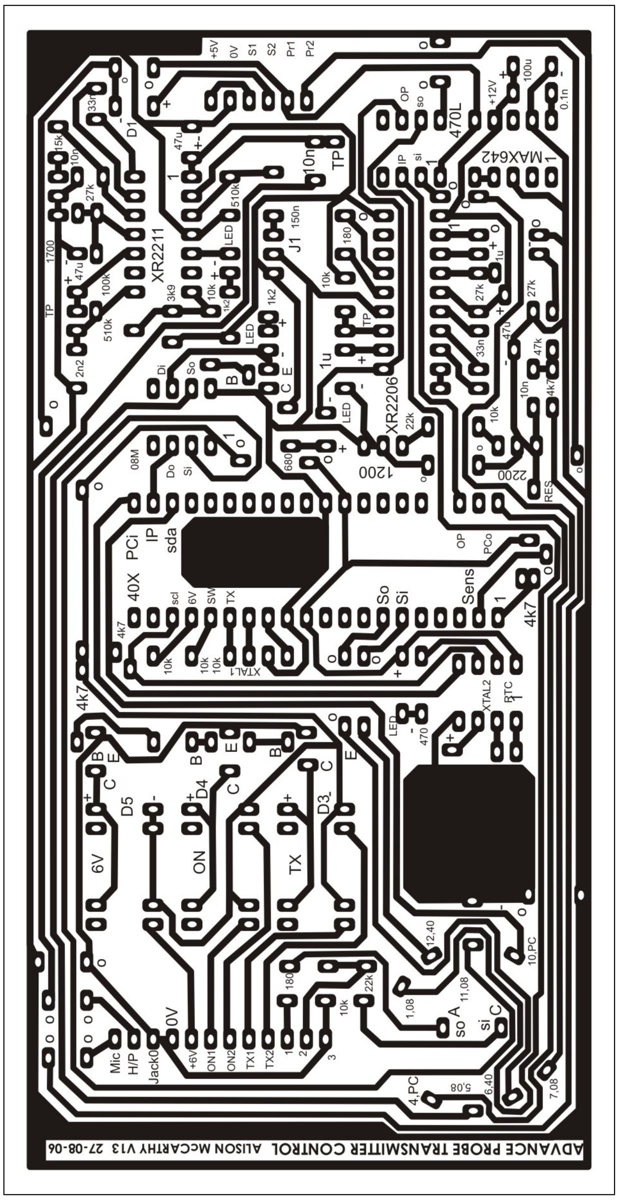


Figure C.4: Printed circuit board of final interfacing circuit

C.3.3 Connections to UHF Citizen Band Radio

The connections made to each UHF citizen band radio to control when the radio was on, off or transmitting are shown in Figures C.5 and C.6. The power supply, 'on' and 'transmit' buttons were controlled via relays by the microcontroller in the interfacing circuit.

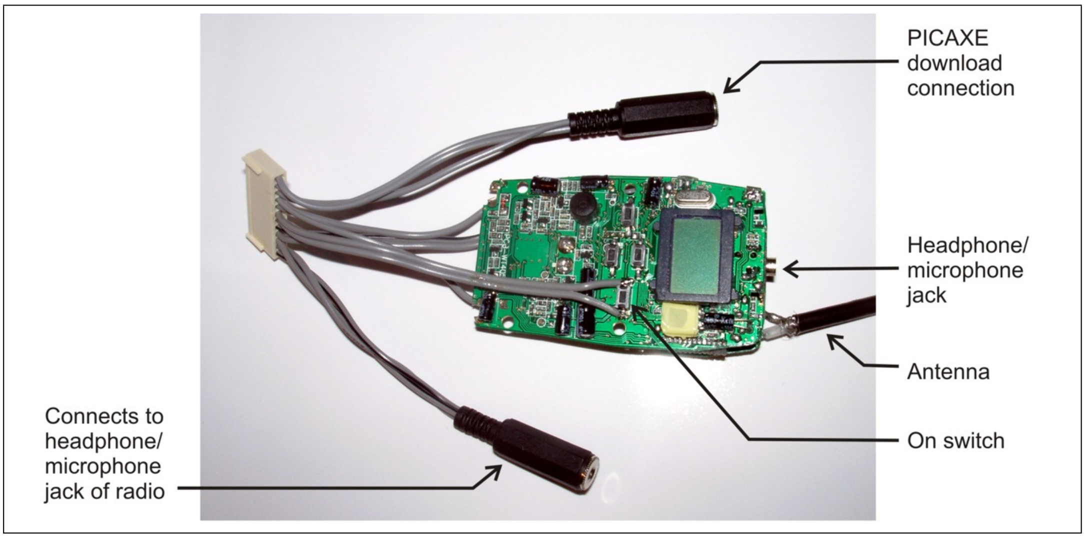


Figure C.5: Connections to front of Dick Smith Electronics radio

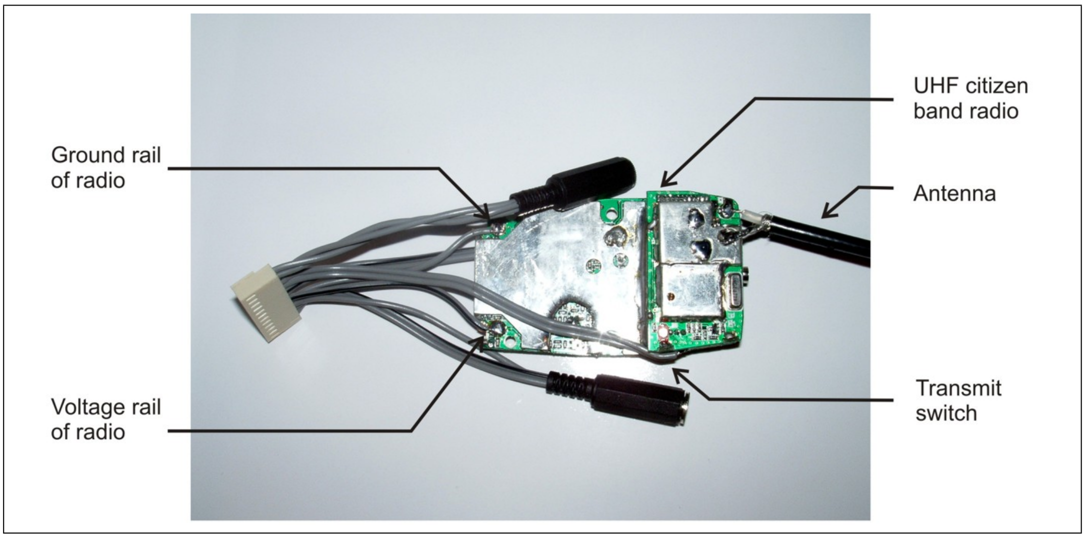


Figure C.6: Connections to back of Dick Smith Electronics radio

C.3.4 Price List for Interfacing Circuit

See Table C.3 for the interfacing circuit price list.

Table C.3: Price list of node constructed for field evaluation

Item	Description	Price	Purchased
Printed circuit board	75 mm × 150 mm	$ 3.78	Dick Smith Electronics
PICAXE-40X	40 pin microcontroller	$14.60	Tech-Supplies UK
PICAXE-08M	8 pin microcontroller	$ 3.74	Tech-Supplies UK
XR2206	FSK modulator IC	$ 6.58	Futurlec Australia
XR2211	FSK demodulator IC	$ 2.33	Futurlec Australia
MAX642	Step up regulator IC	$ 4.32	Futurlec Australia
DS1307	Real time clock IC	$ 3.92	Modtronix Australia
32 kHz crystal	Crystal for DS1307	$ 1.00	Futurlec Australia
3V Lithium Battery	Backup battery for DS1307	$ 1.10	Leading Edge Electronics
Battery holders	1 × 3V lithium	$ 1.55	Jaycar Electroincs
38 × Resistors	11 ×0Ω,	$ 0.013	Futurlec Australia
	2 ×180Ω	$ 0.013	Futurlec Australia
	7 ×10 kΩ	$ 0.013	Futurlec Australia
	2 ×510 kΩ	$ 0.013	Futurlec Australia
	2 ×22 kΩ	$ 0.013	Futurlec Australia
	1 ×470Ω	$ 0.013	Futurlec Australia
	4 ×4.7 kΩ	$ 0.013	Futurlec Australia
	1 ×47 kΩ	$ 0.013	Futurlec Australia
	3 ×27 kΩ	$ 0.013	Futurlec Australia
	1 ×680Ω	$ 0.013	Futurlec Australia
	2 ×1k2Ω	$ 0.013	Futurlec Australia
	1 ×100 kΩ	$ 0.013	Futurlec Australia
	1 ×15 kΩ	$ 0.013	Futurlec Australia
9 × IC socket	1 × 40 pin	$ 0.08	Futurlec Australia
	4 × 14 pin	$ 0.08	Futurlec Australia
	1 × 16 pin	$ 0.08	Futurlec Australia
	3 × 8 pin	$ 0.08	Futurlec Australia
UHF citizen band radio	0.5 W radio	$25.00	Domiclo Australia
Antenna	UHF 2.16 dB gain	$29.95	Michael's Electronics
Container	Fibreglass	$ 2.00	BMS Mitre 10
Batteries	4 × NiMH	$13.00	Futurlec Australia
Solar panel	6 V flexible solar panel	$18.00	Sun Source Power Solar Power Products
	Total	$117.81	

Appendix D

PICAXE Code Listing

D.1 PICAXE Code Listing for Node Buffer

```
' Node Buffer
' Written by:  Alison McCarthy

' Purpose:
'  To receive demodulated data from XR2211 from radio channel

' Inputs:
'  RXPIN - Receives demodulated data to the central node master


' Outputs:
'  DataO - Outputs data demodulated by XR2211

' Constants:
'  PREAMBLEBYTE - Preamble byte
' SOMBYTE - Start of message byte
' EOMBYTE - End of message byte
' POSTAMBLEBYTE - Postamble byte
' RXBAUD - Baud rate of data received over UHF citizen band radio
' BAUD - Baud rate of transmission between master and slave
' SENDERCODE - Identifier of node
```

```
' Variables:
'  RECEIVERCODE - Identifier of node to receive data
' RECEIVERCODE2 - Identifier of node to receive data
' SENDERCODE2 - Identifier of node that is sending the data
' TXBAUD - Baud rate of transmission

'-----------------------------------------------------------------------
' Assign symbol names to PICAXE-08M inputs and outputs

symbol DataO = 4 ' Output 4, pin 3, receives data to be modulated from the master
symbol RXPIN = 1 ' Input 1, pin 6, output data to the modulator XR2206 to transmit
over the radio

'-----------------------------------------------------------------------
' Assign symbol names to constants

symbol PREAMBLEBYTE = $AA
symbol SOMBYTE = $C5
symbol EOMBYTE = $5C symbol SENDERCODE = 10

symbol RXBAUD = N600 ' Receive data at 600 baud
symbol BAUD = T2400 ' Transfer data to master at 2400 baud

'-----------------------------------------------------------------------
' Assign symbol names to variables

symbol RECEIVERCODE = b0 ' 0 ..255 stored in b1
symbol RECEIVERCODE2 = b1 ' 0 ..255 stored in b1
symbol SENDERCODE2 = b2 ' 0 ..255 stored in b2
symbol ACK = b3
symbol POLL = b4
symbol byte = b5

'-----------------------------------------------------------------------

Loop:  '  Read in data received by demodulator
serin RXPIN,RXBAUD,(PREAMBLEBYTE,SOMBYTE),RECEIVERCODE,SENDERCODE2,
```

```
POLL,RECEIVERCODE2,ACK

if RECEIVERCODE <> SENDERCODE2 then Loop ' If the poll is not addressed to this
node then go back to start


high Data0 ' Set flag on pin of interrupt
pause 20 ' Wait for master PICAXE to respond

' Transfer data to master PICAXE
serout Data0,BAUD,(PREAMBLEBYTE,PREAMBLEBYTE,PREAMBLEBYTE,PREAMBLEBYTE,
PREAMBLEBYTE,SOMBYTE,RECEIVERCODE,SENDERCODE,POLL,RECEIVERCODE2,ACK)

low Data0 ' Turn off flag of interrupt pin
goto Loop ' Go back to start to receive more data

'-----------------------------------------------------------------------
' End of Program
'-----------------------------------------------------------------------
```

D.2 PICAXE Code Listing for Node Master

```
' NodeMaster
' Written by:  Alison McCarthy

' Purpose:
'  To modulate and process the demodulated data from the node's PICAXE-08M.

' Inputs:
'  RTS - Input from PICAXE-08M to indicate that data is ready to be sent from slave
to master
' SENSOR - Data from sensor
' RXPIN - Receives XR2211's demodulated data from slave
' TXSTATUS - Input of radio channel to determine if channel is idle


' Outputs:
'  CTS - Transmits signal to slave indicating it is clear to send demodulated data
```

```
' TXPIN - Transmits frequency shift keying to modulator XR2206
' CBSW - Activates relay that connects 'on' switch of UHF citizen band radio
' CB6V - Connects power (+5V) to UHF citizen band radio
' CBTX - Activates relay that connects TX switch of UHF citizen band radio

' Constants:
'  PREAMBLEBYTE - Preamble byte
' SOMBYTE - Start of message byte
' EOMBYTE - End of message byte
' POSTAMBLEBYTE - Postamble byte
' TXBAUD - Baud rate of data transmitted over UHF citizen band radio
' RXBAUD - Baud rate of data received over UHF citizen band radio
' BAUD - Baud rate of transmission between master and slave
' POLYNOMIAL - Polynomial used for crc error checking
' dataLength - Length of transmitted data byte

' Variables:
'  RECEIVERCODE - Identifier of node to receive data
' SENDERCODE - Identifier of central node
' RECEIVERCODE2 - Demodulated identifier of node to receive data (central node)
' SENDERCODE2 - Demodulated identifier of node sending the data
' byte - Data transmitted
' k - Variable used to determine error checking crc
' crc - Calculated crc value
' status - Analogue value used to determine if radio is busy
' temp - Variable for temporary storage
' flag - Set when node is polled
' counter - Counter used to determine the number of poll attempts
' seconds - RTC seconds
' mins - RTC minutes
' hour - RTC hours
' date - RTC date
' month - RTC month
' year - RTC year

'----------------------------------------------------------------------
' Assign symbol names to PICAXE-40X inputs and outputs
```

```
' Inputs
symbol SENSOR = 0 ' Analogue input 0, pin 2
symbol DataI = 3 ' Input 3, pin 22

' Outputs
symbol TXPIN = 5 ' Output 5, pin 39
symbol CB6V = 2 ' Portc output 2, pin 17
symbol CBSW = 1 ' Portc output 1, pin 16
symbol CBTX = 0 ' Portc output 0, pin 15

'----------------------------------------------------------------------
' Assign symbol names to constants

symbol PREAMBLEBYTE = $AA
symbol SOMBYTE = $C5
symbol EOMBYTE = $5C

symbol TXBAUD = N600
symbol BAUD = T2400
symbol INCREMENT = 30 ' Radios are polled every 30 minutes

symbol POLYNOMIAL = $8C

'----------------------------------------------------------------------
' Assign symbol names to variables

symbol byte = b0
symbol k = b1
symbol crc = b2

symbol SENDERCODE = b9 ' 0 ..255
symbol SENDERCODE2 = b7 ' 0 ..255
symbol RECEIVERCODE = b10 ' 0 ..255
symbol RECEIVERCODE2 = b6 ' 0 ..255
symbol POLL = b11
symbol ACK = b12

'----------------------------------------------------------------------
' Main program listing
```

```
Main:  gosub GetRTC ' Read current date and time
temp1 = time-1 ' Time expected for node to be polled
temp2 = time+1 ' Time expected for node to be polled

' Check if time to wake to receive poll
if mins > temp1 or mins < temp2 then Turnon

' If node does not expect to be polled, pause for one second and go back to main
pause 1
goto Main

'-----------------------------------------------------------------------
' Turn on radio ready to receive data

Turnon:

' Turn on radio ready to receive data
high portc CB6V ' Connect power to UHF CB radio
wait 4 ' Wait 4 seconds
high portc CBSW ' Hold 'on' switch of UHF CB radio
wait 3 ' Wait 3 seconds
low portc CBSW ' Disconnect 'on' switch of UHF CB radio
wait 1 ' Wait 1 second

low TXPIN ' Prevent floating pins
setint %00001000,%00001000 ' Activate interrupt when Input 3, RXPIN, goes high

Here:  if flag = 1 then Main ' If poll received, go back to Main
goto Here ' Stay in this loop until a message addressed to this node is received

'-----------------------------------------------------------------------
' Interrupt routine (data has been received by demodulator)

Interrupt:
serin DataI,BAUD,(PREAMBLEBYTE,SOMBYTE),RECEIVERCODE,SENDERCODE,
POLL,RECEIVERCODE2,ACK,byte,crc2

' If node has received poll from base
if POLL = $FF and ACK = $00 and byte = $00 then HandlePoll
```

```
' If node has received ACK from base
if POLL = $00 and ACK = $FF and byte = $00 then HandleACK

' Else data is unrecognisable
EndInterrupt:
return ' Return from interrupt

'----------------------------------------------------------------------
' Send data if poll is received

HandlePoll:
'  Send data to base station
POLL = $00
ACK = $00

gosub CreatePacket
gosub Transmit
goto EndInterrupt

'----------------------------------------------------------------------
' Turn off radio if ACK is received

HandleACK:
flag = 1 ' Set flag for receiving poll
setint %00001000,%00001000 ' Reactivate interrupt when Input 3, RXPIN, goes high
goto EndInterrupt

'----------------------------------------------------------------------
' Prepare a packet to send to the central node

CreatePacket:

' Read data from advance sensor
readadc SENSOR,byte
temp2 = byte

' Prepare data integrity byte to be sent in the message
crc = 0 ' Calculate crc
for temp3 = 0 to 7 ' Crc add
```

```
k = temp2 ∧ crc & 1
if k = 0 then CrcAdd1
k = POLYNOMIAL
CrcAdd1:
crc = crc / 2 ∧ k
temp2 = temp2 / 2
next temp3

POLL = $00
ACK = $00

return ’ Return from subroutine

’-----------------------------------------------------------------------
’ Transmit data over radio

Transmit:
counterTX = counterTX + 1
pause 100
high portc CBTX ’ Hold on the transmit switch of the UHF citizen band radio
pause 100 ’ Pause for 100 ms before sending data


’ For crc:
serout TXPIN,TXBAUD,(PREAMBLEBYTE,PREAMBLEBYTE,PREAMBLEBYTE,PREAMBLEBYTE,
PREAMBLEBYTE,SOMBYTE,RECEIVERCODE,SENDERCODE,data,crc,EOMBYTE)

’ For repetition code:
serout TXPIN,TXBAUD,(PREAMBLEBYTE,PREAMBLEBYTE,PREAMBLEBYTE,PREAMBLEBYTE,
PREAMBLEBYTE,SOMBYTE,RECEIVERCODE,SENDERCODE,POLL,ACK,byte,byte,
byte,byte,byte,EOMBYTE)

low portc CBTX ’ Release the transmit switch of the UHF citizen band radio

setint %00001000,%00001000 ’ Reactivate interrupt when Input 3, RXPIN, goes high
return ’ Return from subroutine

’-----------------------------------------------------------------------
’ Get date and time from real time clock
```

```
GetRTC:
' This code is obtain date and time from RTC
i2cslave %11010000,i2cslow,i2cbyte ' Set DS1307 slave address
pause 100 ' Wait 100 ms

poke seconds,$00 ' 00 Note all BCD format
poke mins,$59 ' 59 Note all BCD format
poke hour,$11 ' 11 Note all BCD format
poke date,$25 ' 25 Note all BCD format
poke month,$07 ' 12 Note all BCD format
poke year,$06 ' 03 Note all BCD format

' Write time and date
writei2c 0,(seconds,mins,hour,date,date,month,year,%00010000)
return

'----------------------------------------------------------------------
' End of Program
'----------------------------------------------------------------------
```

D.3 PICAXE Code Listing for Base Buffer

```
' Base Buffer
' Written by:  Alison McCarthy

' Purpose:
'  To receive demodulated data from XR2211 from radio channel

' Inputs:
'  RXPIN - Receives demodulated data to the central node master


' Outputs:
'  DataO - Outputs data demodulated by XR2211

' Constants:
'  PREAMBLEBYTE - Preamble byte
```

```
' SOMBYTE - Start of message byte
' EOMBYTE - End of message byte
' POSTAMBLEBYTE - Postamble byte
' RXBAUD - Baud rate of data received over UHF citizen band radio
' BAUD - Baud rate of transmission between master and slave
' SENDERCODE - Identifier of node

' Variables:
'  data 1 - Data byte 1
' data 2 - Data byte 2
' data 3 - Data byte 3
' data 4 - Data byte 4
' data 5 - Data byte 5
' data - Corrected Data byte
' RECEIVERCODE - Identifier of node to receive data
' RECEIVERCODE2 - Identifier of node to receive data
' SENDERCODE2 - Identifier of node that is sending the data
' TXBAUD - Baud rate of transmission

'-----------------------------------------------------------------------
' Assign symbol names to PICAXE-08M inputs and outputs

symbol DataO = 4 ' Output 4, pin 3, receives data to be modulated from the master
symbol RXPIN = 1 ' Input 1, pin 6, output data to the modulator XR2206 to transmit
over the radio

'-----------------------------------------------------------------------
' Assign symbol names to constants

symbol PREAMBLEBYTE = $AA
symbol SOMBYTE = $C5
symbol EOMBYTE = $5C symbol SENDERCODE = 10

symbol RXBAUD = N600 ' Receive data at 600 baud
symbol BAUD = T2400 ' Transfer data to master at 2400 baud

'-----------------------------------------------------------------------
' Assign symbol names to variables
```

```
symbol RECEIVERCODE = b0 ' 0 ..255 stored in b1
symbol RECEIVERCODE2 = b1 ' 0 ..255 stored in b1
symbol SENDERCODE2 = b2 ' 0 ..255 stored in b2
symbol ACK = b3
symbol POLL = b4
symbol byte = b5

'-----------------------------------------------------------------------

Loop:  '  Read in data received by demodulator
' For crc:
serin RXPIN,RXBAUD,(PREAMBLEBYTE,SOMBYTE),RECEIVERCODE,SENDERCODE2,data,crc

' For repetition code:
serin RXPIN,RXBAUD,(PREAMBLEBYTE,SOMBYTE),RECEIVERCODE,SENDERCODE,
data1,data2,data3,data4,data5

' Test if three repetition codes are sufficient:
if data1=data2 or data1=data3 then Data1Code
if data2=data1 or data2=data3 then Data2Code
if data3=data1 or data3=data2 then Data3Code

Data1Code:
data=data1
goto CheckCode

Data2Code:
data=data2
goto CheckCode

Data3Code:
data=data3
goto CheckCode

CheckCode:
if RECEIVERCODE <> SENDERCODE2 then Loop ' If the poll is not addressed to this
node then go back to start

' Test if three repetition codes, rather than five can work
```

```
if (data1=data2 and data2=data3) or (data1=data2 and data1=data3) or (data2=data3
and data1=data3) then WorkOutData

' Set flag to send activate interrupt on master PICAXE
high Data0 ' Set flag on pin of interrupt
pause 20 ' Wait for master PICAXE to respond

' Transfer data to master PICAXE
' For crc:
serout Data0,BAUD,(PREAMBLEBYTE,PREAMBLEBYTE,PREAMBLEBYTE,PREAMBLEBYTE,
PREAMBLEBYTE,SOMBYTE,RECEIVERCODE,SENDERCODE,data,crc,EOMBYTE)

' For repetition:
serout Data0,BAUD,(PREAMBLEBYTE,PREAMBLEBYTE,PREAMBLEBYTE,PREAMBLEBYTE,
PREAMBLEBYTE,SOMBYTE,RECEIVERCODE,SENDERCODE,data,EOMBYTE)

low Data0 ' Turn off flag of interrupt pin
goto Loop ' Go back to start to receive more data

'------------------------------------------------------------------------
' End of Program
'------------------------------------------------------------------------
```

D.4 PICAXE Code Listing for Base Master

```
' Base Master
' Written by:  Alison McCarthy

' Purpose:
'  To modulate and process the demodulated data from the base's PICAXE-08M.

' Inputs:
'  SENSOR - Data from sensor
' RXPIN - Receives XR2211's demodulated data from slave

' Outputs:
'  TXPIN - Transmits frequency shift keying to modulator XR2206
```

```
' CBSW - Activates relay that connects 'on' switch of UHF citizen band radio
' CB6V - Connects power (+5V) to UHF citizen band radio
' CBTX - Activates relay that connects TX switch of UHF citizen band radio
' TXPC - Outputs received data serially to be displayed on user interface

' Constants:
'  PREAMBLEBYTE - Preamble byte
' SOMBYTE - Start of message byte
' EOMBYTE - End of message byte
' TXBAUD - Baud rate of data transmitted over UHF citizen band radio
' RXBAUD - Baud rate of data received over UHF citizen band radio
' BAUD - Baud rate of transmission between master and slave
' POLYNOMIAL - Polynomial used for crc error checking
' dataLength - Length of transmitted data byte

' Variables:
'  RECEIVERCODE - Identifier of node to receive data
' SENDERCODE - Identifier of central node
' RECEIVERCODE2 - Demodulated identifier of node to transmit data
' SENDERCODE2 - Demodulated identifier of node sending the data
' byte - Data transmitted
' k - Variable used to determine error checking crc
' crc - Calculated crc value
' status - Analogue value used to determine if radio is busy
' counter - Counter used to determine the number of poll attempts
' seconds - RTC seconds
' mins - RTC minutes
' hour - RTC hours
' date - RTC date
' month - RTC month
' year - RTC year

'----------------------------------------------------------------------
' Assign symbol names to PICAXE-40X inputs and outputs

' Inputs
symbol SENSOR = 0 ' Analogue input 0, pin 2
symbol RXPIN = 3 ' Input 3, pin 22
```

```
' Outputs
symbol TXPIN = 0 ' Output 5, pin 39
symbol CB6V = 2 ' Portc output 2, pin 17
symbol CBSW = 1 ' Portc output 1, pin 16
symbol CBTX = 0 ' Portc output 0, pin 15
symbol TXPC = 3 ' Output 0, pin 33

'-----------------------------------------------------------------------
' Assign symbol names to constants

symbol PREAMBLEBYTE = $AA ' Preamble byte is alternating zeros and ones
symbol SOMBYTE = $C5 ' Start of message byte
symbol EOMBYTE = $5C ' End of message byte

symbol TXBAUD = N600 ' Modulation of transmitted data at 600 baud
symbol BAUD = T2400 ' Receiving demodulated data from buffer PICAXE at 2400 baud
symbol INCREMENT = 15 ' Radios are polled every 30 minutes
symbol NODES = 40 ' 40 nodes in network
symbol SENDERCODE = 10 ' Assigned sender id code
symbol time = 1 ' For node number 10 (if ten nodes are polled per second)

symbol POLYNOMIAL = $8C ' Polynomial byte for crc calculation

'-----------------------------------------------------------------------
' Assign symbol names to variables

symbol byte = b0
symbol k = b1
symbol crc = b2 ' Crc received in message from node
symbol crc2 = b3 ' Crc calculated using data transmitted in node


symbol counterP = b4 ' Poll counter
symbol counter = b5 ' Node counter
symbol temp1 = b8 ' Temporary storage byte
symbol temp2 = b9 ' Temporary storage byte 2


symbol SENDERCODE2 = b9 ' Sender identification number between 0 and 255
```

```
symbol RECEIVERCODE = b10 ' Receiver identification number between 0 and 255
symbol RECEIVERCODE2 = b8 ' Receiver identification number between 0 and 255
symbol POLL = b11 ' Poll byte sent in message ($FF if poll message, $00 if not poll
message)
symbol ACK = b12 ' Acknowledgment byte ($FF is ACK message, $00 if not ACK message)
symbol seconds = b6 ' Seconds from RTC
symbol mins = b7 ' Minutes from RTC
symbol hour = b8 ' Hour from RTC
symbol date = b9 ' Date from RTC
symbol month = b10 ' Month from RTC
symbol year = b11 ' Year from RTC


'-----------------------------------------------------------------------
' Main program

Main:  gosub GetRTC ' Read current date and time
temp1 = time-1 ' Time expected for node to be polled
temp2 = time+1 ' Time expected for node to be polled

' Check if time to wake to receive poll
if mins > temp1 or mins < temp2 then Turnon

' If node does not expect to be polled, pause for one second and go back to main
pause 1
goto Main

low TXPIN ' Initialise pin output to prevent floating pins
counter = 0 ' Initialise counter

'-----------------------------------------------------------------------
Here:  counterP = 0 ' Initialise poll counter
temp = 0 ' Initialise indicating variable
counter = counter + 1 ' Increment node counter

Loop:  '  Send poll to node if less than five polls have already been sent
counterP = counterP + 1 ' Increment poll counter
if counterP>5 then EndHere ' If node has been polled, poll next node
```

```
POLL = $FF
ACK = $00
byte = 0
gosub CreateErrorByte
gosub Transmit

' Wait to receive data from node
wait 4

' If no data is received, send poll again (temp = 1 if data is received)
if temp <> 1 then Loop

EndHere:  goto Here

'-----------------------------------------------------------------------
' Transmit data encoded as audio tones over radio

Transmit:
pause 500 ' Pause after receiving message
high portc CBTX ' Press transmit button
pause 500
serout TXPIN,TXBAUD,(PREAMBLEBYTE,PREAMBLEBYTE,PREAMBLEBYTE,
PREAMBLEBYTE,PREAMBLEBYTEBYTE,SOMBYTE,TXP,RXP,POLL,RXA,ACK,EOMBYTE)
low portc CBTX ' End transmission
setint %00001000,%00001000 ' Reactivate interrupt ready to receive acknowledgment
goto EndOfSendPoll ' Return to subroutine
return

'-----------------------------------------------------------------------
' Interrupt routine to process received data from node

Interrupt:
'  For cyclic redundancy code:
serin RXPIN,BAUD,(PREAMBLEBYTE,SOMBYTE),TXP2,RXP2,byte,crc2

' For repetition code:
serin RXPIN,BAUD,(PREAMBLEBYTE,SOMBYTE),TXP2,RXP2,byte1,byte2,byte3,byte4,byte5

' If base has received data from node send ACK to node and poll to next node
```

```
if POLL = $00 and ACK = $00 and byte <> $00 then HandleData

' Else data is unrecognisable
EndInterrupt:
return ' Return from interrupt

'-----------------------------------------------------------------------
' Data handling code

HandleData:


' For cyclic redundancy code:
gosub CreateErrorByte

' If error checking did not detect errors, send ACK
if crc=crc2 then NoErrors temp = 0 ' Node is polled again if temp <> 1 goto EndInterrupt

NoErrors:
temp = 1
gosub SendACK
goto EndInterrupt

'-----------------------------------------------------------------------
SendACK:
' Send ACK to node
POLL = $00
ACK = $FF
byte = 0
gosub CreateErrorByte
gosub Transmit
goto EndInterrupt

'-----------------------------------------------------------------------
' Create cyclic redundancy code for received data
' (written by 'Happy Hippy' at http://www.hippy.freeserve.co.uk/picaxewf.htm)

CreateErrorByte:
```

```
ErrorCheck:
crc = 0
temp = byte

CrcAdd:
for counter = 0 to 7
k = byte ∧ crc & 1
if k = 0 then CrcAdd1
k = POLYNOMIAL

CrcAdd1:  crc = crc / 2 ∧ k
byte = byte / 2
next

if byte = crc then EndErrorCheck
Failed:
temp=$EE ' Indicate that poll must be sent again to receive data again
serout(PREAMBLEBYTE,PREAMBLEBYTE,PREAMBLEBYTE,PREAMBLEBYTE,PREAMBLEBYTE,SENDERCODE)

EndErrorCheck:
return

'-----------------------------------------------------------------------
' Get date and time from real time clock

GetRTC:
' This code is obtain date and time from RTC
i2cslave %11010000,i2cslow,i2cbyte ' Set DS1307 slave address
pause 100 ' Wait 100 ms

poke seconds,$00 ' 00 Note all BCD format
poke mins,$59 ' 59 Note all BCD format
poke hour,$11 ' 11 Note all BCD format
poke date,$25 ' 25 Note all BCD format
poke month,$07 ' 12 Note all BCD format
poke year,$06 ' 03 Note all BCD format

' Write time and date
writei2c 0,(seconds,mins,hour,date,date,month,year,%00010000)
```

```
return

'----------------------------------------------------------------------
' End of Program
'----------------------------------------------------------------------
```

Appendix E

Additional Circuits

E.1 Radio Shutdown Device

A normal transmission will not exceed three seconds, in accordance with Citizen Band Radio Stations Class Licence restrictions. A radio shutdown device is required for when the master PICAXE falls into an endless loop and continuously transmits.

Hence an additional PICAXE-08M is required for a shutdown function, with the sole purpose of timing three seconds and sending a turn-off command that disconnects the power to the radio (see Figure E.1).

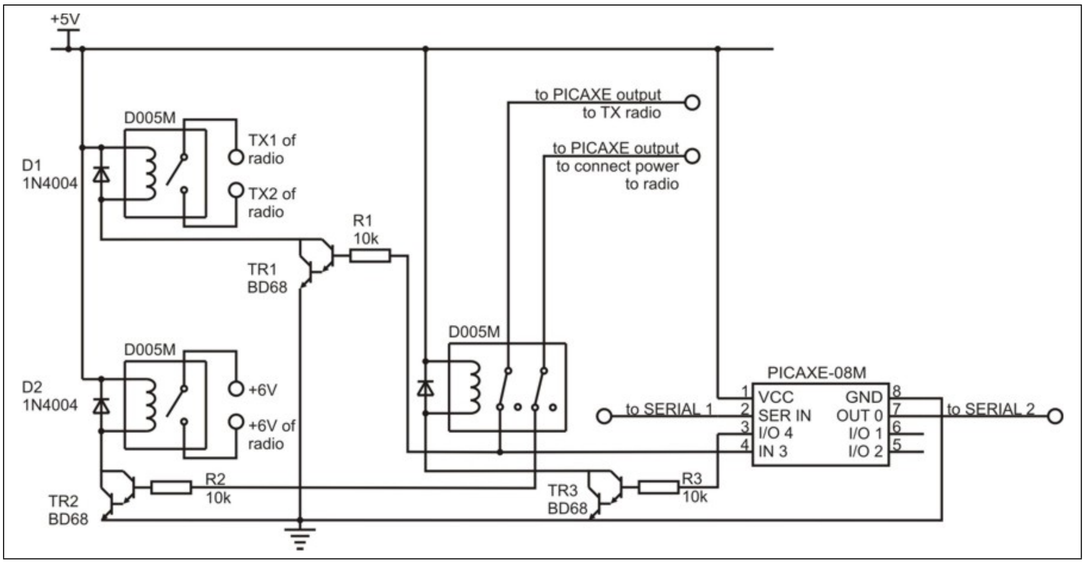


Figure E.1: Schematic of additional circuitry to shut off radio after three seconds of continuous transmission

The base station will not receive any further signals from the node and hence an error will be logged to the sensors. The corresponding PICAXE-08M code is:

```
symbol TX = pin3
symbol Disconnect = pin4
symbol timer = b0
symbol inc = b1
symbol time = 3000

main:
   setint %00001000,%00001000

Loop2:
```

```
  goto Loop2

Interrupt:
  timer = 0 ' Initialise variable to monitor time waited
  inc = time/250 ' Intervals when status of transmission is checked

  Loop:
    if TX = 0 then EndInt ' If the transmission has stopped exit interrupt

    ' Else if radio transmission has not stopped, pause for interval
    ' Time required for other commands in loop are negligible (take < 1 ms)
    pause inc

    ' Update time waited
    timer = timer + inc

    if timer > 3 then StopTX ' If three seconds has passed
    goto Loop

  StopTX:
    high Disconnect ' Energise relay to disconnect radio and stop transmission

  EndInt:
    low Disconnect ' Keep radio connected to circuit

    return ' Return from interrupt
```

E.2 Interface with Other Wireless Technology

An interfacing circuit between the existing node circuit and an XBee Zigbee module is shown in Figure E.2, and an individual Zigbee module in Figure E.3.

Figure E.2: XBee Zigbee Pro module (overall size is 32.94 mm high and 24.38 mm wide)

Source: *PICAXE Connect (AXE210)* (2006)

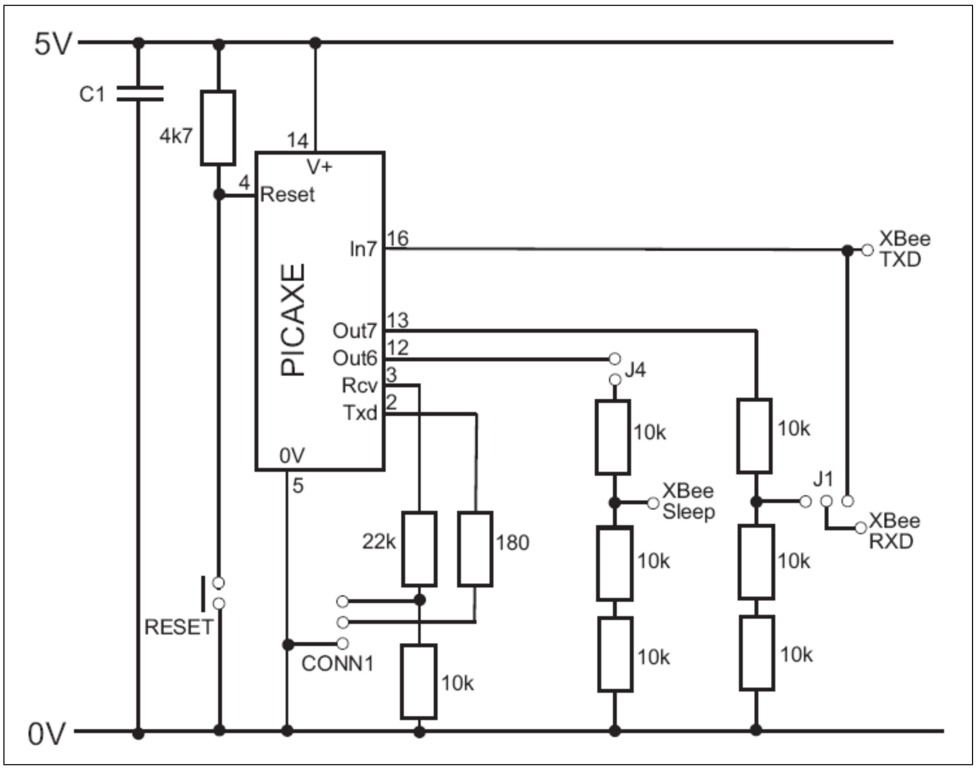


Figure E.3: Schematic of interfacing circuit between PICAXE and XBee Zigbee module

Source: *PICAXE Connect (AXE210)* (2006)

E.3 Global Positioning System

A GPS module available as a kit with the PICAXE-18X is the LS-40EB (see Figure E.4). The LS-40EB operates from 3.3 V and has 5 m accuracy. Therefore a 3.3 V regulator is also required in the circuit. The LS-40EB module costs $70, while an antenna to suit the GPS module costs $36. The circuit that interfaces the GPS module to a

PICAXE-18X is shown in Figure E.4, with the interfacing circuit shown in Figure E.5.

Figure E.4: LS-40EB GPS module

Source: *PICAXE Connect (AXE210)* (2006)

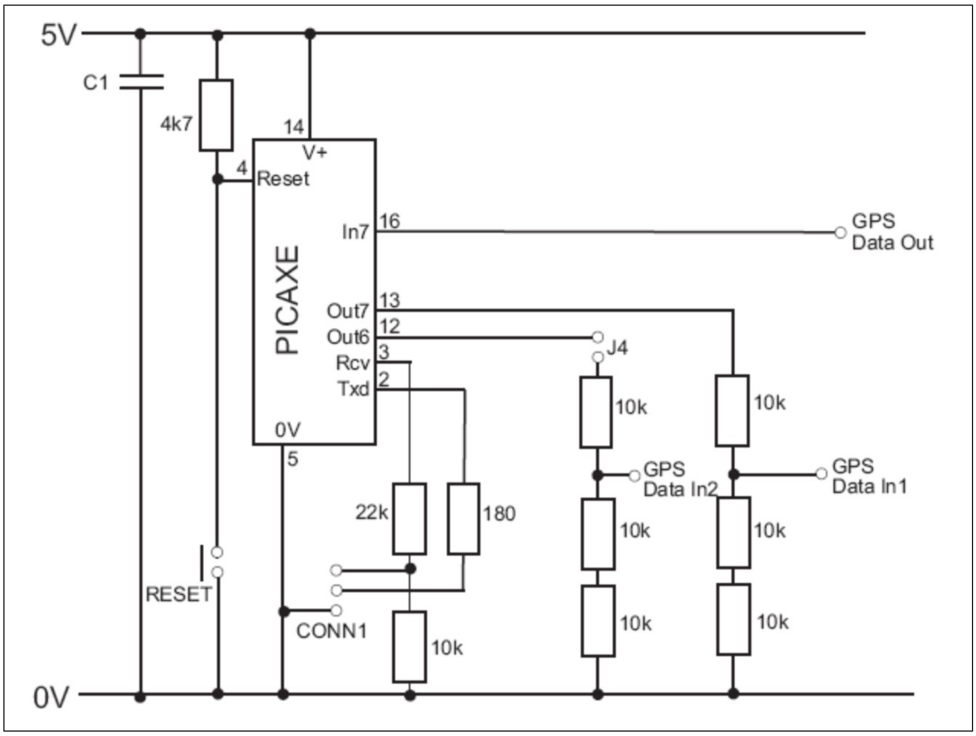


Figure E.5: Schematic of GPS circuit

Source: *PICAXE Connect (AXE210)* (2006)

E.4 Sensor Identification

The following circuit could be used to provide an interface for changing node identification numbers.

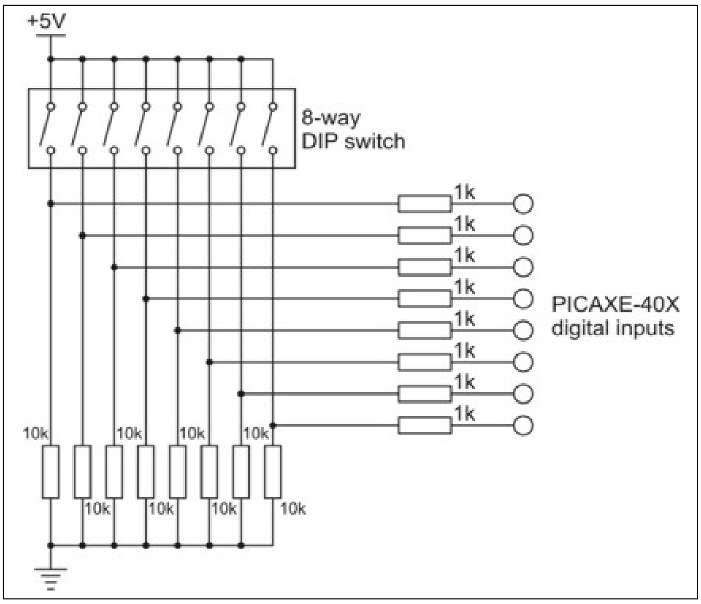


Figure E.6: Schematic of circuit to interface DIP switch to PICAXE inputs

The following line of code is required for the PICAXE to obtain the specified identification number.

```
let b13 = pins ' Determine identification number of sensor node
```

Appendix F

Calculations

F.1 Electrical Current Usage of Circuit

The average current draw (I_{avg}) is calculated based on the current drawn values taken from the circuit. The electrical current usage of the circuit using four Nickel Metal-Hydride batteries is repeated below from Table 7.2 in Chapter 7.

Table F.1: Electrical current requirements of node hardware with step up regulator using 0.5 W radio

Activity	Current Usage	Time/hour	Fraction of time	Current
Radio off	74 mA	50 minutes	0.833	61.67 mA
Radio on and idle	122 mA	~5 minutes	0.0833	10.17 mA
Radio on and receiving	164 mA	~5 minutes	0.0833	13.67 mA
Radio on and transmitting	300 mA	300 ms	0.000005	0.0015 mA
			Total	85.50 mA

The MAX642 step up regulator was then removed, and the circuit was powered from a 12 V lead acid battery.

Table F.2: Electrical current requirements of node hardware without step up regulator using 0.5 W radio

Activity	Current Usage	Time/hour	Fraction of time	Current
Radio off	17 mA	50 minutes	0.833	14.17 mA
Radio on and idle	65 mA	~5 minutes	0.0833	5.42 mA
Radio on and receiving	75 mA	~5 minutes	0.0833	6.25 mA
Radio on and transmitting	230 mA	300 ms	0.000005	0.0015 mA
			Total	25.83 mA

F.1.1 Solar Panel Calculations

Using equations from (*Technical Information for Choosing Solar Module* 2006), I_{app} is the duty cycle of the current draw of the application and L_{avg} is the average illumination on the solar module, and has been assumed to be 8 hours each day.

$$\begin{aligned} L_{avg} &= 100\% \times \text{ equivalent hours of full sun per day} \\ &= 100\% \times \frac{8}{24} = 33.3\% \end{aligned}$$

$$\begin{aligned} I_{module} &= I_{app} \times \frac{100}{L_{avg}} \\ &= 85.50 \times \frac{100}{33.3} \\ &= 256.5 \text{ mA} \end{aligned}$$

Therefore a 256.5 mA solar panel would be required. After removing the step up regulator, replacing the four Nickel Metal-Hydride batteries with a 12 V lead acid battery, the following solar panel calculation was performed.

$$\begin{aligned} I_{module} &= I_{app} \times \frac{100}{L_{avg}} \\ &= 25.83 \times \frac{100}{33.3} \\ &= 77.49 \text{ mA} \end{aligned}$$

Therefore a 100 mA solar panel may be used to charge the batteries.

F.2 External Timing Component Values for XR2206 and XR2211

The XR2206 modulator and XR2211 demodulator must be tuned to encode and decode specific audio tone frequencies. The external components used to tune the modem to 1200 Hz for a logic 0 and 2200 Hz for a logic 1 were based on the values from *1200 Baud Data Modems* (2006).

F.2.1 Tuning the XR2206 Modulator

New external components were required for the XR2206 modulator to output audio tone frequencies of 2600 Hz for a logic 0 and 3600 Hz for a logic 1 (for an increased data rate). Using the equation $f_0 = \frac{1}{RC}$ from the XR2206 data sheet G, the resistances on pin 7 and 8 (to tune logic 0 and 1 frequencies, respectively) are:

$$
\begin{aligned}
f_0 &= \frac{1}{RC} \text{ where } C = 33n \\
2600 &= \frac{1}{33 \times 10^{-9} \times R} \text{ for the frequency representing a logic 0} \\
R &= 11.66k\Omega \rightarrow \text{Choose } R = 12k\Omega
\end{aligned}
$$

$$
\begin{aligned}
f_{logic1} &= \frac{1}{RC} \text{ where } C = 33n \\
3600 &= \frac{1}{33 \times 10^{-9} \times R} \text{ for the frequency representing a logic 0} \\
R &= 8.42k\Omega \rightarrow \text{Choose } R = 10k\Omega
\end{aligned}
$$

Therefore, to change the modulated frequency from 1200 Hz to 2600 Hz, the 10 kΩ resistor on pin 7 of the XR2206 is replaced with a 12 kΩ resistor. Similarly, to change the modulated frequency from 2200 Hz to 3600 Hz, the 22 kΩ resistor on pin 8 of the XR2206 is replaced with a 10 kΩ resistor.

F.2.2 Tuning the XR2211 Demodulator

Now the XR2211 demodulator must be tuned for a centre frequency of $f_0 = 0.5(2600 + 3600) = 3100$. The new resistor value at pin 12 of the demodulator will be:

$$f_c = \frac{1}{RC} \text{ where } C = 33n$$

$$3100 = \frac{1}{33 \times 10^{-9} \times R}$$

$$R = 9.78k\Omega \rightarrow \text{Choose } R = 10k\Omega$$

Therefore, to change the demodulated centre frequency from 1700 Hz to 3100 Hz, the 15 kΩ resistor on pin 12 of the XR2211 is replaced with a 10 kΩ resistor.

Appendix G

Data Sheets

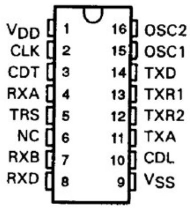
Selected pages of the data sheets for the TCM3105 and FX614 frequency shift-keying modem, XR2206 modulator and XR2211 demodulator are included, downloaded from 'DatasheetCatalog.com' (http://www.datasheetcatalog.com).

TCM3105DWE, TCM3105DWL, TCM3105JE TCM3105JL, TCM3105NE, TCM3105NL FSK MODEM

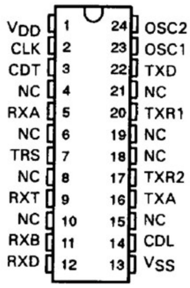
D2862, NOVEMBER 1985–REVISED DECEMBER 1990

- Single-Chip Frequency-Shift-Keying (FSK) Modem
- Meets Both Bell 202 and CCITT V23 Specifications
- Transmit Modulation at 75, 150, 600, and 1200 Baud
- Receive Demodulation at 5, 75, 150, 600, and 1200 Baud
- Half-Duplex Operation Up to 1200 Baud Transmit and Receive
- Full-Duplex Operation Up to 1200 Baud Transmit and 150 Baud Receive
- On-Chip Group Delay Equalization and Transmit/Receive Filtering
- Carrier-Detect-Level Adjustment and Carrier-Fail Output
- Single 5-V Power Supply
- Low Power Consumption
- Reliable CMOS Silicon-Gate Technology

J OR N PACKAGE
(TOP VIEW)



DW PACKAGE
(TOP VIEW)



NC–No internal connection

D packages are available taped and reeled. Add "R" suffix to device type (e.g., TCM3105DWLR)

Caution. These devices have limited built-in gate protection. The leads should be shorted together or the device placed in conductive foam during storage or handling to prevent electrostatic damage to the MOS gates.

description

The TCM3105 is a single-chip asynchronous Frequency Shift Keying (FSK) voiceband modem that uses silicon gate CMOS technology to implement a switched capacitor architecture. It is pin selectable (TXR1, TXR2, and TRS inputs) for a wide range of transmit/receive baud rates and is compatible with the applicable BELL 202 or CCITT V23 standards. Operation is fully reversible, thereby allowing both forward and backward channels to be used simultaneously.

The transmitter is a programmable frequency synthesizer that provides two output frequencies (on TXA), representing the 'marks' and 'spaces' of the digital signal present on the TXD input.

The receive section is responsible for the demodulation of the analog signal appearing at the RXA input and is based on the principle of frequency-to-voltage conversion. This section contains a group delay equalizer (to correct phase distortion), automatic gain control, carrier detect level adjustment, and bias distortion adjustment, thereby optimizing performance and giving the lowest possible bit error rate.

TABLE 1. MODES OF OPERATION

STANDARD	TRS	TXR1	TXR2	TRANSMITTED BAUD RATE	RECEIVED BAUD RATE	TRANSMIT FREQUENCY ASSIGNMENTS (Hz)	RECEIVE FREQUENCY ASSIGNMENTS (Hz)	CLK FREQUENCY (kHz)
CCITT V.23	L	L	L	1200	1200	M 1300 S 2100	M 1300 S 2100	19.11
	H	L	L	1200	75	M 1300 S 2100	M 390 S 450	19.11
	L	L	H	600	75	M 1300 S 1700	M 390 S 450	9.56
	H	L	H	600	600	M 1300 S 1700	M 1300 S 1700	9.56
	L	H	L	75	1200	M 390 S 450	M 1300 S 2100	19.11
	H	H	L	75	600	M 390 S 450	M 1300 S 1700	9.56
	L	H	H	75	75	M 390 S 450	M 390 S 450	1.19
BELL 202	$\overline{CLK}$	L	L	1200	1200	M 1200 S 2200	M 1200 S 2200	19.11
	$\overline{CLK}$/8	L	H	1200	150	M 1200 S 2200	M 387 S 487	19.11
	$\overline{CLK}$/8	L	H	1200	5	M 1200 S 2200	M 387 S 0	19.11
	CLK	H	L	150	1200	M 387 S 487	M 1200 S 2200	19.11
	CLK	H	H	150	150	M 387 S 487	M 387 S 487	2.39
	CLK† H†	H† H†	L† H†	5	1200	M 387 S 0	M 1200 S 2200	19.11
	H	H	H	Transmit Disabled	1200	Transmit Disabled	M 1200 S 2200	19.11

H = high level, L = low level
†In these modes, the modulation is controlled by the TRS and TXR2 pins. TXD is tied high.

MX·COM, INC.
COMMUNICATION SEMICONDUCTORS

MX614

DATA BULLETIN — *Bell 202 Compatible Modem*

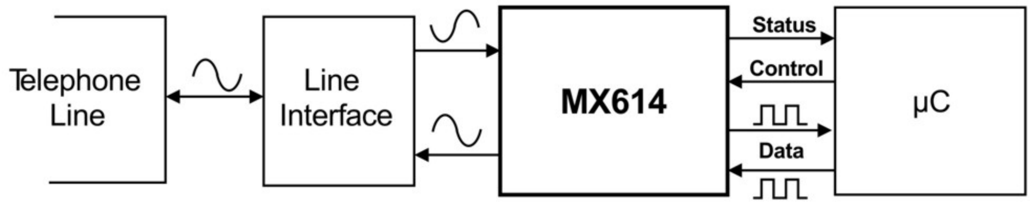
PRELIMINARY INFORMATION

Features

- **1200bps - 1800bps half duplex Bell 202 Compatible Modem**
- **Optional 1200bps Data Retiming Facility can eliminate external UART**
- **Optional 5bps and 150bps Back Channel**
- **Optional Line Equalization**

Applications

- **Low Voltage Operation (3.3V to 5.0V)**
- **Low Power Operation**
  1mA typ. @ 3.3V Operating Mode
  1µA typ. Zero-Power Mode
- **Standard 3.58MHz Xtal/Clock**
- **Telephone Telemetry Applications**



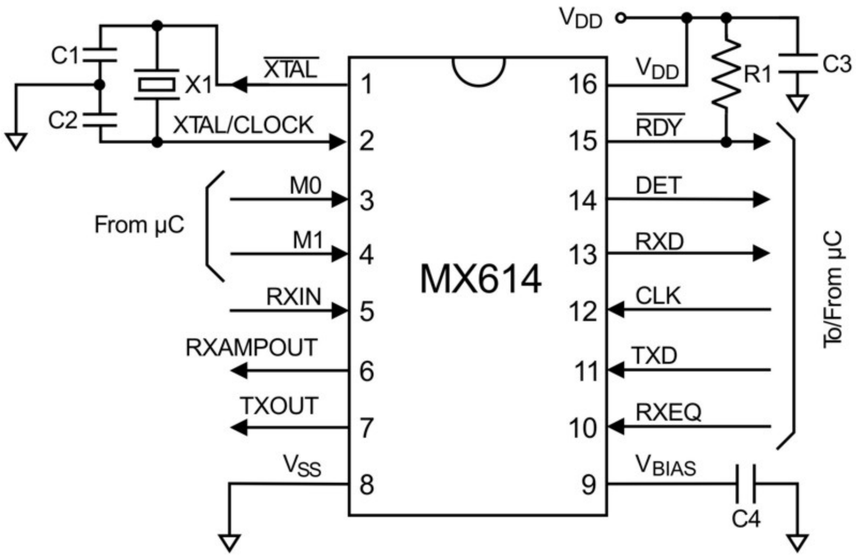
The MX614 is a low voltage, low power CMOS integrated circuit designed for the reception or transmission of asynchronous 1200bps data. This device is compatible with Bell 202 type systems. The MX614 supports 5bps and 150bps 'back channel' operation. Asynchronous data rates up to 1818bps are also supported.

The MX614 provides an optional Tx and Rx data retiming function which can eliminate, based on user preference, the need for a UART in the associated µC when operating at 1200bps. An optional line equalizer has been incorporated into the receive path and is controlled by an external logic level.

The MX614 may be used in a wide range of telephone telemetry systems. A very low current □Zero Power Mode (1µ A typ.) and an operating current of 1mA typ. @ V_{DD} = 3.3V, make the MX614 ideal for portable, terminal and line powered applications. A standard 3.58MHz Xtal/Clock is required and the device operates from a 3.0V to 5.5V supply.

The MX614 is available in 24-pin TSSOP (MX614TN), 16-pin SOIC (MX614DW) and 16-pin PDIP (MX614P) packages.

3. External Components



R1		100kΩ	±5%
C1 C2		18pF	±10%
C3		0.1µF	±10%
C4		0.1µF	±10%
X1	Note 1	3.579545MHz	

Figure 2: Recommended External Components for Typical Application

External Components Notes

1. **IMPORTANT:** This device is capable of detecting and decoding small amplitude signals. To achieve this V_{DD} and V_{BIAS} decoupling and protecting the receive path from extraneous in-band signals are very important. It is recommended that the decoupling capacitors be placed so that connections between them and the device pins are as short as practicable e.g. ≤ 1 inch from device pins. A ground plane protecting the receive path will help attenuate interfering signals
2. A crystal frequency of 3.579545MHz ±0.1% is required for correct FSK operation. For best results, a crystal oscillator design should drive the clock inverter input with signal levels of at least 40% of V_{DD} peak-peak. Tuning fork crystals generally cannot meet this requirement. To obtain crystal oscillator design assistance, consult your crystal manufacturer.

4. General Description

4.1 Xtal Osc and Clock Dividers

Frequency and timing accuracy of the MX614 is determined by a 3.579545MHz clock signal present at the XTAL/CLOCK pin. This may be generated by the on-chip oscillator inverter using the external components C1, C2 and X1 of Figure 2, or may be supplied from an external source to the XTAL/CLOCK input. If supplied from an external source, C1, C2 and X1 should not be fitted.

The on-chip oscillator is turned off in the 'Zero-Power' mode.

If the clock is provided by an external source which is not always running, then the 'Zero-Power' mode must be set when the clock is not available. Failure to observe this rule may cause a significant rise in the supply current drawn by MX614 as well as generating undefined states of the RXD, DET and RDY outputs.

4.2 Mode Control Logic

The MX614's operating mode is determined by the logic levels applied to the M0 and M1 input pins:

M1	M0	Rx Mode	Tx Mode	Data Retime[1]
0	0	1200bps	150bps	Rx
0	1	Off	1200bps	Tx
1	0	1200bps	Off / 5bps	Rx
1	1	'Zero-Power'		-

[1] If enabled

Note: On applying power to the device, the mode must be set to 'ZP', i.e. M0 = '1', M1 = '1', until V_{DD} has stabilized.

In the 'Zero-Power' (ZP) mode, power is removed from all internal circuitry. When leaving the 'ZP' mode there must be a delay of 20ms before any Tx data is passed to, or Rx data read from the device to allow the bias level, filters, and oscillator to stabilize.

4.3 Rx Input Amplifier

This amplifier is used to adjust the received signal to the correct amplitude for the FSK receiver and Energy Detect circuits (see section 5.1).

4.4 Receive Filter and Equalizer

The Receive Filter and Equalizer section is used to attenuate out of band noise and interfering signals, especially the locally generated transmit tones which might otherwise reach the 1200bps FSK Demodulator and Energy Detector circuits. This block also includes a switchable equalizer section. When the RXEQpin is low, the overall group delay of the receive filter is flat over the 1200bps frequency range. If the RXEQpin is high the receive filter's typical overall group delay will be as shown in Figure 3.

XR-2206

Monolithic Function Generator

EXAR ...the analog plus company™

June 1997-3

FEATURES

- Low-Sine Wave Distortion, 0.5%, Typical
- Excellent Temperature Stability, 20ppm/ 5C, Typ.
- Wide Sweep Range, 2000:1, Typical
- Low-Supply Sensitivity, 0.01%V, Typ.
- Linear Amplitude Modulation
- TTL Compatible FSK Controls
- Wide Supply Range, 10V to 26V
- Adjustable Duty Cycle, 1% TO 99%

APPLICATIONS

- Waveform Generation
- Sweep Generation
- AM/FM Generation
- V/F Conversion
- FSK Generation
- Phase-Locked Loops (VCO)

GENERAL DESCRIPTION

The XR-2206 is a monolithic function generator integrated circuit capable of producing high quality sine, square, triangle, ramp, and pulse waveforms of high-stability and accuracy. The output waveforms can be both amplitude and frequency modulated by an external voltage. Frequency of operation can be selected externally over a range of 0.01Hz to more than 1MHz.

The circuit is ideally suited for communications, instrumentation, and function generator applications requiring sinusoidal tone, AM, FM, or FSK generation. It has a typical drift specification of 20ppm/5C. The oscillator frequency can be linearly swept over a 2000:1 frequency range with an external control voltage, while maintaining low distortion.

ORDERING INFORMATION

Part No.	Package	Operating Temperature Range
XR-2206M	16 Lead 300 Mil CDIP	-555C to +1255C
XR-2206P	16 Lead 300 Mil PDIP	–405C to +855C
XR-2206CP	16 Lead 300 Mil PDIP	05C to +705C
XR-2206D	16 Lead 300 Mil JEDEC SOIC	05C to +705C

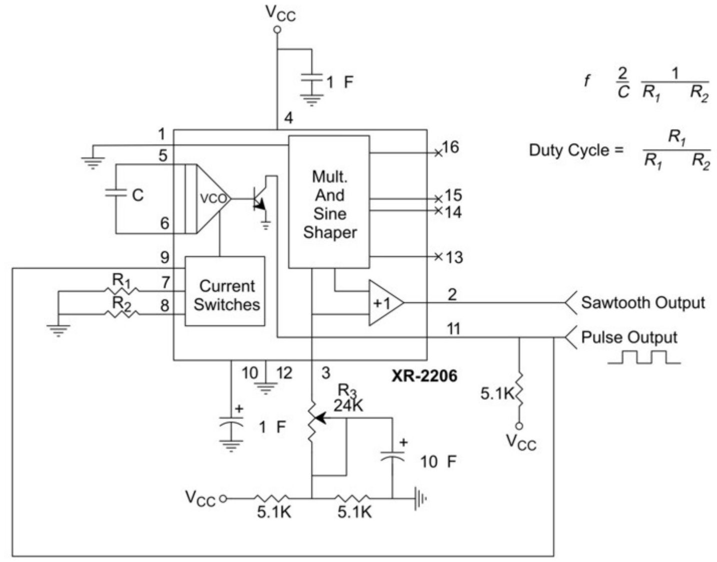


Figure 14. Circuit for Pulse and Ramp Generation.

Frequency-Shift Keying

The XR-2206 can be operated with two separate timing resistors, R_1 and R_2, connected to the timing Pin 7 and 8, respectively, as shown in *Figure 13.* Depending on the polarity of the logic signal at Pin 9, either one or the other of these timing resistors is activated. If Pin 9 is open-circuited or connected to a bias voltage 2V, only R_1 is activated. Similarly, if the voltage level at Pin 9 is 1V, only R_2 is activated. Thus, the output frequency can be keyed between two levels. f_1 and f_2, as:

$f_1 = 1/R_1C$ and $f_2 = 1/R_2C$

For split-supply operation, the keying voltage at Pin 9 is referenced to V^-.

Output DC Level Control

The dc level at the output (Pin 2) is approximately the same as the dc bias at Pin 3. In *Figure 11, Figure 12* and *Figure 13,* Pin 3 is biased midway between V+ and ground, to give an output dc level of $V^+/2$.

APPLICATIONS INFORMATION

Sine Wave Generation

Without External Adjustment

Figure 11 shows the circuit connection for generating a sinusoidal output from the XR-2206. The potentiometer, R_1 at Pin 7, provides the desired frequency tuning. The maximum output swing is greater than $V^+/2$, and the typical distortion (THD) is < 2.5%. If lower sine wave distortion is desired, additional adjustments can be provided as described in the following section.

The circuit of *Figure 11* can be converted to split-supply operation, simply by replacing all ground connections with V^-. For split-supply operation, R_3 can be directly connected to ground.

With External Adjustment:

The harmonic content of sinusoidal output can be reduced to -0.5% by additional adjustments as shown in *Figure 12.* The potentiometer, R_A, adjusts the sine-shaping resistor, and R_B provides the fine adjustment for the waveform symmetry. The adjustment procedure is as follows:

1. Set R_B at midpoint and adjust R_A for minimum distortion.
2. With R_A set as above, adjust R_B to further reduce distortion.

Triangle Wave Generation

The circuits of *Figure 11* and *Figure 12* can be converted to triangle wave generation, by simply open-circuiting Pin 13 and 14 (i.e., S_1 open). Amplitude of the triangle is approximately twice the sine wave output.

FSK Generation

Figure 13 shows the circuit connection for sinusoidal FSK signal operation. Mark and space frequencies can be independently adjusted by the choice of timing resistors, R_1 and R_2; the output is phase-continuous during transitions. The keying signal is applied to Pin 9. The circuit can be converted to split-supply operation by simply replacing ground with V^-.

Pulse and Ramp Generation

Figure 14 shows the circuit for pulse and ramp waveform generation. In this mode of operation, the FSK keying terminal (Pin 9) is shorted to the square-wave output (Pin 11), and the circuit automatically frequency-shift keys itself between two separate frequencies during the positive-going and negative-going output waveforms. The pulse width and duty cycle can be adjusted from 1% to 99% by the choice of R_1 and R_2. The values of R_1 and R_2 should be in the range of 1k   to 2M .

PRINCIPLES OF OPERATION

Description of Controls

Frequency of Operation:

The frequency of oscillation, f_o, is determined by the external timing capacitor, C, across Pin 5 and 6, and by the timing resistor, R, connected to either Pin 7 or 8. The frequency is given as:

$$f_0 \quad \frac{1}{RC}\ Hz$$

and can be adjusted by varying either R or C. The recommended values of R, for a given frequency range, as shown in *Figure 5.* Temperature stability is optimum for 4k   < R < 200k . Recommended values of C are from 1000pF to 100 F.

Frequency Sweep and Modulation:

Frequency of oscillation is proportional to the total timing current, I_T, drawn from Pin 7 or 8:

$$f \quad \frac{320 I_T\,(mA)}{C(\quad F)}\ Hz$$

Timing terminals (Pin 7 or 8) are low-impedance points, and are internally biased at +3V, with respect to Pin 12. Frequency varies linearly with IT, over a wide range of current values, from 1 A to 3mA. The frequency can be controlled by applying a control voltage, V_C, to the activated timing pin as shown in *Figure 10.* The frequency of oscillation is related to VC as:

$$f \quad \frac{1}{RC} \quad 1 \quad \frac{R}{R_C} \quad 1 - \frac{V_C}{3} \quad Hz$$

where V_C is in volts. The voltage-to-frequency conversion gain, K, is given as:

$$K \quad f \quad V_C \quad -\frac{0.32}{R_C C} \quad Hz \quad V$$

CAUTION: *For safety operation of the circuit, I_T should be limited to 3mA.*

XR-2211

FSK Demodulator/ Tone Decoder

EXAR ...the analog plus company™

June 1997-3

FEATURES

- Wide Frequency Range, 0.01Hz to 300kHz
- Wide Supply Voltage Range, 4.5V to 20V
- HCMOS/TTL/Logic Compatibility
- FSK Demodulation, with Carrier Detection
- Wide Dynamic Range, 10mV to 3V rms
- Adjustable Tracking Range, ±1% to 80%
- Excellent Temp. Stability, ±50ppm/5C, max.

APPLICATIONS

- Caller Identification Delivery
- FSK Demodulation
- Data Synchronization
- Tone Decoding
- FM Detection
- Carrier Detection

GENERAL DESCRIPTION

The XR-2211 is a monolithic phase-locked loop (PLL) system especially designed for data communications applications. It is particularly suited for FSK modem applications. It operates over a wide supply voltage range of 4.5 to 20V and a wide frequency range of 0.01Hz to 300kHz. It can accommodate analog signals between 10mV and 3V, and can interface with conventional DTL, TTL, and ECL logic families. The circuit consists of a basic PLL for tracking an input signal within the pass band, a quadrature phase detector which provides carrier detection, and an FSK voltage comparator which provides FSK demodulation. External components are used to independently set center frequency, bandwidth, and output delay. An internal voltage reference proportional to the power supply is provided at an output pin.

The XR-2211 is available in 14 pin packages specified for military and industrial temperature ranges.

ORDERING INFORMATION

Part No.	Package	Operating Temperature Range
XR-2211M	14 Pin CDIP (0.300º)	-555C to +1255C
XR-2211N	14 Pin CDIP (0.300º)	-405C to +855C
XR-2211P	14 Pin PDIP (0.300º)	-405C to +855C
XR-2211ID	14 Lead SOIC (Jedec, 0.150º)	-405C to +855C

EXAR Corporation, 48720 Kato Road, Fremont, CA 94538 (510) 668-7000 FAX (510) 668-7017

DESIGN EQUATIONS

(All resistance in , all frequency in Hz and all capacitance in farads, unless otherwise specified)

(See *Figure 3* for definition of components)

1. VCO Center Frequency, f_O:

$$f_O \quad \frac{1}{R_0 \cdot C_0}$$

2. Internal Reference Voltage, V_{REF} (measured at pin 10):

$$V_{REF} \quad \frac{V_{CC}}{2} \quad -650mV \ in \ volts$$

3. Loop Low-Pass Filter Time Constant, :

$$C_1 \cdot R_{PP} \ (seconds)$$

where:

$$R_{PP} \quad \frac{R_1 \cdot R_F}{R_1 \quad R_F}$$

if R_F is or C_F reactance is , then R_{PP} = R1

4. Loop Damping, :

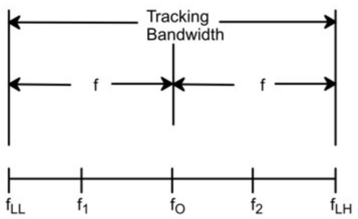
$$\overline{\quad\quad} \quad \frac{1250 \cdot C_0}{R_1 \cdot C_1}$$

Note: *For derivation/explanation of this equation, please see TAN-011.*

5. Loop-tracking

bandwidth, $\frac{f}{f_0}$

$$\frac{f}{f_0} \quad \frac{R_0}{R_1}$$



6. FSK Data filter time constant, tF:

$$_F \quad \frac{R_B \cdot R_F}{(R_B \quad R_F)} \cdot C_F \text{ (seconds)}$$

7. Loop phase detector conversion gain, Kd: (Kd is the differential DC voltage across pin 10 and pin11, per unit of phase error at phase detector input):

$$K_d \quad \frac{V_{REF} \cdot R_1}{10{,}000 \cdot} \frac{volt}{radian}$$

Note: *For derivation/explanation of this equation, please see TAN-011.*

8. VCO conversion gain, Ko: (Ko is the amount of change in VCO frequency, per unit of DC voltage change at pin 11):

$$K_0 \quad \frac{-2}{V_{REF} \cdot C_0 \cdot R_1} \qquad \frac{radian\ second}{volt}$$

9. The filter transfer function:

$$F(s) \quad \frac{1}{1 \quad SR_1 \cdot C_1} \text{ at 0 Hz.} \qquad S = J \text{ and } = 0$$

10. Total loop gain. K_T:

$$K_T \quad K_O \cdot K_d \cdot F(s) \qquad \frac{R_F}{5{,}000 \cdot C_0 \cdot (R_1 \quad R_F)} \frac{1}{seconds}$$

11. Peak detector current I_A:

$$I_A \quad \frac{V_{REF}}{20{,}000} \text{ (}V_{REF}\text{ in volts and }I_A\text{ in amps)}$$

Note: *For derivation/explanation of this equation, please see TAN-011.*

The code listing for the user interface in Borland Delphi Version 6 is:

```
unit Unit;

interface

uses
Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
Dialogs, Menus, StdCtrls, ComCtrls, ExtCtrls, ComPort, StrUtils, Math;

const
//Data files to store database fields
//Three databases - sensor information, field
//information and received sensor data
cdata='sensor.dat';
cdata2='infield.dat';
cdata3='farmdata.dat';
TEMP='temp.dat';


//Sensor information database details
type
sensordata=record
id:string[4];
farmno:integer;
sensor:string[15];
xcoordinate:integer;
ycoordinate:integer;
end;


//Received sensor data database details
infielddata=record
transmitter:string[4];
data:string[4];
date:string[8];
time:string[8];
end;
```

```
//Field information database details
farmdata=record
farmno:string[4];
farmwidth:integer;
farmlength:integer;
furrowsmon:integer;
end;

TForm1 = class(TForm)
GroupBox1:  TGroupBox;
GroupBox2:  TGroupBox;
MainMenu1:  TMainMenu;
File1:  TMenuItem;
Exit1:  TMenuItem;
Label1:  TLabel;
Label2:  TLabel;
Label3:  TLabel;
edtlength:  TEdit;
edtwidth:  TEdit;
edtfurrows:  TEdit;
Label4:  TLabel;
boxtype:  TComboBox;
Label5:  TLabel;
edtid:  TEdit;
Label6:  TLabel;
edtx:  TEdit;
edty:  TEdit;
Label7:  TLabel;
Label8:  TLabel;
btneditline:  TButton;
ListView1:  TListView;
btnsave:  TButton;
btnedit:  TButton;
btnsaveedit:  TButton;
btndelete:  TButton;
btnadd:  TButton;
btnreturn:  TButton;
```

```
btncancel: TButton;
btnchangedata: TButton;
btnok: TButton;
btnfarmcancel: TButton;
memstatus: TMemo;
Image1: TImage;
Label11: TLabel;
ComPort1: TComPort;
btnport: TButton;
Label10: TLabel;
cbports: TComboBox;
btneditOK: TButton;
ListView2: TListView;
ComboBox1: TComboBox;
Label9: TLabel;
ComboBox2: TComboBox;
ComboBox3: TComboBox;
Label12: TLabel;
Label13: TLabel;
Label14: TLabel;
procedure Exit1Click(Sender: TObject);
procedure btneditlineClick(Sender: TObject);
procedure FormCreate(Sender: TObject);
procedure btnsaveClick(Sender: TObject);
procedure btneditClick(Sender: TObject);
procedure btnsaveeditClick(Sender: TObject);
procedure btndeleteClick(Sender: TObject);
procedure btnaddClick(Sender: TObject);
procedure btnreturnClick(Sender: TObject);
procedure btncancelClick(Sender: TObject);
procedure btnchangedataClick(Sender: TObject);
procedure btnfarmcancelClick(Sender: TObject);
procedure btnokClick(Sender: TObject);
procedure FormClose(Sender: TObject; var Action: TCloseAction);
procedure ComPort1ReceiveCallBack(Data: String);
procedure btnportClick(Sender: TObject);
procedure ListView1ColumnClick(Sender: TObject; Column: TListColumn);
```

```
procedure ListView1Compare(Sender:  TObject; Item1, Item2:  TListItem;
Data:  Integer; var Compare:  Integer);
procedure btneditOKClick(Sender:  TObject);
procedure ComboBox2Change(Sender:  TObject);
procedure Image1MouseMove(Sender:  TObject; Shift:  TShiftState; X,
Y: Integer);
procedure Image1Click(Sender:  TObject);
private
( Private declarations )
procedure Deletec1(drop:  String);
procedure Deletec3(drop3:  String);
procedure Refresh(listview1:TListView);
procedure ShowImage(Image1:TImage);
public
( Public declarations )
end;

var
Form1:  TForm1;
// Define fields of sensor information database
c:sensordata;
ci:file of sensordata;
co:file of sensordata;
i:Integer;
ID:string[4];
Field:integer; // field were sensor is located
Name:string[15]; // type of sensor
Xcoord:integer; // x coordinate of sensor
Ycoord:integer; // y coordinate of sensor

// Define fields of received sensor data database
c2:infielddata;
ci2:file of infielddata;
co2:file of infielddata;
i2:Integer; // record locations
idv2:string[4]; // sensor identifier
datav2:string[4]; // data
datev2:string[8]; // date
```

```
timev2:string[8]; // time

// Define fields of farm information data database
c3:farmdata;
ci3:file of farmdata;
co3:file of farmdata;
i3:Integer; // record locations
farmid:string[4]; // farm number
farmwidth:integer; // farm width
farmlength:integer; // farm length
nofurrows:integer; // number of furrows

// Other variables
n,t,indicator,farms,farmcount,farmsel,counter:integer;
drop,drop2,drop3:string;
bmp:TBitmap;
cwidth,cheight:integer; // dimensions of canvas for paddock
wide,high,newwide,newhigh,furrows,fields:integer; // number of furrows

blnPortOpened:boolean;

ColumnToSort:  Integer;
LastSortedColumn:integer;
Ascending:boolean;

implementation

($R *.dfm)

procedure TForm1.Exit1Click(Sender:  TObject);
begin
Application.Terminate; // Exit graphic user interface
end;

procedure TForm1.btneditlineClick(Sender:  TObject);
begin
//When edit button of database is clicked, load new window
btneditline.Hide;
btnchangedata.Hide;
```

```
Image1.Hide;
memstatus.Hide;
Label13.Hide;
Label14.Hide;
GroupBox1.Hide;
GroupBox2.Hide;
listview1.Show;
listview2.Hide;
btnadd.Show;
btndelete.Show;
btnedit.Show;
btnreturn.Show;
indicator:=1;
Refresh(listview1);
end;

procedure TForm1.FormCreate(Sender:  TObject);
var
farms,i:integer;
begin
blnPortOpened:=False;
GroupBox1.Hide;
GroupBox2.Hide;
listview1.Hide;
listview2.Hide;
btnAdd.Hide;
btnEdit.Hide;
btnDelete.Hide;
btnReturn.Hide;
Button3.Show;
// Show serial ports available to transfer information base station
cbports.Show;
EnumPorts( cbPorts.Items );
if cbPorts.Items.Count > 0 then
cbPorts.ItemIndex := 0; // show first available port
```

```
Assignfile(ci3,cdata3);
($i-)
Reset(ci3);
($i+)
if ioresult <> 0 then rewrite(ci3);
farms:=filesize(ci3);
closefile(ci3);

Combobox2.Text:=inttostr(1);



for i:=1 to farms do
begin
ComboBox2.Items.Add(inttostr(i));
end;

farmsel:=0;
Label14.Show;
memstatus.Show;
ShowImage(Image1);
end;

procedure TForm1.btnsaveClick(Sender:  TObject);
begin
// If button pressed to save new database entry
btnAdd.Show;
btnEdit.Show;
btnDelete.Show;
btnReturn.Show;
GroupBox1.Hide;
GroupBox2.Hide;
Assignfile(ci,cdata);
($i-)
Reset(ci);
($i+)
if ioresult <> 0 then rewrite(ci);
i:=filesize(ci);
// Save entered values to new database entry
```

```
c.id:=edtid.Text;
c.farmno:=strtoint(Combobox3.Text);
c.sensor:=boxtype.Text;
c.xcoordinate:=strtoint(edtx.Text);
c.ycoordinate:=strtoint(edty.Text);
seek(ci,i);
write(ci,c);
closefile(ci);
edtid.Text:='';
boxtype.Refresh;
edtx.Text:='';
edty.Text:='';
// Refresh display of database entries
Refresh(listview1);
end;

procedure TForm1.btneditClick(Sender:  TObject);
begin
// If edit button of database is pressed, edit the entry of the database selected
btnadd.Hide;
btnedit.Hide;
btndelete.Hide;
btnreturn.Hide;
// For farm sensors
if indicator=1 then
begin
GroupBox1.Hide;
GroupBox2.Show;
btnsave.Hide;
btnsaveedit.Show;
// Obtain each database entry was selected
t:=listview1.Selected.Index;
// Display database entries in editboxes to edit
edtid.Text:=listview1.Items[t].Caption;
combobox3.Text:=listview1.Items[t].SubItems[0];
boxtype.Text:=listview1.Items[t].SubItems[1];
edtx.Text:=listview1.Items[t].SubItems[2];
```

```
edty.Text:=listview1.Items[t].SubItems[3];
end
// For farm data
else if indicator=2 then
begin
GroupBox1.Show;
GroupBox2.Hide;
btnOK.Hide;
btneditOK.Show;
// Obtain each database entry was selected
t:=listview2.Selected.Index;
// Display database entries in editboxes to edit
ComboBox1.Text:=listview2.Items[t].Caption;
edtlength.Text:=listview2.Items[t].SubItems[0];
edtwidth.Text:=listview2.Items[t].SubItems[1];
edtfurrows.Text:=listview2.Items[t].SubItems[2];
end;
end;

procedure TForm1.btnsaveeditClick(Sender:  TObject);
begin
// If entry in database has been edited and now saved
btnadd.Show;
btnedit.Show;
btndelete.Show;
btnReturn.Show;
GroupBox1.Hide;
GroupBox2.Hide;
Assignfile(ci,cdata);
($i-)
Reset(ci);
($i+)
if ioresult <> 0 then rewrite(ci);
// Save over database entry with new values
c.id:=edtid.Text;
c.farmno:=strtoint(combobox3.Text);
c.sensor:=boxtype.Text;
```

```
c.xcoordinate:=strtoint(edtx.Text);
c.ycoordinate:=strtoint(edty.Text);
seek(ci,t);
write(ci,c);
closefile(ci);
Refresh(listview1);
end;

procedure TForm1.btndeleteClick(Sender:  TObject);
begin
GroupBox1.Hide;
GroupBox2.Hide;
// For farm sensors
if indicator=1 then
begin
t:=listview1.Selected.Index;
drop:=listview1.Items[t].Caption;
Deletec1(drop);
Refresh(listview1);
end
// For farm data
else if indicator=2 then
begin
t:=listview2.Selected.Index;
drop3:=listview2.Items[t].Caption;
Deletec3(drop3);
Refresh(listview2);
end;
end;

// procedures I wrote

procedure Tform1.Deletec1(drop:  String);
var
FETCH:Integer;
found:boolean;
begin
Assignfile(ci,cdata);
```

```
Reset(ci);
($i-)
found:=false;
Reset(ci);
FETCH:=-1; // don't skip first file
//(it contains file information) - if we did, FETCH:=0;
while (not eof(ci)) and (not found) do
begin
FETCH:=FETCH+1;
Seek(ci,FETCH);
Read(ci,c);
if (drop=c.id) then
($i+)
begin
c.id:='De';
Seek(ci,FETCH);
Write(ci,c);
found:=true;
end;
end;
Closefile(ci);
Rename(ci,'@@.@@@');
Assignfile(ci,'@@.@@@');
Reset(ci);
Assignfile(co,TEMP);
Rewrite(co);
while not eof(ci) do
begin
($i-)
Read(ci,c);
($i+)
if (c.id<>'De') then Write(co,c);
end;
Closefile(ci);
Closefile(co);
Rename(co,cdata);
Erase(ci);
```

```
end;

procedure Tform1.Deletec3(drop3:  String);
var
FETCH:Integer;
found:boolean;
begin
// To delete a database entry
Assignfile(ci3,cdata3);
Reset(ci3);
($i-)
found:=false;
Reset(ci3);
FETCH:=-1; // don’t skip first file
//(it contains file information) - if we did, FETCH:=0;
while (not eof(ci3)) and (not found) do
begin
FETCH:=FETCH+1;
Seek(ci3,FETCH);
Read(ci3,c3);
if (drop3=c3.farmno) then
($i+)
begin
c3.farmno:=‘De’;
Seek(ci3,FETCH);
Write(ci3,c3);
found:=true;
end;
end;
Closefile(ci3);
Rename(ci3,‘@@.@@@’);
Assignfile(ci3,‘@@.@@@’);
Reset(ci3);
Assignfile(co3,TEMP);
Rewrite(co3);
while not eof(ci3) do
begin
```

```
($i-)
Read(ci3,c3);
($i+)
if (c3.farmno<>‘De’) then Write(co3,c3);
end;
Closefile(ci3);
Closefile(co3);
Rename(co3,cdata3);
Erase(ci3);
end;

procedure TForm1.btnaddClick(Sender:  TObject);
begin
// To add database entry
btnadd.Hide;
btnedit.Hide;
btndelete.Hide;
btnreturn.Hide;
// For farm sensors
if indicator = 1 then
begin
btnsaveedit.Hide;
btnsave.Show;
GroupBox1.Hide;
GroupBox2.Show;
edtid.Text:=‘’;
boxtype.ItemIndex:=-1;
edtx.Text:=‘’;
edty.Text:=‘’;
end
// For farm data
else if indicator = 2 then
begin
btneditOK.Hide;
btnOK.Show;
GroupBox1.Show;
GroupBox2.Hide;
```

```
ComboBox1.Text:=‘’;
edtlength.Text:=‘’;
edtwidth.Text:=‘’;
edtfurrows.Text:=‘’;
end

else if indicator = 3 then
begin
btneditOK.Hide;
btnOK.Show;
GroupBox1.Show;
GroupBox2.Hide;
ComboBox1.Text:=‘’;
edtlength.Text:=‘’;
edtwidth.Text:=‘’;
edtfurrows.Text:=‘’;
end;
end;

procedure TForm1.btnreturnClick(Sender:  TObject);
begin
// To return from window to edit or add
// database entry to graphical display
Label13.Hide;
btneditline.Show;
btnchangedata.Show;
indicator:=0;
Image1.Show;
GroupBox1.Hide;
GroupBox2.Hide;
btnadd.Hide;
btndelete.Hide;
btnedit.Hide;
btnreturn.Hide;
listview1.Hide;
listview2.Hide;
indicator:=1;
```

```
//Obtain database entries to draw new
//graphical display of farm information
Refresh(listview1);
indicator:=2;
Refresh(listview2);

Assignfile(ci3,cdata3);
($i-)
Reset(ci3);
($i+)
if ioresult <> 0 then rewrite(ci3);
farms:=filesize(ci3);
closefile(ci3);

Combobox2.Clear;

for i:=1 to farms do
begin
ComboBox2.Items.Add(inttostr(i));
end;
Combobox2.Text:=inttostr(1);

// Update graphic display
Label14.Show;
memstatus.Show;
ShowImage(Image1);
end;

procedure TForm1.btncancelClick(Sender:  TObject);
begin
btnadd.Show;
btnedit.Show;
btndelete.Show;
btnReturn.Show;
GroupBox1.Hide;
GroupBox2.Hide;
end;
```

```
procedure TForm1.btnchangedataClick(Sender:  TObject);
begin
// Change database addressed
btneditline.Hide;
btnchangedata.Hide;
Image1.Hide;
Label13.Hide;
Label14.Hide;
memstatus.Hide;
GroupBox1.Hide;
GroupBox2.Hide;
listview1.Hide;
listview2.Show;
btnadd.Show;
btndelete.Show;
btnedit.Show;
btnreturn.Show;
indicator:=2;
Refresh(listview2);
end;

procedure TForm1.btnfarmcancelClick(Sender:  TObject);
begin
//If change to database entry cancelled
btnadd.Show;
btnedit.Show;
btndelete.Show;
btnReturn.Show;
GroupBox1.Hide;
GroupBox2.Hide;
end;

procedure TForm1.btnokClick(Sender:  TObject);
var
i:  integer;
begin
// If change in database entry confirmed update database entries
btnAdd.Show;
```

```
btnEdit.Show;
btnDelete.Show;
btnReturn.Show;
GroupBox1.Hide;
GroupBox2.Hide;


Assignfile(ci3,cdata3);
// ($i-)
Reset(ci3);
// ($i+)
if ioresult <> 0 then rewrite(ci3);
i:=filesize(ci3);
c3.farmno:=Combobox1.Text;
c3.farmlength:=strtoint(edtlength.Text);
c3.farmwidth:=strtoint(edtwidth.Text);
c3.furrowsmon:=strtoint(edtfurrows.Text);
seek(ci3,i);
write(ci3,c3);
closefile(ci3);
Combobox1.Text:='';
edtlength.Text:='';
edtwidth.Text:='';
edtfurrows.Text:='';
indicator:=2;
Refresh(listview2);
end;

procedure TForm1.FormClose(Sender: TObject; var Action: TCloseAction);
begin
bmp.Free;
ComPort1.Close; // on form close
ComPort1.Free; // on form destroy
end;

procedure TForm1.ComPort1ReceiveCallBack(Data: String);
var
t,counter:integer;
```

```
rstring,rid,rdatau,rdatal,rday,rmonth,ryear,rhr,rmin,rsec:string;
begin
Label13.Hide;
if counter=1 then sleep(3000);
sleep(5000); // pause for 5 seconds
memstatus.Lines.Add(Data); // display incoming serial data
rstring:=Data;

// ---------------------------------------------------------------------------
// Identify sensor identification, data, date and time parts of message
// ---------------------------------------------------------------------------
t:=Pos('-',rstring); // find location of delimiter, '-'
Delete(rstring,1,t); // delete start of message bytes

for counter := 1 to 8 do
begin
t:=Pos('-',rstring); // find location of delimiter, '-'

if counter = 1 then
rid := copy(rstring,0,t-1) // separate id
else if counter = 2 then
begin
rdatau := copy(rstring,0,t-2); // separate datau (decimal)
rdatal := copy(rstring,t-1,t-2); // separate datal (decimal)
end
else if counter = 3 then
rday := copy(rstring,0,t-1) // separate day
else if counter = 4 then
rmonth := copy(rstring,0,t-1) // separate month
else if counter = 5 then
ryear := copy(rstring,0,t-1) // separate year
else if counter = 6 then
rhr := copy(rstring,0,t-1) // separate hour
else if counter = 7 then
rmin := copy(rstring,0,t-1) // separate minute
else if counter = 8 then
rsec := copy(rstring,0,t-1); // separate second
```

```
Delete(rstring,1,t); // delete string section just separated
end;

t:=Pos('-',rstring); // find location of delimiter, '-'
Delete(rstring,t,t+2); // delete end of message bytes
// ---------------------------------------------------------------------------

// ---------------------------------------------------------------------------
// Interpret data
// ---------------------------------------------------------------------------
// Decide where the sensors will be positioned

// ---------------------------------------------------------------------------

// Put into comma separated values (csv) file format
Assignfile(ci2,cdata2);
($i-)
Reset(ci2);
($i+)
if ioresult <> 0 then rewrite(ci2);
i2 := filesize(ci2);
c2.transmitter := rid;
c2.data := rdatau + '.'  + rdatal;
c2.date := rday + '-' + rmonth + '-' + ryear;
c2.time := rhr + ':'  + rmin + ':'  + rsec;

seek(ci2,i2);
write(ci2,c2);
closefile(ci2);

//If number of memo lines exceeds 500, do not show these lines
if memstatus.Lines.Count > 500 then
memstatus.Clear;
Label14.Show;
memstatus.Show;
ShowImage(Image1);
end;
```

```
procedure TForm1.btnportClick(Sender:  TObject);
begin
cbPorts.Enabled := true;
ComPort1.Port := cbPorts.Items[cbPorts.ItemIndex];

// Close port
if blnPortOpened=True then
begin
blnPortOpened:=False;
ComPort1.Close;
end
else
// Open port
begin
counter:=1;
blnPortOpened:=True;
ComPort1.Open
end;
end;

procedure Tform1.Refresh(listview1:TListView);
begin
// Refreshes the display of the database
// For farm sensors
if indicator=1 then
begin
listview1.Items.Clear;
Assignfile(ci,cdata);
($i-)
Reset(ci);
($i+)
while not eof(ci) do
begin
read(ci,c);
listview1.Items.Add;
n:=listview1.Items.Count-1;
// Insert database entries into database displayed in user interface
listview1.Items[n].Caption:=c.id;
```

```
listview1.Items[n].SubItems.Add(inttostr(c.farmno));
listview1.Items[n].SubItems.Add(c.sensor);
listview1.Items[n].SubItems.Add(inttostr(c.xcoordinate));
listview1.Items[n].SubItems.Add(inttostr(c.ycoordinate));
end;
Closefile(ci);
end
// For farm data
else if indicator=2 then
begin
listview2.Items.Clear;
Assignfile(ci3,cdata3);
($i-)
Reset(ci3);
($i+)
while not eof(ci3) do
begin
read(ci3,c3);
listview2.Items.Add;
n:=listview2.Items.Count-1;
// Insert database entries into database displayed in user interface
listview2.Items[n].Caption:=c3.farmno;
listview2.Items[n].SubItems.Add(inttostr(c3.farmlength));
listview2.Items[n].SubItems.Add(inttostr(c3.farmwidth));
listview2.Items[n].SubItems.Add(inttostr(c3.furrowsmon));
end;
Closefile(ci3);
end
else if indicator=3 then
begin
listview2.Items.Clear;
Assignfile(ci2,cdata2);
$i-
Reset(ci2);
$i+
while not eof(ci2) do
begin
```

```
read(ci2,c2);
listview2.Items.Add;
n:=listview2.Items.Count-1;
// Insert database entries into database displayed in user interface
listview2.Items[n].Caption:=c2.transmitter;
listview2.Items[n].SubItems.Add(c2.data);
listview2.Items[n].SubItems.Add(c2.date);
listview2.Items[n].SubItems.Add(c2.time);
end;
Closefile(ci2);
end;
end;

procedure TForm1.btneditOKClick(Sender:  TObject);
var
t:integer;
begin
// If database entry was edited
btnadd.Show;
btnedit.Show;
btndelete.Show;
btnReturn.Show;
GroupBox1.Hide;
GroupBox2.Hide;
Assignfile(ci3,cdata3);
($i-)
Reset(ci3);
($i+)
if ioresult <> 0 then rewrite(ci3);
// Write editted values entered in window to database
c3.farmno:=Combobox1.Text;
c3.farmlength:=strtoint(edtlength.Text);
c3.farmwidth:=strtoint(edtwidth.Text);
c3.furrowsmon:=strtoint(edtfurrows.Text);
t:=listview2.Selected.Index;
seek(ci3,t);
write(ci3,c3);
```

```
closefile(ci3);
indicator:=2;
// Refresh database display
Refresh(listview2);
end;

procedure Tform1.ShowImage(Image1:TImage);
var
f,tt,s,x,y,ypos,noitems,state,progress,length:integer;
info,sense,fheight:string;
begin
// Update the graphical display of sensors and farm information
Image1.Canvas.Pen.Color:=clMaroon;//$000080FF;
Image1.Canvas.Brush.Color:=clMaroon;//$000080FF;
cwidth:=550;
cheight:=650;
Image1.Canvas.Rectangle(0,0,cwidth,700);

// Get farm coordinates for each field
indicator:=2;
Refresh(listview2);
high:=strtoint(listview2.Items[farmsel].SubItems[0]);
wide:=strtoint(listview2.Items[farmsel].SubItems[1]);
furrows:=strtoint(listview2.Items[farmsel].SubItems[2]);

// Find number of field from data
Assignfile(ci3,cdata3);
{$i-}
Reset(ci3);
{$i+}
if ioresult <> 0 then rewrite(ci3);
n:=filesize(ci3);
closefile(ci3);

// Draw field to fit in canvas in correct ratios
Image1.Canvas.Brush.Color:=$000080FF;
Image1.Canvas.Brush.Color:=clGreen;
newhigh:=cheight;
```

```
newwide:=cwidth;
Image1.Canvas.Rectangle(0,0,newwide,newhigh);

for tt:=1 to furrows do
begin
// Draw furrows on paddock as black lines
Image1.Canvas.Brush.Color:=clBlack;
Image1.Canvas.Pen.Color:=clBlack;
Image1.Canvas.Rectangle(round((tt-0.05)*newwide/...
(furrows+1)),0,round((tt+0.05)*newwide/(furrows+1)),newhigh);

// Label furrows on paddock
Image1.Canvas.Brush.Color:=$000080FF;
Image1.Canvas.Pen.Color:=clBlack;
Image1.Canvas.TextOut(round((tt-0.05)*newwide/(furrows+1)),newhigh+5,inttostr(tt));
end;

// Determine location (furrow number) of sensor
indicator:=1;
Refresh(listview1);
f:=1; // initialise furrow counter
Image1.Canvas.Brush.Color:=$000080FF;
Image1.Canvas.Pen.Color:=clBlack;
Image1.Canvas.TextOut(round((f-0.05)*newwide/(furrows+1)),newhigh+5,inttostr(f));
for t := 0 to 3 do //listview1.Items.Count-1 do
begin
if listview1.Items[t].SubItems[0]=inttostr(farmsel+1) then
begin
x:=strtoint(listview1.Items[t].SubItems[2]); // x-coordinate of sensor
y:=strtoint(listview1.Items[t].SubItems[3]); // y-coordinate of sensor

// Display sensor information on field
Image1.Canvas.Brush.Color:=clLtGray;
Image1.Canvas.Pen.Color:=clBlack;

if t>1 then
begin
// If two sensors have the same x-coordinate
```

```
// sensors are located in the same furrow
if strtoint(listview1.Items[t].SubItems[2])<>...
strtoint(listview1.Items[t-1].SubItems[2]) then
begin
if strtoint(listview1.Items[t].SubItems[0])=...
strtoint(listview1.Items[t-1].SubItems[0]) then
begin
f:=f+1;
Image1.Canvas.Brush.Color:=clLtGray;
Image1.Canvas.Pen.Color:=clBlack;
Image1.Canvas.TextOut(round((f-0.05)*newwide/(furrows+1)),newhigh+5,inttostr(f));
end;
end;
end;

indicator:=2;
Refresh(listview2);
length:=strtoint(listview2.Items[farmsel].SubItems[0]);

// Read in sensor data from .dat file to display on canvas
indicator:=3;
Refresh(listview2);
state:=strtoint(listview2.Items[t].SubItems[0]);

// Update furrow water progress
indicator:=1;
Refresh(listview1);
if listview1.Items[t].SubItems[1]='Advance meter' then
begin
// Show advance front progress down field
Image1.Canvas.Brush.Color:=clBlue;
Image1.Canvas.Pen.Color:=clBlue;
progress:=round(state*cheight*strtoint(listview1.Items[t].SubItems[3])/length);
Image1.Canvas.Rectangle(round((f-0.05)*newwide/(furrows+1)),0,
round((f+0.05)*newwide/(furrows+1)),round(progress));
// Assign colour for drawing sensor
// yellow for advance, red for inflow/outflow, green for other
Image1.Canvas.Brush.Color:=clYellow;
```

```
end
else if (listview1.Items[t].SubItems[0]='Inflow meter') or
(listview1.Items[t].SubItems[0]='Outflow meter') then
Image1.Canvas.Brush.Color:=clRed
else
Image1.Canvas.Brush.Color:=clGreen;
Image1.Canvas.Brush.Color:=clYellow;

// Draw sensors on paddock
for s:=0 to listview1.Items.Count-1 do
begin
Image1.Canvas.Pen.Color:=clBlack;
y:=strtoint(listview1.Items[s].SubItems[3]); // y-coordinate of sensor
Image1.Canvas.Ellipse(round((f-0.05)*newwide/(furrows+1)),
round((y/high*cheight-0.1*newwide/(furrows+1))),
round((f+0.05)*newwide/(furrows+1)),
round(y/high*cheight)); // Draw circle with black outline
end;
end;
for s:=0 to listview1.Items.Count-1 do
begin
Image1.Canvas.Pen.Color:=clBlack;
if listview1.Items[s].SubItems[1]='Advance meter' then // for advance meter
Image1.Canvas.Brush.Color:=clYellow
else if (listview1.Items[s].SubItems[1]='Inflow meter')
or (listview1.Items[s].SubItems[0]='Inflow meter') then // for inflow or outflow
meter
Image1.Canvas.Brush.Color:=clRed
else
Image1.Canvas.Brush.Color:=clGreen;

y:=strtoint(listview1.Items[s].SubItems[3]); // y-coordinate of sensor
Image1.Canvas.Ellipse(round((f-0.05)*newwide/(furrows+1)),
round(0.95*(y/high*cheight-0.1*newwide/(furrows+1))),
round((f+0.05)*newwide/(furrows+1)),
round(0.95*y/high*cheight)); // Draw circle with black outline // Draw yellow circle
with black outline
Image1.Canvas.Ellipse(round((f-0.05)*newwide/(...
```

```
furrows+1)),round(0.95*(y/wide*cwidth-0.1*newwide/(furrows+1))
),...
round((f+0.05)*newwide/(furrows+1)),round(0.95*y/wide*cwidth))
end;
Image1.Show;
end;

procedure TForm1.Image1MouseMove(Sender:  TObject; Shift:  TShiftState; X,
Y: Integer);
var
nn,xmouse,ymouse,xpos,ypos,xdisp,ydisp,ff:integer;
sense,info:string;
begin
Label13.Show;
Label14.Show;
memstatus.Show;
ShowImage(Image1);

// Find location of mouse
xmouse:=mouse.CursorPos.X;
ymouse:=mouse.CursorPos.Y;

indicator:=1;
Refresh(listview1);
ff:=1; // initialise furrow counter

for nn:=0 to listview1.Items.Count-1 do // For each of the sensors
begin
indicator:=1;
Refresh(listview1);
xpos:=strtoint(listview1.Items[nn].SubItems[2]); // x-coordinate of sensor
ypos:=strtoint(listview1.Items[nn].SubItems[3]); // y-coordinate of sensor

// Identify sensor that mouse is passing over
if nn>1 then
begin
// If two sensors have the same x-coordinate
// sensors are located in the same furrow
```

```
if xpos<>strtoint(listview1.Items[nn-1].SubItems[2]) then
begin
if strtoint(listview1.Items[nn].SubItems[0])=...
strtoint(listview1.Items[nn-1].SubItems[0]) then
begin
ff:=ff+1;
end;
end;
end;

indicator:=2;
Refresh(listview2);
high:=strtoint(listview2.Items[farmsel].SubItems[0]); // Length of field
wide:=strtoint(listview2.Items[farmsel].SubItems[1]); // Width of field
furrows:=strtoint(listview2.Items[farmsel].SubItems[2]);// No of furrows
xpos:=round(ff*newwide/(furrows+1)+270); // Transformed x-coordinate of sensor
ypos:=round(ypos*cheight/high+80); // Transformed y-coordinate of sensor

// Define pen and brush colours ready to write sensor text on canvas
Image1.Canvas.Brush.Color:=clWhite;
Image1.Canvas.Pen.Color:=clBlack;

// If the mouse is dragged over a point near a sensor then display info
if (xmouse>0.95*xpos) and (xmouse<1.05*xpos) and
(ymouse>0.95*ypos) and (ymouse<1.05*ypos) then
begin
// Display sensor information on field
xdisp:=xpos-300;
ydisp:=ypos-105;

// Read in sensor data from .dat file to display on canvas
indicator:=3;
Refresh(listview2);
sense:=listview2.Items[nn].Caption;
info:=listview2.Items[nn].SubItems[0];
Image1.Canvas.TextOut(xdisp,ydisp,sense+',''+info);
end;
end;
```

```
end;

procedure TForm1.ComboBox2Change(Sender:  TObject);
begin
farmsel:=Combobox2.ItemIndex;
Label14.Show;
Label13.Hide;
memstatus.Show;
ShowImage(Image1);
end;

procedure TForm1.Image1Click(Sender:  TObject);
var
xmouse,ymouse,ff,nn,xpos,ypos,xdisp,ydisp:integer;
sense:string;
begin
Label13.Show;
Label13.Color:=clYellow;
Label14.Show;
memstatus.Show;
ShowImage(Image1);
// Find location of mouse
xmouse:=mouse.CursorPos.X;
ymouse:=mouse.CursorPos.Y;

indicator:=1;
Refresh(listview1);
ff:=1; // initialise furrow counter

for nn:=0 to listview1.Items.Count-1 do // For each of the sensors
begin
indicator:=1;
Refresh(listview1);
xpos:=strtoint(listview1.Items[nn].SubItems[2]); // x-coordinate of sensor
ypos:=strtoint(listview1.Items[nn].SubItems[3]); // y-coordinate of sensor
xdisp:=xpos;
ydisp:=ypos;
```

```
// Identify sensor that mouse is passing over
if nn>1 then
begin
// If two sensors have the same x-coordinate
// sensors are located in the same furrow
if xpos<>strtoint(listview1.Items[nn-1].SubItems[2]) then
begin
if strtoint(listview1.Items[nn].SubItems[0])=...
strtoint(listview1.Items[nn-1].SubItems[0]) then
begin
ff:=ff+1;
end;
end;
end;

indicator:=2;
Refresh(listview2);
high:=strtoint(listview2.Items[farmsel].SubItems[0]); // Length of field
wide:=strtoint(listview2.Items[farmsel].SubItems[1]); // Width of field
furrows:=strtoint(listview2.Items[farmsel].SubItems[2]);// No of furrows
xpos:=round(ff*newwide/(furrows+1)+270); // Transformed x-coordinate of sensor
ypos:=round(ypos*cheight/high+80); // Transformed y-coordinate of sensor

// Define pen and brush colours ready to write sensor text on canvas
Image1.Canvas.Brush.Color:=clLtGray;
Image1.Canvas.Pen.Color:=clBlack;

// If the mouse is dragged over a point near a sensor then display info
if (xmouse>0.95*xpos) and (xmouse<1.05*xpos) and
(ymouse>0.95*ypos) and (ymouse<1.05*ypos) then
begin
// Read in sensor data from .dat file to display on canvas
indicator:=3;
Refresh(listview2);
sense:=listview2.Items[nn].Caption;
xdisp:=xpos+10;
ydisp:=ypos-50;
Label13.Caption:='Data incoming...';
```

```
Label13.Left:=xdisp;
Label13.Top:=ydisp;
// ComPort1.send();
end;
end;
end;

end.
```