

9

MAC and Physical Layer Energy Efficiency for Ad Hoc Wireless Sensor Networks

Zoran Luka Josip Basich



This thesis is submitted in partial fulfillment of the
academic requirements for the degree of
Master of Science in Electrical Engineering
in the Faculty of Engineering and The Built Environment
University of Cape Town
August 2006

The copyright of this thesis vests in the author. No quotation from it or information derived from it is to be published without full acknowledgement of the source. The thesis is to be used for private study or non-commercial research purposes only.

Published by the University of Cape Town (UCT) in terms of the non-exclusive license granted to UCT by the author.

SUBMISSION APPROVAL

**As the candidate's supervisor I have approved this
dissertation for submission**

Signature
Name



H Anthony Chan

Dated in Cape Town, this *8th* day of September, 2006.

Declaration

The research work presented in this dissertation was performed by Mr Zoran Luka Josip Basich, under the supervision of Prof. H Anthony Chan, at the Centre of Excellence at the University of Cape Town. This work was financially supported by THRIP, Telkom S.A., and Siemens as part of the Centres of Excellence programme.

The whole dissertation, unless specifically indicated to the contrary in the text, is Zoran Luka Josip Basich's own work and it has not been plagiarised from any source or submitted in part or in whole for any other university degree purpose. Where collaboration with other people has taken place, or material generated by other researchers is included, the parties and/or materials are cited in the acknowledgements or are explicitly stated with references as appropriate.

I am presenting this thesis for examination for the Degree of Master of Science in Electrical Engineering. I also grant the University free license to reproduce the above thesis in whole, or in part, for research purposes.

Signature

Zoran Basich
.....

Dated in Cape Town, this *8* day of September, 2006.

MAC and Physical Layer Energy Efficiency for Ad Hoc Wireless Sensor Networks

By Zoran Luka Josip Basich

Abstract

The research work undertaken involves the design of a new, energy efficient Medium Access Control (MAC) layer for Ad Hoc Wireless Sensor Networks (AHWSN). MAC solutions are either contention based or non-contention based. Energy inefficiencies in contention based MAC protocols suffered from collisions, overhearing, control overhead and idle listening. Non-contention based MAC protocols introduced TDMA / CDMA / FDMA that did not suffer from those problematic issues. However, they suffered from other problems, such as energy inefficient hierarchies. The hierarchy uses cluster-heads to co-ordinate neighbours which is a continual requires that is energy inefficient. The proposal named Colour TDMA MAC is introduced, which does not have a hierarchy or cluster-head problems. It uses a single channel and simple transmitters. It also uses a distributed algorithm from colouring graph mathematics to ensure that the hidden terminal and exposed terminal problems of wireless data communication do not occur.

Colour TDMA MAC also introduces the two concepts that allow nodes to sleep longer. The two are:

- The Timed PicoRadio
- The Mailbox

A node may sleep when it is not using the channel. Yet it may use one of the above concepts to receive a message destined for it. In the case of the timed PicoRadio, the node is awakened if it is sleeping. In the case of the mailbox, the radio signal is stored in memory and when the microprocessor awakens, it can deal with the message.

A comparison of the central idea (Colour TDMA MAC) to a mainstream contention based MAC protocol (S-MAC) for AHWSN reveals that S-MAC suffers from collisions and idle listening (to a great extent) which is energy wasted. Its other energy inefficiencies are overhearing and control overhead. The two scheduling algorithms are compared via timing diagrams to see which delivers a message successfully in the shortest time. They are also placed head to head in some random tests to evaluate which is more energy efficient.

Research work shows that the Colour TDMA MAC greatly improves energy savings. On the downside, it trades off energy for channel usage, thus messages take longer to reach their destination. Results also show that the Colour TDMA MAC is exceptionally good for unicast messaging where both the sender and the destination are known.

Acknowledgements

First and foremost, I wish to express my special thanks to Professor H Anthony Chan for his great help, guidance, and patience over the last two years. Thank you also for assisting me in preparing a number of conference papers.

A vote of big thanks to Telkom S.A for financing my master's degree studies as well as the sponsors of the Centre of Excellence University of Cape Town namely THRIP, Telkom S.A, and Siemens. Also to the University of Cape Town for their financial support which enabled me to attend two international conferences, as well as two conferences held in South Africa where I presented results of my work and where I was able to discuss and exchange views with researchers from other countries on the subject matter.

I would like to express my deepest gratitude to my family for their encouragement and support, (meals, drinks, transport, clean clothes, house parties, prayers, love plus everything else) and for all that they have sacrificed for me, and my Dad for helping to edit this thesis.

Thank you Lord God Almighty for being the biggest inspiration of my life and for granting me the opportunity to be where I am. Also for loving me and giving me enough intellect to accomplish everything I have done to date.

I also wish to record my debt of gratitude to Andrew Markham, my fellow MSc and BSc friend, for listening to my ideas and bringing me back down to earth, as well as for his comments on my dissertation and editing help.

Finally, I am grateful to my friends and postgraduate CoE colleagues for making the past two years a very satisfying and memorable.

Table of Contents

Abstract	iv
Acknowledgements	v
Table of Contents	vi
List of Abbreviations	x
List of Figures	xi
List of Tables	xv

1 INTRODUCTION	1
1.1 Research Objectives	1
1.2 Ad Hoc Wireless Sensor Networks	1
1.3 AHWSN Node's Energy Utilisation	2
1.4 Intrinsic AHWSN Node Properties	3
1.5 Assumptions	3
1.6 Thesis Outline	4
2 THE MAC LAYER IN AHWSN	6
2.1 Contention and Non-Contention Protocols	6
2.2 Flat versus Hierarchical Arrangement	6
2.3 Contention Based MAC Protocols	8
2.3.1 Sensor-MAC	10
2.3.2 Timeout-MAC	12
2.3.3 Berkeley-MAC	12
2.4 Non-Contention Based MAC Protocols	13
2.4.1 Bit-Map-Assisted MAC and LEACH	14
2.4.2 GANGS	14
2.4.3 Energy and Rate-based MAC	14
2.4.4 Node Adaptive Multiple Access and Traffic Adaptive Medium Access	14
2.4.5 E-MAC	15

2.5 Contention versus Non-Contention	15
3 COLOUR TDMA MAC	18
3.1 The MAC Protocol Proposal – Colour TDMA MAC	18
3.1.1 Aims of Colour TDMA MAC	19
3.1.2 Higher Level Abstraction of the Scheme	19
3.1.3 Colour TDMA MAC Example	21
3.2 The Colouring Algorithm	23
3.2.1 One-Hop and Two-Hop Neighbours	24
3.2.2 Unidirectional Radio Links	25
3.2.3 The Two-Hop Colouring Algorithm and Determination of α & β	25
3.2.4 Sacrificing Channel Usage for Collision Avoidance	29
3.2.5 Assumptions in Initialisation	30
3.2.6 The Colouring Algorithm during Initialisation	31
3.2.7 Avoiding the Ripple Effect	33
3.2.8 The Colouring Algorithm after Initialisation	35
3.2.9 The Hidden and Exposed Terminal Issue	35
3.3 The Scheduling Algorithm	35
3.3.1 The Awake/Asleep Schedule	35
3.3.2 The Scheme	35
3.3.3 Broadcast and Unicast	37
3.3.4 Special Events and the Final Schedule	37
3.3.5 The Neighbour Information Table Example	40
3.3.6 Special Events and Colouring After Initialisation	41
3.3.7 The Choice of Node _i	45
3.3.8 Nodes Exiting the Colour TDMA MAC Schedule	45
3.3.9 Network's TDMA Cycle	45
3.4 Synchronisation and Neighbour Discovery	46
3.4.1 Synchronisation	46
3.4.2 Number of Neighbours is More Than the Maximum and Can Not Be Reduced	46

3.5 Initialisation	48
3.5.1 Initialisation Overview of Colour TDMA MAC	48
3.5.2 The Synchronisation Process	49
3.5.3 Discover Neighbours and Reduce Radio Connectivity where Necessary	51
3.5.4 The Timeslot Allocation Process	52
3.6 Scheme Operation	54
3.7 The PicoRadio	54
3.8 Advantages and Disadvantages	55
4 THE PICORADIO AND MAILBOX	56
4.1 Physical Layer Considerations	56
4.1.1 Wireless Data Transmission	56
4.1.2 Transmitter, Preamble, Modulation, and Encoding	57
4.2 The PicoRadio	58
4.2.1 A General Design of a PicoRadio	58
4.2.2 A MEMS PicoRadio	59
4.2.3 A Feasible PicoRadio for AHWSN	59
4.3 The Mailbox	60
5 RESULTS	63
5.1 Comparison of Colour TDMA MAC to S-MAC	63
5.2 Implementation of Microcontroller and MAC Properties to Obtain Results	64
5.3 S-MAC Mathematical Contention Analysis	66
5.3.1 The S-MAC Schedule	66
5.3.2 Contenders	77
5.3.3 Probability of Collisions	68
5.4 Colour TDMA MAC Properties and the Schedule	73
5.5 S-MAC versus Colour TDMA MAC	74
5.5.1 One Example of S-MAC versus Colour TDMA	74
5.5.2 Simulations	76
5.5.3 Plotting the Schemes versus Each Other in Equality	77

5.5.4 Boundary Values – Two Nodes and Two Messages	79
5.5.5 Boundary Values – Two Nodes and Ten Messages	88
5.5.6 Boundary Values – Five Nodes and Two Messages	90
5.5.7 Boundary Values – Five Nodes and Ten Messages	91
5.5.8 Points in Between and Points Outside the Boundaries	93
5.5.9 Plotting the Energy Usage	93
5.5.10 Time to Pass a Message from One Hop to the Next	94
5.5.11 S-MAC Initialisation versus Colour TDMA MAC Initialisation	95
5.6 Guideline Results for Non-Contention Based MAC	96
6 CONCLUSIONS	97
6.1 Colour TDMA MAC versus Other MAC Protocols	97
6.1.1 Colour TDMA MAC versus S-MAC	97
6.1.2 Colour TDMA MAC versus other TDMA MAC	98
6.1.3 Colour TDMA MAC Uniqueness in AHWSN	98
6.1.4 Extra Hardware and Costs in Colour TDMA	99
7 FUTURE WORK	100
7.1 The Mailbox	100
7.2 Time Sharing	100
7.3 The MAC Layer and Routing	101
8 BIBLIOGRAPHY AND REFERENCES	102
9 APPENDICES	104
9.1 Probability and the Birthday Algorithm	104
9.1.1 The Birthday Algorithm	104
9.1.2 Collisions in Contention and the Birthday Algorithm	105
9.1.3 Combinations of the Birthday Algorithm	106
9.1.4 Code for Combinations	110

List of Abbreviations

In this dissertation the following abbreviations have been used :

AHWSN:	Ad Hoc Wireless Sensor Network(s).
MAC:	Medium Access Control.
CSMA/CA:	Carrier Sense Multiple Access with Collision Avoidance.
TDMA:	Time Division Multiple Access.
FDMA:	Frequency Division Multiple Access.
CDMA:	Code Division Multiple Access.
SYNC:	Synchronisation.
RTS:	Request To Send.
CTS:	Clear To Send.
ACK:	Acknowledgement.
S-MAC:	Sensor-MAC.
T-MAC:	Timeout-MAC.
B-MAC:	Berkeley MAC.
ER-MAC:	Energy and Rate-based MAC.
NAMA:	Node Activation Multiple Access.
TRAMA:	Traffic Adaptive Medium Access.
TRP:	Timeslot Request Packet.
NIT:	Neighbour Information Table.
SAW:	Surface Acoustic Wave.
ADC:	Analogue to Digital Converter.
CMOS:	Complimentary Metal Oxide Semiconductor.
FET:	Field Effect Transistor.
S/N:	Signal to Noise ratio
2D:	Two-dimensional
3D:	Three-dimensional

List of Figures

Figure 1.1 Sensor and Data Energy Requirements.....	2
Figure 1.2 The Awake / Asleep Duty Cycle of a Node.....	3
Figure 2.1 Node Arrangement and Division	7
Figure 2.2 The Scheduling Algorithm of S-MAC.....	10
Figure 2.3 Probability of At Least One Collision in the SYNC Contention Window versus the Number of Contenders	11
Figure 2.4 Probability of At Least One Collision in the DATA Contention Window versus the Number of Contenders	11
Figure 3.1 Example of an AHWSN with Three Randomly Chosen Nodes Showing Their Radio Range.....	20
Figure 3.2 Colouring Algorithm Providing Unique Numbers over Two Nodes Radio Connectivity.....	20
Figure 3.3 Scheduling of the Awake and Sleep Times.....	21
Figure 3.4 Channel User Number 1's Timeslot Subdivision.....	21
Figure 3.5 Colour TDMA MAC Example with Channel User 1 Nodes	22
Figure 3.6 Colour TDMA MAC Example with Channel User 2 Nodes	22
Figure 3.7 Colour TDMA MAC Example with Channel User 3 Nodes	23
Figure 3.8 Node A and C both use Timeslot 4 and there is a Collision at Node B	24
Figure 3.9 Node E is in Radio Range of Node D, But Node D is Not in Radio Range of Node E.....	25
Figure 3.10 Round One: Node A Sends Out a Message that it has Chosen Timeslot 1	27
Figure 3.11 Round One: Nodes A, B, C and D Neighbourhood Information Tables .	27
Figure 3.12 Round Two: Nodes B and D Sends Out a Message requesting a Timeslot and Confirming Node A has Timeslot 1.....	27

Figure 3.13 Round Two: Nodes A, B, C and D Neighbourhood Information Tables.	28
Figure 3.14 Round Three: Node A Announces Who Won the Timeslot	28
Figure 3.15 Round Three: Nodes A, B, C and D Neighbourhood Information Tables	28
Figure 3.16 Round Four: Node D Sends out a Request for a Timeslot.....	28
Figure 3.17 Round Four: Nodes A, B, C and D Neighbourhood Information Tables	29
Figure 3.18 Values of β when α is between 1 and 10	30
Figure 3.19 One AHWSN with Different Node Densities over Different Areas	30
Figure 3.20 Another Example of the Colouring Algorithm with $\alpha = 3$	32
Figure 3.21 Flow Diagram of Choices Node B Computes if it is Requested to Change Timeslots during Scheme Operation	34
Figure 3.22 The Scheduling Algorithm With Instantaneous Wake Up and Data Transmission.....	36
Figure 3.23 The Scheduling Algorithm Accommodating Radio Wake Up and Non-Instantaneous Data Transmission	36
Figure 3.24 The Scheduling Algorithm in Detail	37
Figure 3.25 Example Colour TDMA MAC AHWSN with a Maximum Limit of Four Neighbours.....	40
Figure 3.26 Example Colour TDMA MAC AHWSN with a Maximum of Four Neighbours and Their Unique Identification Numbers	40
Figure 3.27 The Special Events Section of the Colour TDMA MAC Scheme	42
Figure 3.28 Node _i 's Message to Tell All its Neighbours it is Still There	43
Figure 3.29 Node _i 's Helping Section to Any Newcomers	43
Figure 3.30 Node _i Not being in Radio Range of All of the Newcomer's Neighbours	44
Figure 3.31 Node 'A' is a Gateway Node and if it were to Leave the Network it Would Split it in Two.	47
Figure 3.32 Initialisation Overview of Colour TDMA MAC	48
Figure 3.33 Synchronisation, TDMA Schedule and Neighbour Discovery being Propagated Outwards.....	49

Figure 3.34 Later Stages of Synchronisation, TDMA Schedule and Neighbour Discovery being Propagated Outwards	50
Figure 3.35 Synchronisation, TDMA Schedule and Neighbour Discovery being Propagated Outwards as well as the Colouring Algorithm taking place	50
Figure 3.36 Number of Neighbours Can Not be Reduced	51
Figure 3.37 Nodes have Timeslots over One-Hop in β Duty Cycles	53
Figure 3.38 Nodes have Timeslots over Two-Hops in 2β Duty Cycles	53
Figure 3.39 Nodes have Timeslots over Three-hops in 3β Duty Cycles	53
Figure 4.1 Simplified Network Stack Model.....	56
Figure 4.2 Lower Section of the Simplified Networking Stack Model.....	57
Figure 4.3 Physical Layer Transmission Stack	57
Figure 4.4 Diagram of a PicoRadio and Microcontroller With Integrated Timer	58
Figure 4.5 Mailbox Position in the System	60
Figure 4.6 Options for Dealing with the Incoming Signal	61
Figure 5.1 S-MAC Schedule	66
Figure 5.2 Illustrations of Contenders	67
Figure 5.3 Probability of At Least One Collision in the RTS and CTS Section of S-MAC Given a Set Number of Contenders.....	68
Figure 5.4 Number of Successful Transmissions for 3 Contenders Given There Will Be At Least One Collision.....	69
Figure 5.5 Number of Successful Transmissions for 4 Contenders Given There Will Be At Least One Collision.....	69
Figure 5.6 Number of Successful Transmissions for 5 Contenders Given There Will Be At Least One Collision.....	70
Figure 5.7 Number of Successful Transmissions for 6 Contenders Given There Will Be At Least One Collision.....	70
Figure 5.8 Number of Successful Transmissions for 7 Contenders Given There Will Be At Least One Collision.....	71
Figure 5.9 Number of Successful Transmissions for 8 to 15 Contenders Given There Will Be At Least One Collision.....	71

Figure 5.10 Markov Chain to Find the Number of Retries until Success	72
Figure 5.11 One Timeslot of Colour TDMA MAC.....	73
Figure 5.12 The Special Events Timeslot of Colour TDMA MAC	74
Figure 5.13 S-MAC Three Nodes Energy Utilisation when	75
Node 1 is Sending Data to Node 2	75
Figure 5.14 Colour TDMA MAC Three Nodes Energy Utilisation when Node 1 is Sending Data to Node 2.....	76
Figure 5.15 Basic 3D Graph to Compare S-MAC and Colour TDMA MAC with no Variable for the Y-Axis	77
Figure 5.16 Number of Different Senders and Receivers and Their Energy Consumption.....	78
Figure 5.17 S-MAC Two Nodes, One Node Sending Two Messages to its Neighbour	80
Figure 5.18 Colour TDMA Two Nodes, One Node Sending Two Messages to its Neighbour	80
Figure 5.19 S-MAC Results	94
Figure 5.20 Colour TDMA MAC Results.....	94
Figure 5.21 Time to Pass a Message Hop to Hop in S-MAC with Collisions versus Colour TDMA MAC, Both with Constant Traffic.....	95
Figure 9.1 The Birthday Algorithm Tree.....	106
Figure 9.2 The Interface	111

List of Tables

CHAPTER 2	
Table I Node Arrangement and MAC Protocols	8
Table II Evolution of Contention Based MAC Protocols for AHWSN	8
Table III Comparing Factors of General CSMA to General TDMA	16
CHAPTER 3	
Table IV Channel User 6's Neighbourhood Information (ID: HH)	41
CHAPTER 5	
Table V The Nordic nRF9E5	64
Table VI S-MAC Properties	65
Table VII Number of Nodes in Contention and Their Associated Number of Additional Duty Cycles Until Message is Successful	72
Table VIII Configuration of Two Nodes and Two Messages To Be Sent	79
Table IX S-MAC Energy Usage For Two Nodes Where One Node Sends Two Messages	81
Table X S-MAC Energy Usage For Two Nodes Where Each Node Sends One Message	82
Table XI Colour TDMA MAC Energy Usage For Two Nodes Where One Node Sends Two Messages	83
Table XII Colour TDMA MAC Energy Usage For Two Nodes Where Each Node Sends One Message	84
Table XIII Comparison of S-MAC vs Colour TDMA MAC Energy Usage Two Nodes Where Two Messages need to be Sent	86
Table XIV Two Nodes Sending Between 1 and 5 Packets Between Each Other	88
Table XV Five Nodes, and Ten Messages to be Sent	92

1 INTRODUCTION

This introduction states the research objectives and it provides an explanation of what an Ad Hoc Wireless Sensor Network (AHWSN) is. It elaborates on how the energy consumption of a node is split into two categories. It also enumerates the properties of a node in the network, and finally, it lists the assumptions made during this research and it discusses the outline for this paper.

1.1 Research Objectives

This thesis proposes a new Time Division Multiple Access (TDMA) Medium Access Control (MAC) protocol. It is designed to increase energy savings for each node in an AHWSN thereby increasing the life-span of the network.

1.2 Ad Hoc Wireless Sensor Networks

What is an AHWSN?

It is an infrastructure-less network created to sense the environment and to cooperatively pass the sensor readings back to a central point.

An AHWSN comprises a number of self-contained electronic devices (nodes) communicating wirelessly with the particular purpose of sensing a parameter. Such parameters could be machine vibrations or temperature changes or many other measurable properties of the environment. The acquired data is then fed to an information sink.

Ad hoc means formed for a specific purpose without pre-planning (such as the installation of infrastructure). The devices themselves (nodes) are the resources of the network, to each other.

Information transmission is the most energy expensive function of the node (as shown in Table V).

What are AHWSNs used for?

There are thousands of different applications of which a few are listed here:

- **Military** – Tracking an enemy tank by deploying these devices in a field. [18]
- **Industry** – Using the devices in a factory environment to take readings on the state, performance and operation of the machinery.
- **Research** – Placing these devices in birds' nests on an isolated island. [13]

The primary goals of AHWSN in order of importance are:

- 1 **Energy efficiency** – to prolong the life of the network. One means of energy conservation is for nodes to exercise an awake / asleep duty cycle
- 2 **Self-organising** – if individual failure occurs the network still carries on or it can repair itself
- 3 **Scalability** – the protocol must support between two and ten thousand nodes
- 4 **Cost reduction** – by using less expensive nodes a great amount of money is saved when the network is large, such as a system with one thousand nodes.

1.3 AHWSN Node's Energy Utilisation

The nodes of the network operate off a small non-renewable fixed source of energy. They are small devices typically the size of a coin [1], often placed in hard to reach places, and their power sources are small and limited. Since the network is required to last over a long period, saving energy is of highest importance.

To achieve such goal, the node's power requirements must be examined. The node's power usage is split between its two functions (as illustrated by Figure 1.1)

- 1 Sensing the environment.
- 2 Routing acquired data (either from the sensor or from a neighbouring node) to where it is required via a multi-hop path.

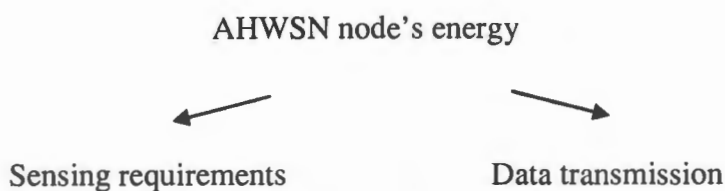


Figure 1.1 Sensor and Data Energy Requirements

This thesis will concentrate only on the telecommunication needs of the AHWSN node, which is the data transmission.

1.4 Intrinsic AHWSN Node Properties

AHWSNs focus on saving energy. An efficient method for conserving energy is to employ an awake / asleep duty cycle as illustrated by Figure 1.2. The node goes in to a powered-down status for a relatively long period and then awakens and fulfils all its requirements. Protocols that use this awake / asleep duty cycle perform with better energy savings [1], [5], [9]

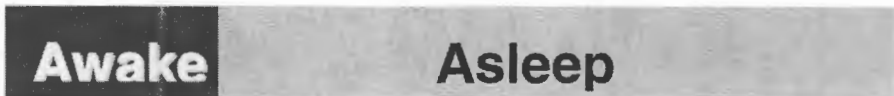


Figure 1.2 The Awake / Asleep Duty Cycle of a Node

The duty cycle illustrated in Figure 1.2 takes place repetitively over time. This greatly reduces the average power consumption, because the microcontroller and the radio transmitter, which are the greatest energy users, need only be on for a short time. Microcontrollers can use one to four milliamps or more while in operation and radio transmitters approximately ten to thirty milliamps. A chip will be introduced in Table V in Chapter 5 with specifications.

The disadvantage is that passing a message in a hop-to-hop manner (neighbour to neighbour) can only happen during the 'awake' period. Between each hop, there is a delay of one 'asleep' cycle. Hence these nodes cannot perform real-time operations without a delay.

Each node has a unique identification number. This is to allow nodes to be able to distinguish their neighbours one from another.

1.5 Assumptions

The nodes are stationary but new stationary nodes may be added to the network at some point in the future.

AHWSN is the name of a specialised type of network so 'Ad Hoc' does not imply its traditional telecommunications definition of mobile nodes. When examining applications of AHWSN (see subsection 1.2.), it can be seen that non-mobile nodes are deployed for a specific purpose but that at any time, more stationary nodes may be added to the network. If we take the example of military application in Section 1.2 then a plane may fly over head and drop nodes on to a field that monitor enemy tank movement. Some time later the military might want to track soldiers in the field so the aeroplane flies over the field again and drops more nodes on to the field. The new and old nodes need to talk to each other.

Messages between nodes consist of two types:

- 1 Data – Sensor readings
- 2 Control – Nodes communicating information about the current scheme.

Messages are sent between neighbours in the form of packets.

Data and Control readings are transmitted in fixed packet sizes.

All nodes are aware of the fixed length of a packet. If the data of one message does not fit in to one packet, then it is divided into sequential packets of the fixed length. A data message is thirty-two bytes and a control message is ten bytes.

During transmission of a signal the node is not aware if any collisions take place.

At the sender's side, if a radio is sending and it could receive at the same time, then the sending signal will drown out any receiving signal. At the receiver's side, the near-far problem [15] could occur. So the only way for a sending node to know whether its signal has collided is by not receiving an acknowledgement (ACK) from the message's destination. Sometimes, the sending signal is lost on the way to the receiving node and occasionally, the ACK from the receiving node is lost on the way back to the original data sender. In both cases the sender will resend the message.

Nodes are aware of the offset of their clock to their neighbours, after synchronisation, with the allowance of an error in milliseconds.

There is one channel both for signalling and for data transmissions

One channel covers all the nodes' wireless requirements. It is not subdivided, nor does it have any additional channels. The reasons for that are discussed in Subsection 2.3.

Only internal collisions are investigated and not external collisions.

Internal collisions are when nodes, whose radio ranges overlap, use the medium at the same time. External collisions occur when another network uses the same frequency as the network under investigation.

1.6 Thesis outline

This thesis investigates a novel MAC layer design for AHWSN. It begins with an introduction in Chapter 1 as to what AHWSN are, then moves onto their properties and this in turn is followed by the assumptions applied to the undertaken work.

The literature review follows, in which contention based MAC protocols are investigated first and one particular popular scheme is discussed. It then moves onto non-contention based MAC schemes on which the proposal of this thesis is based. It

highlights the problem issues in current MAC schedules and it indicates where the proposal fits in.

The central idea is then presented in Chapter 3, beginning with a high level view of how it works and the algorithm of graph colouring. This is the how the network ensures that channel usage is never overlapping in a two-hop neighbour range. It then goes onto the scheduling algorithm depicted over time. The refined details of scheme operation, implementation, and initialisation follow. Both the good and bad points of the proposed MAC protocol are highlighted and they are summarised in the advantages and disadvantages list at the end of the chapter.

Chapter 4 discusses the physical layer only briefly because the focus of this paper is on the MAC layer. The PicoRadio device of the physical layer is discussed and a hardware diagram presented because it can complement the MAC layer in saving energy. An improvement to the PicoRadio would be a mailbox. This is where an incoming signal is stored in memory while a node is asleep and deciphered when the microcontroller awakens. Such a design idea is presented for future work.

In Chapter 5, results are discussed. The choice and reasons for comparing S-MAC to Colour TDMA MAC are presented followed by the proof of the claim in the literature review that S-MAC is energy inefficient in collisions and control overhead (RTS and CTS). Then the two schemes are investigated for similarities and those that can be compared directly are plotted onto the same graph.

Chapter 6 summarises the findings from Chapter 5, while future work is briefly discussed in Chapter 7. Chapter 8 lists all references. Finally background information, mathematical algorithms and computer programmes are presented as appendices in Chapter 9.

2 THE MAC LAYER IN AHWSN

This section deals with the literature review of what work has been done thus far and the problems that have not been solved (or possibly created). It starts with an introduction to contention and non-contention MAC protocols. It then discusses the arrangement of nodes, which is the background information to how the AHWSN are viewed from a researcher's perspective. This is followed by a literature review of contention MAC protocols and the non-contention based MAC protocols. Finally, the summary compares the two approaches.

2.1 Contention and Non-Contention Protocols

Nodes in contention based MAC protocols sense the channel for activity and if no activity is detected they place a 'Request to Send' (RTS) to use the channel. They receive a 'Clear to Send' (CTS) from the node they want to communicate with. The first successful pair of nodes to RTS and CTS, secures the right to use the medium. Then data transmission may occur between the two nodes and all other neighbouring nodes are required to 'keep quiet.' At the end of the data transmission the receiving node sends an 'Acknowledgement' (ACK) that it has obtained the data. Some contention protocols do not use RTS / CTS / ACK, they just send their data when it is available [19].

Non-contention based MAC protocols use any of the following (or a combination) to assign guaranteed access to the channel for transmission:

- Time
- Frequency
- Code
- Phase
- Spatial Diversity

2.2 Flat versus Hierarchical Arrangement

How the arrangement of nodes in the network is viewed is linked to which MAC approach is used in an AHWSN. The word topology refers to the arrangement of nodes, but topology has an established usage in routing. The central idea is not routing. Therefore the word topology is avoided throughout to avoid confusion.

As an infrastructure-less network the nodes can either:

- 1 all be equal – referred to as *flat*, or
- 2 certain nodes have authority over others – referred to as *hierarchical*.

Generally, the nature of the MAC protocol determines how the arrangement is viewed. Contention based protocols use a flat arrangement because they are of an “equal chance for all” type. A hierarchy is used in non-contention based MAC protocols due to the need to co-ordinate nodes or create manageable clusters.

However, this is not always the case, as a hierarchical approach can arise either from the details of the MAC protocol or from the different properties of the node, such as one node having a longer range radio than its surrounding neighbours.

The advantages of each are outlined below:

Flat	Hierarchical
<p>It is the most scalable arrangement. If any node fails in the network, it is not detrimental to the network. If any node fails in the network, it is not detrimental to the network, whereas in a hierarchical arrangement, should the co-ordinating node fail, then that part of the network is inaccessible. The nodes are then in mayhem until a new administrator is appointed. The change of co-ordinator whether it is random or dynamic is energy consuming.</p>	<p>It is the best network organiser. Co-ordination takes place by assigned nodes and they arrange the whole network into something more manageable. Breaking a large network down into smaller entities is easier to control. A flat arrangement offers no such organisation.</p>

The hierarchical approach is also called “Cluster-based.”

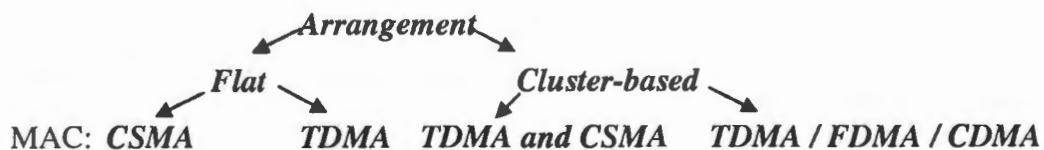


Figure 2.1 Node Arrangement and Division

The above Figure 2.1 illustrates different approaches to viewing the arrangement of the nodes in a network.

TABLE I

NODE ARRANGEMENT AND MAC PROTOCOLS

Arrangement:	MAC Protocols that use that arrangement:
Flat CSMA	CSMA/CA, PAMAS, MACAW, IEEE 802.11 DCF, S-MAC, T-MAC.
Flat TDMA	Slotted ALOHA (DARPA Packet Radio Network) [20]
Hierarchy or Cluster-based TDMA	ER-MAC, Bit-Map-Assisted MAC, LEACH, GANGS, EMAC

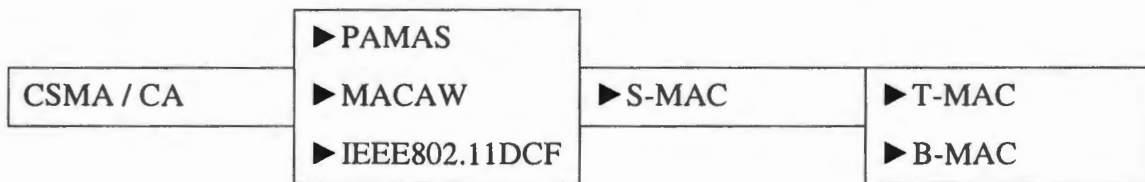
This thesis proposal fits into the Flat TDMA field in Table I. It is described and discussed in the next chapter called “Colour TDMA MAC.”

2.3 Contention Based MAC Protocols

TABLE II

EVOLUTION OF CONTENTION BASED MAC PROTOCOLS FOR AHWSN

Father of contention MAC Protocols	First generation “spin-offs” to improve energy efficiency	Biggest success MAC and most utilised in AHWSN	“Improvements” to S-MAC
------------------------------------	---	--	-------------------------



Each generation step (▶) is an improvement in energy efficiency in the protocol

CSMA / CA is the father of all Contention based MAC protocols. S-MAC [1], T-MAC [2], B-MAC [3], PAMAS [5], MACAW [1], IEEE 802.11 DCF [1].

Contention based protocols suffer from four adverse power consumption issues [6],[8]:

- 1 Collisions – Nodes send packets at anytime during a contention slot and the signals collide in the air.
- 2 Overhearing – Nodes hear messages that are not intended for them.
- 3 Control overhead – Nodes send RTS and CTS packets to secure the channel.
- 4 Idle listening – Even if there is no information being sent or intended for a node, the nodes are still required to listen to the channel.

Contention protocols are unable to eliminate all four of these power consuming issues. At best, some are able to reduce two of them to almost nothing. PAMAS and S-MAC reduce overhearing by allowing nodes to go to sleep during data transmission and they also reduce idle listening by incorporating a duty cycle.

The first generation “spin-offs” (Table II) were improvements to CSMA / CA, in that they introduced a sleep mode and hence a duty cycle. Since then, most protocols have enforced a duty cycle, and those that do not use a duty cycle use a wake up radio [16] or a solar panel and they are awake when there is energy [13].

The “spin offs” used multiple channels to avoid both collisions and overhearing, which is not suitable for AHWSN. This is because channel hopping is highly energy inefficient [11] and the transmitters used are always more costly than single channel transmitters [10]. S-MAC and PAMAS are very similar, the main difference being that S-MAC uses only one channel [9]. Further improvements to S-MAC were T-MAC and B-MAC. The T-MAC improvement was an adaptive duty cycle and a shortening of the active window of S-MAC. This means that the nodes could sleep longer, but with a shorter awake period it is harder to adapt to changes in the network and listen to the traffic. (The adaptive listening protocol of S-MAC requires that the nodes listen to the traffic of other nodes [1]). B-MAC which is based on S-MAC, introduced a set of interfaces that tweaked the contention algorithm further.

A problem with contention MAC protocols is that if they do not employ a fair method of ensuring all nodes obtain access to the channel, then resource starvation occurs.

Contention based protocols are good for new nodes joining the network, as they can just enter the scheme thus making it a very scalable solution.

2.3.1 Sensor-MAC [1]

Sensor-MAC (S-MAC) uses the contention scheme in Figure 2.2

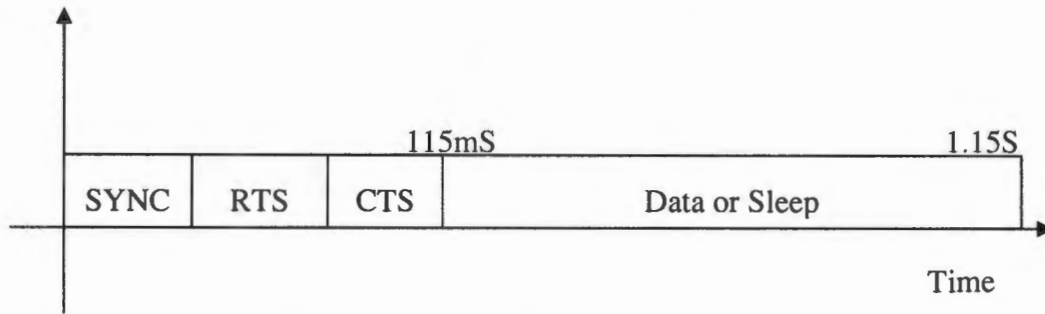


Figure 2.2 The Scheduling Algorithm of S-MAC

The SYNC period is for the synchronisation of messages. It is used by nodes to check that their schedule starts at the same time as the other nodes in the scheme. It is also used by nodes to let their neighbours know that they are still participating in the network.

The RTS and CTS section have contention windows for contenders. The nodes send their request to use the channel in the RTS slot. They hopefully receive an acknowledgement (CTS) that they can send the data during the 'data or sleep' stage. If no CTS is received or a node does not receive a RTS intended for itself, then it goes to sleep during the 'data and sleep' period.

Problem Issues:

S-MAC suffers from the four power consuming contention issues. The duration of the awake period is static (the same) for all nodes. This is unfair for nodes with less energy. An increase in energy efficiency may be achieved by making weaker nodes sleep more. (This is one of the improvements which T-MAC makes over S-MAC).

Collisions are a major loss of energy, as indicated below:

The following S-MAC properties are taken from [1] as it was simulated in NS2.

Control packet length:	10 bytes.
Data packet length:	up to 250 bytes.
MAC header length:	8 bytes.
Duration of listen interval:	115mS.
Frame length / duty cycle:	1150mS.
Contention window for SYNC:	15 slots.
Contention window for data:	31 slots.
Max number of neighbours:	14.

By using the birthday algorithm (Appendix 9.1.2), the following graphs (Figures 2.3 and 2.4) of the probability of collision for one duty cycle of a node with its neighbouring nodes are obtained.

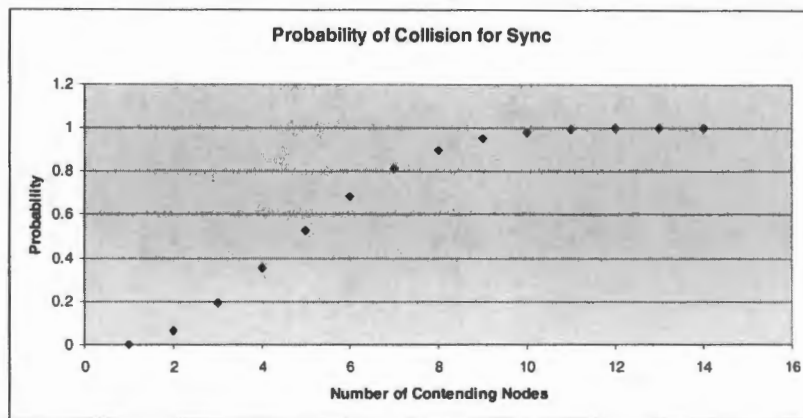


Figure 2.3 Probability of At Least One Collision in the SYNC Contention Window versus the Number of Contenders

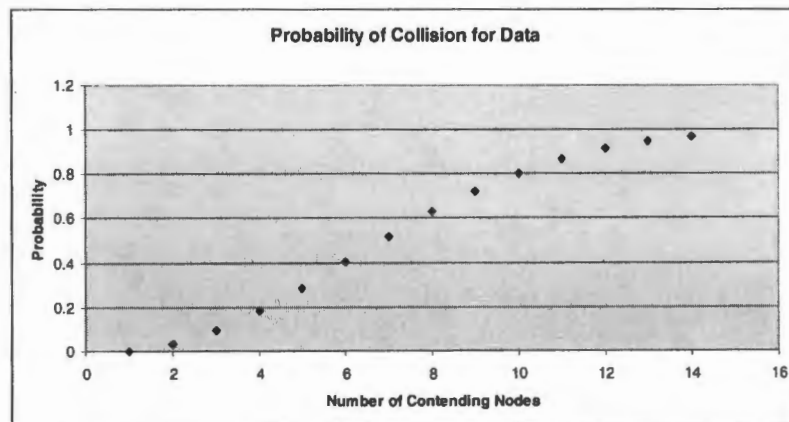


Figure 2.4 Probability of At Least One Collision in the DATA Contention Window versus the Number of Contenders

From above Figure 2.4, it follows that there is a 40% chance of a collision occurring between six contenders in any one duty cycle. However, there are 75 130 duty cycles in any one day, thus there is a significant loss of energy in one day.

The success of S-MAC:

After S-MAC’s development, it became freely available for everyone to use. It is a complete and stable MAC solution, which is also easy to understand and implement. It is constantly being updated and is “plug ‘n play”, thus researchers working on routing algorithms in AHWSN can add S-MAC to their microcontroller. Its code is a part of the TinyOS systems software. All these reasons sum up why it is the most used

MAC in AHWSN implementation, even though it is not the most energy efficient solution. S-MAC's popularity is shown by use in [13], [21], [22] and more. A Google Scholar search for "S-MAC Sensor-MAC" provided over 800 citations.¹

2.3.2 Timeout-MAC [2]

Timeout-MAC (T-MAC) is based on S-MAC and it is an improvement to the protocol.

T-MAC's improvement to S-MAC is that the nodes are allowed to "timeout" after a non-fixed interval. Timeout means that after a certain time, if the node has not heard from its neighbours, it may go to sleep. Thus, nodes with no data to send or nodes with less energy can sleep longer. Furthermore, if a node's neighbour promised to send data and has not done so for a long time, then a timeout occurs instead of the node waiting for the whole sleep period.

This concept was introduced by [2] and then simulated in Matlab. Searching on the Internet for the code resulted in zero hits.² Those who work with AHWSN, and who want a "plug 'n play" T-MAC, are not able to find it easily and hence do not use or implement it.

Problem Issues:

T-MAC suffers from same contention issues which are collisions, control overhead, idle listening. T-MAC predecessor S-MAC was available as a complete and stable solution that developers can use, but T-MAC is not.

2.3.3 Berkeley-MAC [3]

Berkeley-MAC (B-MAC) is based on S-MAC and it is an improvement to the protocol through interfaces that plug-in to the contention protocol. B-MAC is implemented in the TinyOS software but it is still in the beta stages. B-MAC is only available in the non-stable version of TinyOS, which developers are less likely to use. Users want to be sure that the code has been well tested if they are going to use it.

Problem Issues:

B-MAC has the same contention problems as S-MAC, with no collision avoidance. It also suffers from control overhead such as RTS, CTS exists within the network. Again, unlike its predecessor S-MAC it is not available as a stable solution.

¹ Search done on the 29th August 2006

² 29th August 2006

2.4 Non-Contention Based MAC Protocols

The following strategies remove the collisions, control overhead, idle listening and overhearing issues of contention based MAC protocols.

- TDMA arranges the channel into discrete time intervals and only one node may use each timeslot.
- CDMA provides nodes with nearly orthogonal, non-interfere communication channels.
- FDMA divides the channel up in to different frequency channels and each node is assigned its own channel.

If a node has a certain timeslot (or frequency or channel), then the protocol is less scalable. This is because nodes entering the scheme need to go through the administration of obtaining a timeslot (frequency or channel). They cannot use any available one.

TDMA schemes do not suffer from the issue of overhearing because while a sender transmits data to the receiver, the other nodes sleep or power down their radios. Idle listening is reduced to a minimum because nodes have specific times (frequencies or channels) when their neighbour will inform them if they want to send a message. Thus, TDMA / CDMA / FDMA do not suffer from the issues of contention, but they do introduce new problems. CDMA and FDMA are very energy intensive [11] hence they will not be considered for the rest of this thesis.

The TDMA issues are:

- 1 The TDMA scheme requires administration.
- 2 Low bandwidth utilisation occurs if timeslots are assigned but not used.
- 3 Protocols that use a hierarchy suffer from clusters interfering with each other; i.e. the radio ranges of nodes in a cluster overlap into other clusters.

The second problem mentioned above depends on the network. If there are a large number of neighbours then a large number of timeslots exist in the TDMA scheme. If only a few nodes send messages, then channel utilisation would be low. If the same nodes were to use a contention based MAC, there is the probability of collisions, which would lead to resubmissions and channel utilisation would be dynamic. Therefore, over time, in a contention based scheme, there would be low bandwidth usage due to a large number of collisions, and at other times there would be high channel utilisation with very few nodes actually sending data. Thus, contention based solutions are not free of the second problem issue either.

2.4.1. Bit-Map-Assisted MAC and LEACH

The hierarchy approach divides the network up into smaller groups of nodes with one node having authority over the others. This network communication can be split in two types:

- 1 Inter-cluster – cluster-heads or masters talking to each other.
- 2 Intra-cluster – communication within the one cluster.

Intra-cluster communication deals with a small group of nodes and how they communicate with their master. Inter-cluster communication deals with how the master nodes communicate with each other. Some papers propose an intra-cluster communication MAC protocol but further reading indicates that this is all they offer and that they are not an overall MAC solution.

Problem Issues:

Bit-Map-Assisted MAC and LEACH are not complete MAC solutions. They use the role of cluster-heads, which is energy inefficient.

2.4.2 GANGS [8]

This MAC protocol uses a combination of contention and non-contention. Cluster-heads communicate with each other (inter-cluster) via a TDMA schedule, while intra-cluster nodes communicate via a contention based scheme.

Problem Issues:

Even though a cluster can only correspond with its cluster-head, the nodes still interfere with other clusters because of radio ranges overlapping. This scheme suffers from both the issues of contention and non-contention based MAC protocols. The role of cluster-heads is energy taxing for a node to administrate.

2.4.3 Energy and Rate-based MAC [6]

In ER-MAC, each node is assigned two transmission timeslots. A node may only sleep during its allotted timeslot and it has no data to send. This means the node keeps its radio in receive mode throughout other neighbour's timeslots. This is idle listening and from an energy consumption perspective, it is inefficient.

Problem Issues:

A node's radio must stay awake during other node's timeslots.

2.4.4 Node Activation Multiple Access (NAMA) and Traffic Adaptive Medium Access (TRAMA)

NAMA [23] and TRAMA [9] are closely related, but NAMA was not designed with AHWSN in mind while TRAMA was, therefore only TRAMA is investigated.

TRAMA splits the channel up into contention and non-contention. The nodes that have data to send contend for a timeslot. Then a distributed election scheme chooses the nodes that will receive a message and assigns them timeslots. The senders are then informed in which TDMA slot they may send. Nodes also exchange their two-hop neighbourhood information to ensure that transmitters and receivers with the same timeslot do not overlap in radio connectivity.

Thus, there is contention which suffers from collisions and control overhead followed by a distributive scheme to assign slots and ensure collision free transmission. This is followed by the data transmission, which follows a TDMA scheme.

Problem Issues:

NAMA: Not designed with energy efficiency in mind for AHWSN

TRAMA: Collisions every duty cycle, control overhead, each duty cycle has a distributive scheme to assign data slots, all of which are energy wasteful.

2.4.5 E-MAC [10]

EMAC avoids the use of cluster-heads by using a distributed and self-organised TDMA scheme. It does employ a hierarchy where nodes fall into three categories and nodes of one category have a timeslot assigned for their supervision. At the beginning of their timeslot is a contention scheme for nodes that need to use a timeslot to send data. The timeslot co-ordinator may also choose to send data in the slot to which it will refuse all other contenders. Timeslots are re-used in non-overlapping areas.

Problem Issues:

A hierarchy is present where nodes are required to rotate their position in it. Contention is used in every round and nodes in charge of TDMA slots must send data every round.

2.4 Contention versus Non-Contention

Table III summarises the issues that were brought up in subsections 2.3 and 2.4.

TABLE III

COMPARING FACTORS OF GENERAL CSMA TO GENERAL TDMA

Requires initialisation before use:	No*	Yes
Requires Awake/Asleep synchronisation:	Yes	Yes
Collisions taking place:	Yes	No*
Overhearing other nodes data transmissions:	Yes	No*
Requires co-ordination by cluster-heads:	No*	Yes
Hidden / Exposed terminal problem:	Yes	No*
Maintains a schedule to avoid collisions:	No	Yes*
Sense the medium before transmission:	Yes	No*
Control overhead:	Yes	No*
Idle Listening:	Yes	No*
Easy Scalability:	Yes	No
Quality of Service:	No	Yes*
Bandwidth Utilisation if many neighbours	Low	Good*
Bandwidth Utilisation if few neighbours	Good*	Low
Latency:	Duty cycle dependent	TDMA cycle dependent

* Starred items are more energy efficient than non-starred items

The discussed protocols of AHWSN all share one common view, and that is energy efficiency. All of them trade off time to deliver a message for energy savings. They fall in to different categories but there is one that has not been investigated:

A TDMA scheme which:

- employs no cluster-heads
- has no need for a hierarchy while in operation such as a master slave relationship
- uses only one channel
- nodes only awake up when they are guaranteed to be involved in communication tasks
- allows neighbouring nodes to sleep during other node's timeslots and are only woken up if they need to receive data

With the aim of improving energy efficiency and trading off other factors such as throughput to achieve better energy savings.

3 COLOUR TDMA MAC

The central idea of the thesis is discussed in this chapter. Firstly, the reasons for the new protocol are stated and its aims are examined. Secondly, a higher level explanation of the protocol is presented. Thirdly, a graphical example of the scheme shows how it works, after which the Colouring Problem is introduced. Following that are the details of the schedule and the PicoRadio is introduced to further energy savings. Finally, the protocol's finer details as well as its advantages and disadvantages are discussed.

The thesis proposal is a solution for the MAC layer, but for added energy efficiency, it incorporates a low power wakeup PicoRadio, which is a part of the physical layer. The proposal focuses on the MAC layer, therefore, it is investigated first and the physical layer is examined in the following chapter.

3.1 The MAC Protocol Proposal – Colour TDMA MAC

Colour TDMA MAC is a new TDMA MAC protocol for AHWSN. It uses a TDMA scheme so that it does not suffer from the four main power issues of contention [6], [8], namely:

- Collisions
- Overhearing
- Control overhead
- Idle listening

It incorporates three attributes together into a unique unexplored TDMA solution that other TDMA AHWSN MAC protocols do not address. Namely:

- 1 A distributed algorithm
- 2 The use of only one channel
- 3 A scheduling algorithm that allows nodes to maximise their sleep time and minimise their awake communications time

All three points contribute to energy savings in the network. By using only one channel, energy is saved as multiple channels are not searched either to locate activity or to find an empty channel. The main aim of Colour TDMA MAC is to improve energy efficiency and to achieve maximum power savings, Colour TDMA MAC trades off throughput considerably.

The name 'Colour TDMA MAC' has been chosen for the proposal because it is TDMA MAC that uses 'Colouring Graph Theory' to ensure that the neighbouring nodes' radios are not simultaneously sending data (also known as internal collisions).

3.1.1 Aims of Colour TDMA MAC

The goals of Colour TDMA MAC are:

- To reduce to a minimum the energy wasted due to collisions, overhearing, control overhead and idle listening in a CSMA/CA scheme,
- To eliminate the need to employ cluster heads or administrative nodes to co-ordinate TDMA,
- To consume low power,
- To tolerate changes in the network (nodes entering and exiting).

The first goal is achieved by using a TDMA MAC protocol rather than a contention based MAC protocol. The second goal is achieved by the fact that a distributed algorithm is used to provide nodes with non-interfering timeslots; i.e. timeslots that do not require co-ordination and only rare synchronisation. The methods to achieve goals three and four are contained throughout this chapter.

3.1.2 Higher Level Abstraction of the Scheme

The Colour TDMA MAC proposal requires many different concepts working together to form the overall protocol. Each different subsection in this chapter is a building block leading to the provision of the overall protocol proposal. For example subsection 3.2 “The Colouring Algorithm” only concentrates on a distributed algorithm that assigns nodes timeslots. Thus, a higher level abstraction of the scheme is presented, so that the reader gains some insight into the idea of the final protocol and understands the flow of each subsection. Point 3 and onwards in the following assumptions will be proved or discussed in the later subsections:

- Nodes are arranged in a flat topology, i.e. in a two-dimensional plane.
- Nodes are stationary – but new stationary nodes may join (see Section 1.5)
- Each node is synchronised with its neighbour.
- Nodes are aware of their neighbours and they have a fixed maximum number of neighbours α .
- The maximum number of timeslots is β .

Figure 3.1 presents an example of an AHWSN with the small circles representing the nodes and the large circles their radio range.

- Timeslots are required for each node to access the medium.
- All nodes within the large red circle are in radio range of the red node.
- Each must have unique timeslots for collision-free communications.

Clearly, an algorithm can issue each node in the network its own, unique timeslot. However, this will not scale well to large numbers of nodes, and it introduces a high degree of latency into the network, as nodes will have to wait for a long time to gain access to the medium. As the radio range of each node is limited and controllable

(illustrated in Figure 3.1), it will not extend over the entire coverage of the network. Thus, slots in geographically distinct parts of the network can be reused. But it must be guaranteed that no two nodes within range of an intermediate node share the same timeslot.

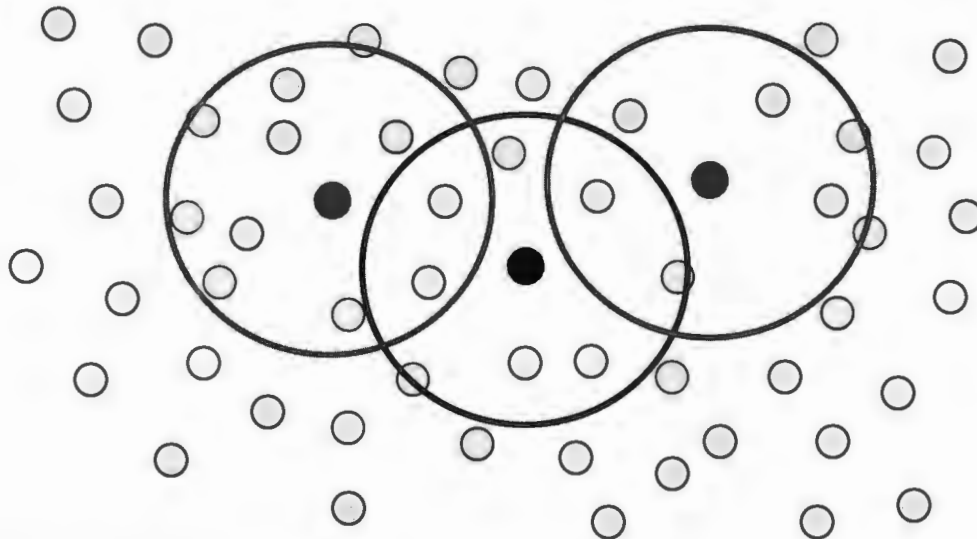


Figure 3.1 Example of an AHWSN with Three Randomly Chosen Nodes Showing Their Radio Range.

Figure 3.2 illustrates how a number, representing a timeslot, is reused over spread out parts of the networks. For example, there is a blue seven in the radio overlap range of green and black, and it can be seen that none of seven's neighbours have chosen the same timeslot and none of seven's neighbour's neighbours have chosen seven as their timeslot. In this way, collisions in the same neighbourhood are avoided.

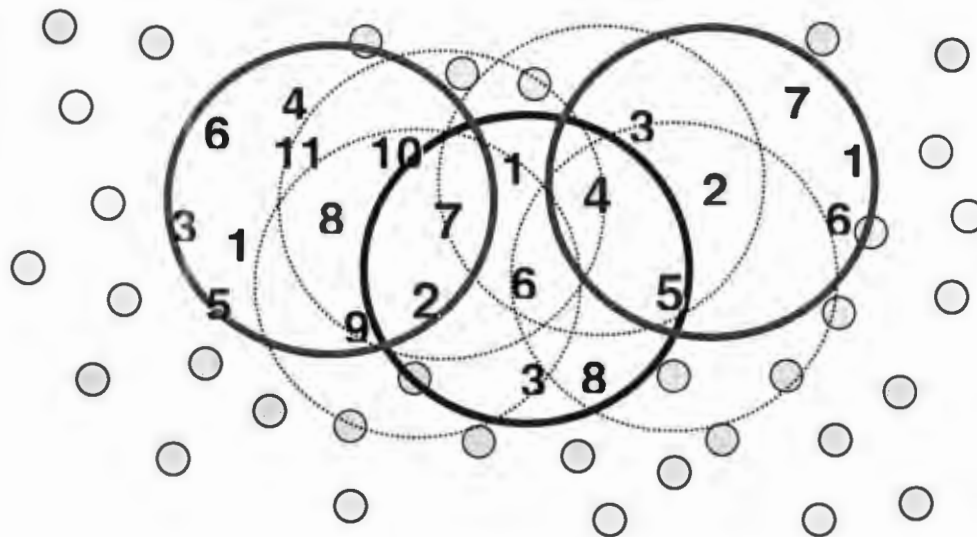


Figure 3.2 Colouring Algorithm Providing Unique Numbers over Two Nodes Radio Connectivity

In Colour TDMA MAC, the channel is divided into timeslots and one TDMA definition exists over the whole network. It is assumed that all nodes are aware of the one TDMA scheme. The node assigned to a certain timeslot is called the channel user because it has authority of the medium / channel for that time period.

- Nodes awake when they become the channel user. This is represented in Figure 3.3 by the dark grey. The light grey means they are asleep and in power saving mode.
- If the channel user has data destined for a neighbour node that is asleep, then it awakens that node.

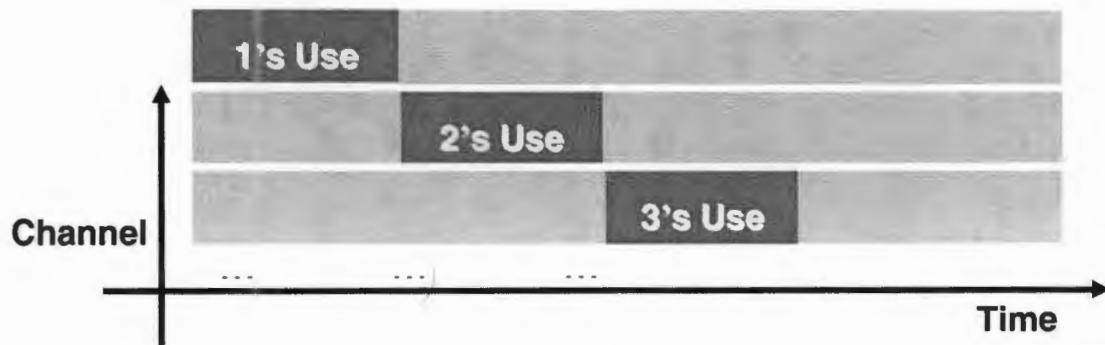


Figure 3.3 Scheduling of the Awake and Sleep Times

Each timeslot of a channel user is further divided into sections as illustrated by Figure 3.4 indicating:

- when the channel user may wake up a specific neighbour
- what time the channel user may send data to the neighbours once they have awakened.

Channel user 1's timeslot

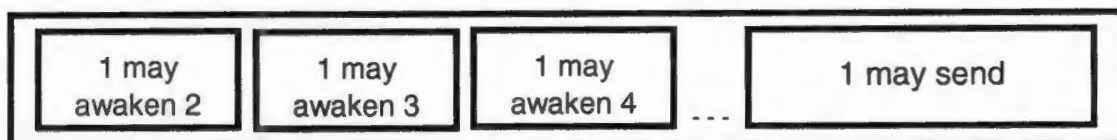


Figure 3.4 Channel User Number 1's Timeslot Subdivision

3.1.3 Colour TDMA MAC Example

This is an example of how the network would run using the information presented thus far. The bold highlighted small circles in Figure 3.5 are the nodes in the network and the bold highlighted large circles are their (ideal) radio range. The circles are the same for all diagrams in Chapter 3, even though radio ranges are never circular; they are merely diagrams to show the concept of the scheme and not to model a real life scenario. The thin, small non-highlighted circles are the sleeping nodes. They have their microcontrollers in power saving mode. The large, thin non-highlighted circles are their respective radio ranges.

The highlighted items of Figure 3.5 are all the nodes in the network that chose Timeslot 1 and are awake. The active nodes may wake their neighbours if they have data for them, and transmit until the end of their slot. Note that there are no cluster-heads or nodes co-ordinating data transfer. Figure 3.6 illustrates the beginning stages of Timeslot 2, while Figure 3.7 depicts the commencement of Timeslot 3.

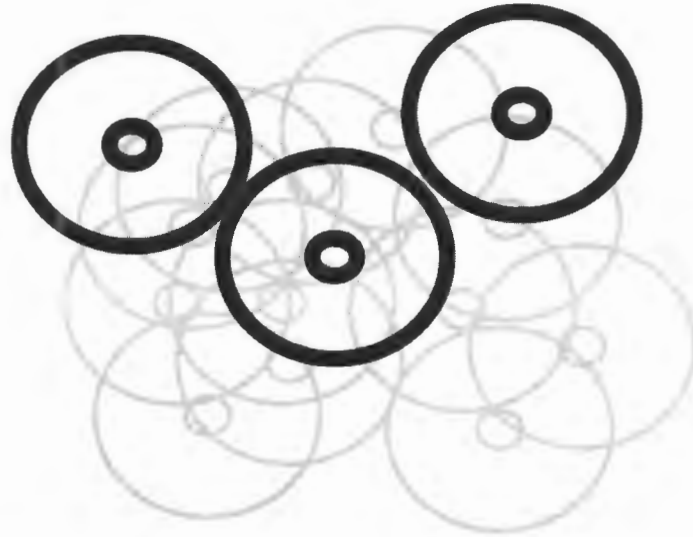


Figure 3.5 Colour TDMA MAC Example with Channel User 1 Nodes

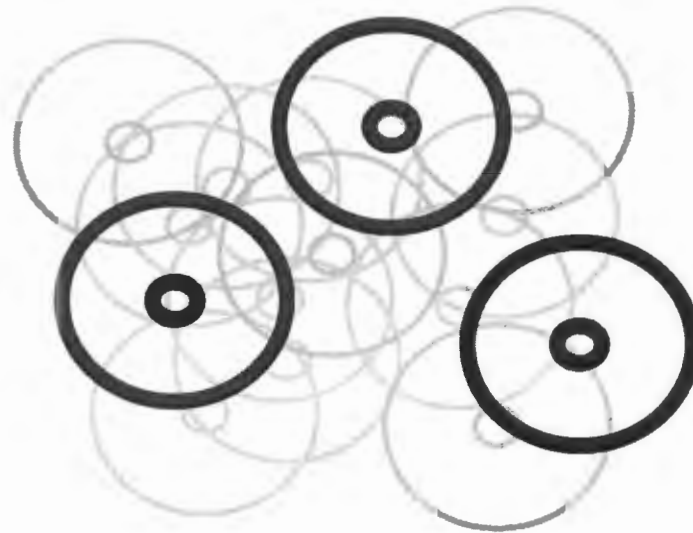


Figure 3.6 Colour TDMA MAC Example with Channel User 2 Nodes

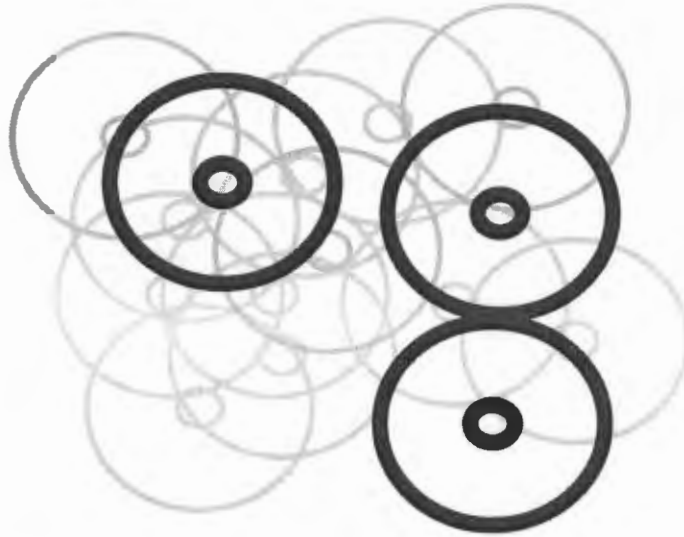


Figure 3.7 Colour TDMA MAC Example with Channel User 3 Nodes

3.2 The Colouring Algorithm

Colour TDMA MAC divides one channel into a fixed number of timeslots. Multiple nodes in geographically distinct parts of the network (where no two nodes are within range of an intermediate node) can reuse the same timeslot. Assigning timeslots, so that none is re-used in a two-hop node radius, is the focus of this section. The solution is an algorithm that stems from graph colouring, which is a part of the mathematics of graph theory. This colouring algorithm will “move” a network in the stages of chaos or contention, or both into the stage of ordered timesharing. It does this in a process called initialisation, which happens at the beginning of the network.

Initialisation is a two part process; the first part it incorporates is synchronisation of the nodes and the second part is timeslot allocation. This section focuses on the second part of timeslot allocation. The section begins with an example of the Four Colour Theorem [33], then sub-sections 3.2.1 – 3.2.4 provide the two-hop colouring algorithm followed by an explanation of the colouring algorithm during initialisation.

Initially, the following issue is considered: A map of the world is required to be coloured in and there are four available colours only. No adjacent countries may be the same colour. What is the solution?

The solution is to code a deterministic computer program that executes the Four Colour Theorem algorithm. It starts at some random point and then uses one colour to fill in one country, then it ensures that the surrounding countries are different. It continues this process until either it reaches an error where two neighbouring countries have the same colour or it manages to solve the problem. If it does run into an error that indicates that there is no solution using four colours.

This country colouring is analogous to the problem that in the Colour TDMA MAC scheme, in that it is necessary to ascertain in which areas of the network a timeslot may not be used. In other words, in that area (or country) of the AHWSN where that colour or timeslot has been taken by another node.

Solving the Colour TDMA MAC “colouring problem”, is not as simple as writing a computer code to start somewhere randomly in the AHWSN and attempt to “colour” the network. In the case of the map of the world, the layout of the map is known and it is possible to start anywhere. However, in an AHWSN, the entire network is unknown at any point and only one non-predictable point starts the process. It is non-predictable because one method of initialising the network is for each node to count down to a large random number and the first to reach it starts the initialisation process.

Also the computer “moves” over to every point on the map and colours it in, whereas in an AHWSN one point starts the process but with its limited radio range it can not tell distant nodes which colour to choose. Thus, with the computer there is a centralised co-ordinator that informs every area on the map what colour they must be, whereas with an AHWSN one random point starts the colouring algorithm and it cannot inform all nodes what colour they must choose. Thus, a distributed colouring algorithm is required for AHWSN because a deterministic computer can not be used to solve the problem.

3.2.1 One-hop and Two-hop Neighbours [9], [16], [23], [24], [25]

Colour TDMA MAC stipulates one TDMA definition over the entire wireless network where multiple nodes share the same timeslot in spatially distinct regions. Immediate neighbours, also known as one-hop neighbours, may not choose the same timeslot. This ensures no internal collisions between direct neighbours but internal collisions may still occur due to the hidden terminal problem [26] of wireless networks, illustrated by Figure 3.8. If a node A chooses Timeslot 4, its neighbour B (which is one hop away) may not use the same timeslot. However, if its neighbours' neighbours (i.e. node B's neighbour), node C, also chooses Timeslot 4, then there will be collisions at neighbour B. Thus, one-hop neighbours would suffer by not being able to communicate due to collisions at an intermediate node.

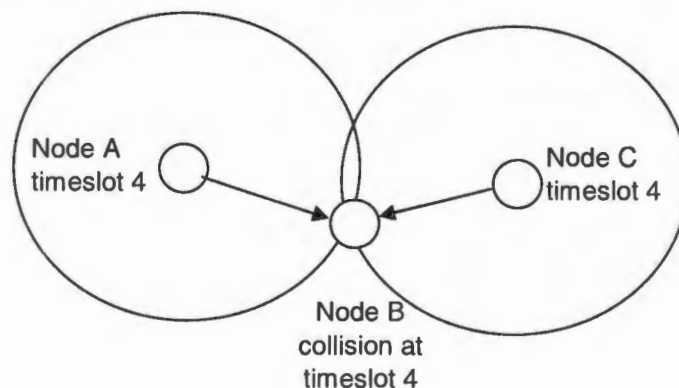


Figure 3.8 Node A and C both use Timeslot 4 and there is a Collision at Node B

An algorithm is required to stop the hidden terminal problem from arising. The solution stems from the two-hop colouring problem in graph theory. Clearly, this means that all nodes must know the timeslots chosen by their one-hop and two-hop neighbours. Thus in node A's table of one and two hop neighbours, no other nodes may possess timeslot four. However, this implies that there can be no unidirectional radio links in the AHWSN. Radio connectivity in the figures thus far has been shown as a perfect circle, which it almost never is, as shown by the scenario in Figure 3.9. Even ellipses do not truly illustrate the radio range of a practical transmitter and one ellipse does not imply that that is the sending and receiving range of the node.

3.2.2 Unidirectional Radio Links

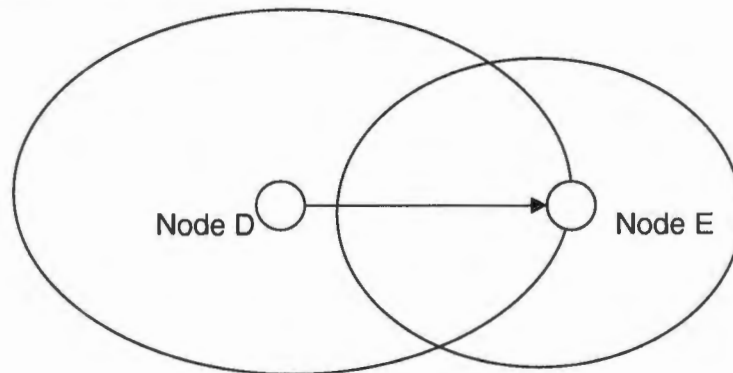


Figure 3.9 Node E is in Radio Range of Node D, But Node D is Not in Radio Range of Node E

In Figure 3.9, Node E can hear its neighbour D but it cannot communicate with D. It must still count D as a neighbour and record node D's chosen timeslot. Node E must inform its neighbours that such a slot is taken by an inaccessible neighbour. Thus, none of Node E's neighbours will attempt to send messages to Node E in Node D's timeslot.

3.2.3 The Two-Hop Colouring Algorithm and Determination of α & β

The two-hop colouring algorithm, which stems from graph theory, ensures that a timeslot that a node chooses is not used by its one-hop or two-hop neighbours. There are other MAC protocols [16], [29], [30], [31], [32] for AHWSN that use the two-hop colouring algorithm. All of them use a randomised method of colouring the graph. The reason for this is that no single entity in the network knows the layout of the nodes. In other words, the positions, radio ranges and the number of neighbours each node has is not known by one single node in the network able to optimally assign colours. Because of this, a distributed algorithm is used (and also for reasons of communication and computational efficiency). The algorithms the other MAC protocols use roughly follow these four general steps:

- 1 Determine the number of colours required.
- 2 Let nodes discover how many, and what colours are available.
- 3 Randomly choose a colour from what is offered.
- 4 Let the rest of the nodes know what has been chosen.

Points 2 to 4 require less investigation than point 1; point 2 requires that a random number be chosen, point 3 requires the chosen number to be broadcast and point 4 the node listens to the data of its neighbours. Therefore the rest of this sub-section 3.3.3 concentrates on determining how many colours are required and issues regarding that.

The algorithm for assigning timeslots is derived from [16] and it has been adapted to Colour TDMA MAC. [16] is a non-contention based multiple channel MAC protocol for AHWSN that uses multiple frequencies (FDMA scheme). Colour TDMA MAC only uses one channel. [16] uses a distributed colouring scheme to assign different channels to each node.

The algorithm for determining the number of colours required in [16] starts by defining a network as a partial graph G . This incomplete graph consists of two variables: V and E .

$$G = (V, E).$$

V (for vertices) represents the nodes, and E (for edge) corresponds to the radio links between nodes [16]. The degree of the graph G is the maximum number of neighbours (α) a node can have, and it is denoted as 'd' [16]. Using the Brook and Vizing theorem, in the area of the AHWSN where d occurs, the chromatic number of colours that is needed is:

$$d(d-1)+1 \tag{3.1}$$

and in less dense areas, it is less than that [16]. The number of colours is equivalent to the number of timeslots that Colour TDMA MAC requires. Thus, the number of time slots in the network is given by:

$$\beta = \alpha(\alpha-1) + 1 \tag{3.2}$$

Colouring is an NP complete problem [16]; hence if β is the number of timeslots then the nodes require β chances to pick a timeslot. Nodes pick a timeslot using a contention based scheme. Thus, if there is even one collision, more than β chances are required to pick a timeslot. If a node has one chance every duty cycle to pick a timeslot, then at a minimum β duty cycles are required before a specific group of nodes all have a non-interfering timeslot. But, in one duty cycle, the node has more than one chance to pick a timeslot. If the contention scheme uses the S-MAC schedule to contend for timeslots (Figure 2.2), there are 31 contention slots during RTS and CTS when the nodes can send a control packet to request a certain timeslot. Thus providing the nodes β duty cycles is a generous amount of time for timeslot allocation.

If the network is not a complete graph, then equation 3.2 is valid. A complete graph in a two-dimensional (2D) case would be a circle (or a square, or a triangle or any two-dimensional enclosed shape), and in the three-dimensional (3D) case it would be a

cube. Thus in a 2D network with the number of neighbours, $\alpha = 2$ and a 3D network with $\alpha = 3$ there will there be a possibility of equation 3.2 being invalid.

An example of where the equation, and therefore the colouring algorithm, fails is demonstrated via examples in Figures 3.10, 3.12, 3.14, and 3.16, where the circles in the figures are the nodes and the lines between them represent a two-way radio link.

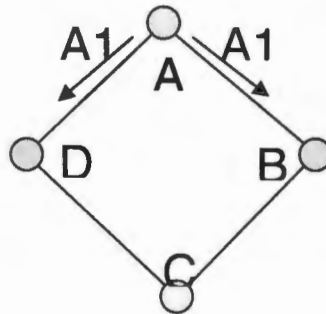


Figure 3.10 Round One: Node A Sends Out a Message that it has Chosen Timeslot 1

Figure 3.10 is a small two neighbour network, (where $\alpha = 2$ and thus $\beta = 3$ from Eq. 3.2). Node A starts the colouring algorithm in round one. It informs its neighbours (represented by the arrows) that it possesses timeslot one. Thus, the nodes populate their information tables as shown in Figure 3.11 where the numbers represent the timeslots and the letters represent the nodes. This format is the same in Figures 3.13, 3.15 and 3.17

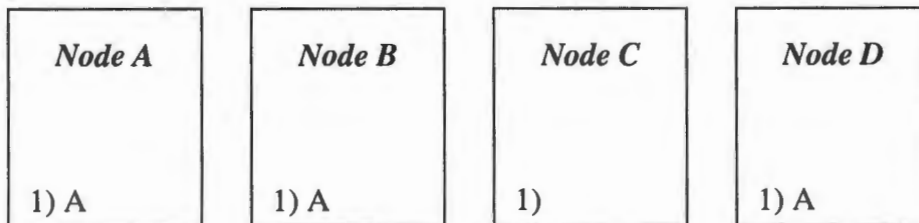


Figure 3.11 Round One: Nodes A, B, C and D Neighbourhood Information Tables

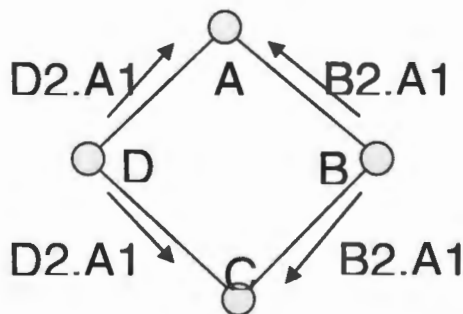


Figure 3.12 Round Two: Nodes B and D Sends Out a Message requesting a Timeslot and Confirming Node A has Timeslot 1

In round two (Figure 3.12), nodes B and D send out a request for a specific timeslot. There is a conflict that B and D are unaware of (as they are not direct neighbours) and A randomly decides who obtains the timeslot and transmits that information. Their tables are shown in Figure 3.13

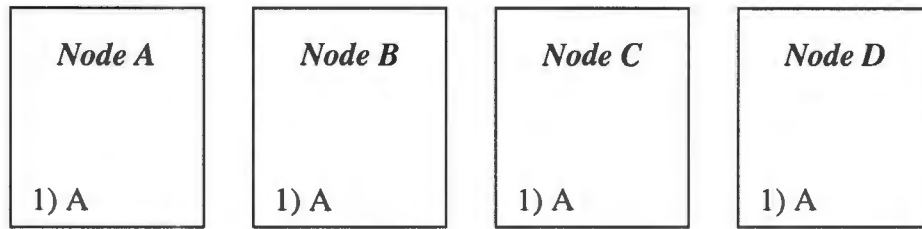


Figure 3.13 Round Two: Nodes A, B, C and D Neighbourhood Information Tables

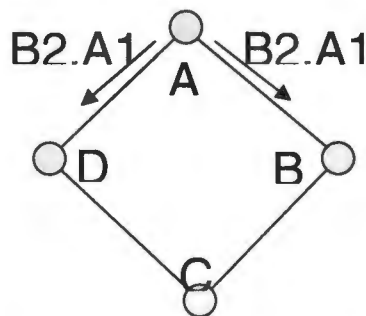


Figure 3.14 Round Three: Node A Announces Who Won the Timeslot

Nodes B and D assumed they received timeslot two. However, in round three, node A (Figure 3.14) now informed them who won and they adjust their tables to reflect this new information (shown in Figure 3.15).

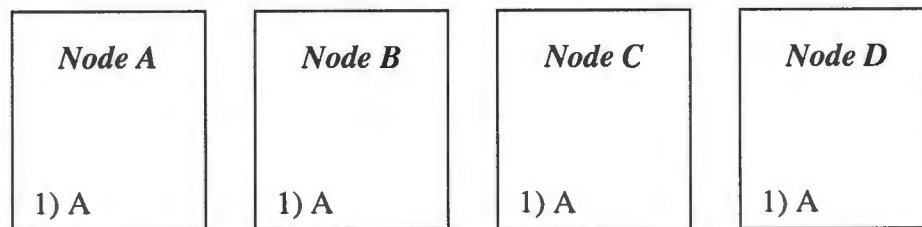


Figure 3.15 Round Three: Nodes A, B, C and D Neighbourhood Information Tables

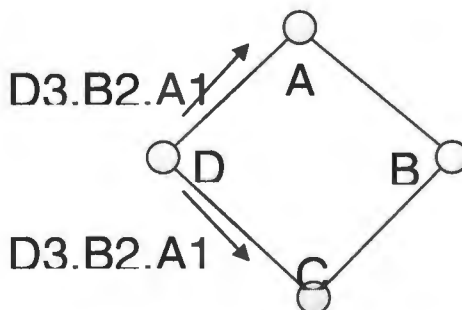


Figure 3.16 Round Four: Node D Sends out a Request for a Timeslot

Round four takes place as shown in Figure 3.16 and node D is requesting timeslot three. The network's neighbourhood information table appears in Figure 3.17.

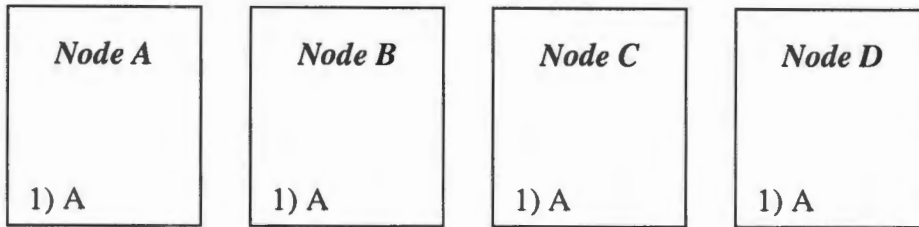


Figure 3.17 Round Four: Nodes A, B, C and D Neighbourhood Information Tables

Node C has nothing available to it. Any complete graph will result in the same issue. Since nodes do not know the global network setup, there is a need to resolve this issue. The solution is in Section 3.4.2. “Number of Neighbours is More Than Maximum and Radio Range Can Not Be Reduced” is not a complete solution. It is future work and a suggestion of how this issue could possibly be resolved.

3.2.4 Sacrificing Channel Usage for Collision Avoidance

Figure 3.18 investigates the effect of the number of neighbours on the number of timeslots required

$$\beta = \alpha(\alpha - 1) + 1$$

if α (maximum number of neighbours) equals three, then β (number of timeslots throughout the network) is seven. The issue is that as α increases, β rises exponentially and thus the channel efficiency decreases rapidly. For example, for a node ‘A’ with a maximum of four neighbours, thirteen timeslots are required. The channel efficiency for node ‘A’ and its one hop neighbours, if each neighbour uses their timeslot, is five out of seven and that is less than 40% ($5 / 13 = 0.38461$).

An example of inter-nodal *throughput* follows:

One node is sending to one neighbour. BW is the link bandwidth. The *throughput* of the network will be: BW / β^2 or in other words: BW / α^4 , which is an extreme trade-off. If a node is a seven-hop neighbour to the information sink, then seven hops of BW / α^4 are required for the data to reach its destination.

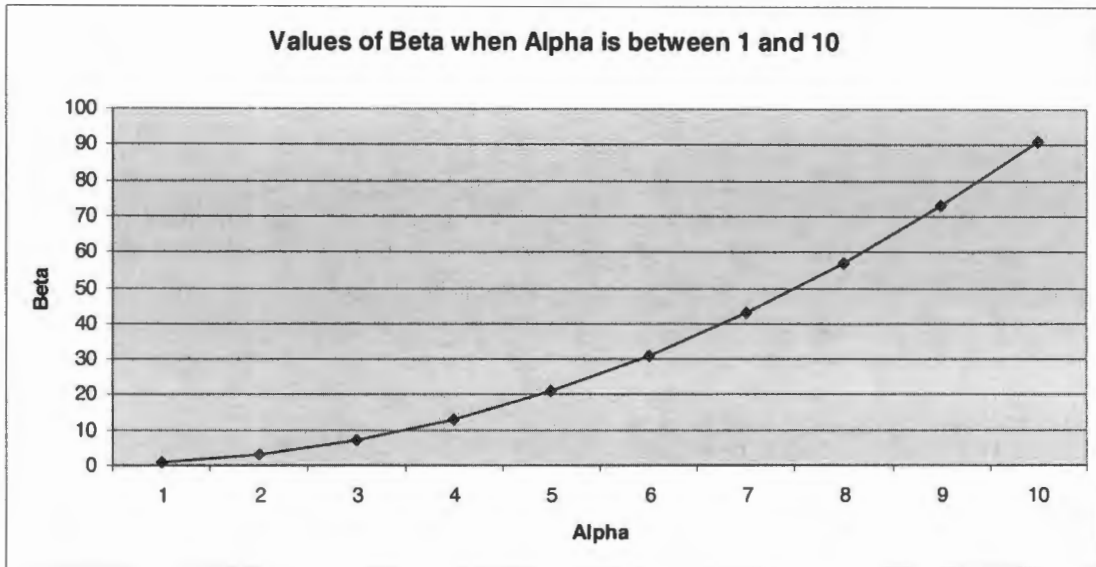


Figure 3.18 Values of β when α is between 1 and 10

Colour TDMA MAC stipulates one TDMA definition over the whole network. Thus, when a value for α is chosen, β is set for the whole network. AHWSN have different node densities over different sections as shown in Figure 3.19. The figure illustrates only some of the radio ranges of the nodes in the figure. If α is four (because at the far left of Figure 3.19 there are four neighbours) and the average number of neighbours is two over the rest of the network, then channel efficiency is about one-quarter over most of the network if channel usage is maximum. Ideally, the colouring algorithm should provide different numbers of colours for different densities over the same network but this is complex with no simple solution.

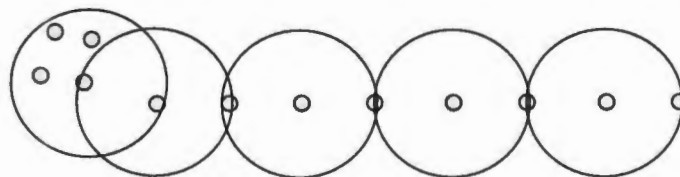


Figure 3.19 One AHWSN with Different Node Densities over Different Areas

Channel usage is therefore being traded off to ensure that no internal collisions occur. The question to be considered is whether this is acceptable. An examination of [1] showed that when the duty cycle was introduced into AHWSN, it meant that one message over a ten hop network took eleven seconds to reach its destination compared to under two seconds with no sleep cycles. But the energy consumption over ten hops using a duty cycle was 5 Joules versus the 28 Joules of the no duty cycle [1]. In the same way, Colour TDMA MAC is trading off bandwidth use for energy savings.

3.2.5 Assumption in Initialisation

There is an assumption that is required for the next sub-section 3.2.6 which will be proved later in Section 3.5 "Initialisation." The assumption is that the nodes which are

about to enter the colouring algorithm are synchronised, that they do not have more than α neighbours, and that they know the TDMA scheme but have not yet chosen timeslots. This implies that any node starting the second phase of initialisation (which is colouring) has already completed the first phase (of synchronisation and neighbourhood discovery).

A distributed algorithm is used for colouring as Colour TDMA MAC does not employ cluster-heads but has a flat arrangement. However, during initialisation there is a need for one node to inform its neighbours when they can start the algorithm and when they must wait. Thus, a rule exists to ensure the colouring algorithm does not attempt to execute over the same area that synchronisation and neighbourhood discovery is taking place. The rule is that a node may only “colour” once 3β duty cycles have passed after synchronisation. This is explained much later in subsection 3.5.4 “The Timeslot Allocation Process.”

3.2.6 The Colouring Algorithm during Initialisation

The nodes are synchronised, they do not have more than α neighbours and they know the TDMA scheme. Thus, the total number of timeslots is known by each node because they know the TDMA schedule.

The algorithm is very basic and follows these three steps in a cyclic manner until it has settled on a non-conflicting timeslot or initialisation is over.

- 1 Discover what timeslots are available.
- 2 Randomly choose a timeslot from those not chosen.
- 3 Let every neighbour know your neighbour information.

The nodes await 3β duty cycles to compete for timeslots. They are productive in making the colouring scheme converge more quickly while they wait, by broadcasting their chosen timeslot to their neighbours. They do so in a contention based manner during any node’s data transmission period (See Figure 3.4)³. If they hear from their neighbour that it is going to compete for the same timeslot, then they randomly choose either to stick to that slot or to change. If they change, they only select from the “available” timeslots which have had no requests. If they hear from their neighbour that it has been assigned a timeslot, then they enter that information into their neighbour information table.

At the end of 3β duty cycles and the commencement of the colouring algorithm, the nodes transmit a request to book a timeslot. All neighbours who hear the request add

³ This will not affect any neighbours that have those timeslots assigned to them because no data transmissions occur during initialisation.

it to their table of information and when they make their request, they also transmit which timeslot requests they have heard.

Nodes make decisions based on their current neighbour timeslot allocation information. If a conflict happens and two or more nodes choose the same timeslot, the winner is determined either by which node has fewer choices of timeslots (if that information is available) or by which node has the lowest identification number (see Chapter 1, section 1.4 “Intrinsic AHWSN Node Properties”). If the nodes have the same number of choices, then the node with the lowest identification number wins the timeslot. This is discussed further in 3.2.7 “Avoiding the Ripple Effect.”

Previously in section 3.2.3, it was shown that complete graphs (circle networks) with an $\alpha = 2$ cannot be coloured using equation 3.1. Figure 3.20 illustrates an incomplete graph with loops in it. One of loops in Figure 3.20 is BFGCB. An investigation is required to find out if the equation 3.1 and the colouring algorithm will work for an incomplete graph with loops. This is demonstrated in Figure 3.20 where most of the nodes have a timeslot assigned already and the number of timeslots is seven ($\alpha = 3$).

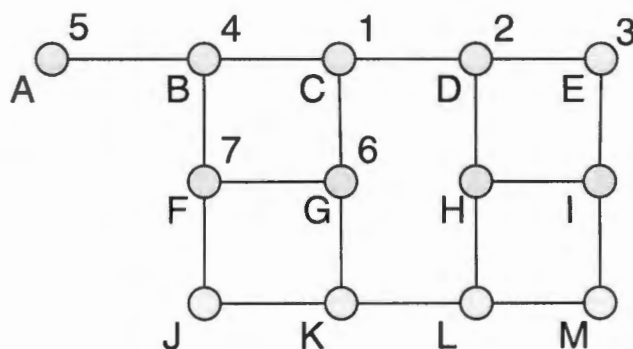


Figure 3.20 Another Example of the Colouring Algorithm with $\alpha = 3$

In Figure 3.20, nodes K and H are required to find a timeslot number. It can be seen that it is potentially possible to make the mistake of thinking that there are no available timeslots for K and H. This is because looking at node C, and its two-hop neighbours, C has already filled in timeslots one to seven, thus that there are none left. This is incorrect because the colouring algorithm does not state that two-hop neighbours’ timeslots are to be unique. The correct viewpoint is that each timeslot is required to be unique over a two-hop range. One node may have two two-hop neighbours who use the same timeslot but who are not two-hop neighbours to each other. Thus, K has the choice of timeslots two, three, four and five. H has the choice of four, five, six and seven.

Figure 3.20 has ten loops inside it and it is an incomplete graph. There seem to be no issues pertaining to the assignment of timeslots and the author cannot think of an example where the graph is incomplete, has loops and equation 3.2 fails. Hence, provided the graph is not complete, the minimum number of timeslots in equation 3.2 may be used without failure.

3.2.7 Avoiding the Ripple Effect

The ripple effect occurs when a node that has been told to change timeslots, changes timeslots but then causes its neighbours to change timeslots, causing their neighbours to change timeslots and so on. The ripple effect is highly undesirable due to the amount of energy that would be used. Almost every node in the network would have to change its timeslot. Ideally, a node should not be able to tell another node to change timeslots, but during initialisation, if two nodes choose the same timeslot then the common neighbour of both nodes must inform one of them to change. Thus, two policies are enforced to stop the ripple effect:

- 1 A node that has been told to change its timeslot must choose another timeslot and if it can not find any available slots then it must go to sleep.
- 2 One node *can tell* another node to change time during initialisation only. After initialisation, nodes may only *request* that a neighbour change its timeslot.

Initialisation

During initialisation, it may occur that two of node A's one-hop neighbours have chosen the same timeslot. Node A must now inform either both of them or just the one to change. The following are the choices it has to derive a final decision:

- If both of node A's neighbours choose the same timeslot, then A tells both to choose again provided it knows very little about the neighbours of each.
- If both of node A's neighbours choose the same timeslot and node A is aware of what each nodes' neighbours have chosen, then it will tell the node with the fewer free timeslot options to claim the timeslot and the other node to choose again.
- If one of node A's neighbours has had the timeslot for a few duty cycles already, then A informs the node that has just chosen the timeslot to change.

During Scheme Operation

After initialisation, new stationary nodes may join the network or leave it (1.5 "Assumptions"). The protocol for dealing with newcomers is discussed in subsection 3.3. The following flow diagram in Figure 3.21 illustrates the decisions involved when node A requests its neighbour B to change its timeslot due to a newcomer requesting that timeslot.

Three examples to illustrate how these rules work now follow:

Example one:

The network is in its initialisation stage and the scenario in Figure 3.8 occurs where two nodes are making requests for the same timeslot. The node (or nodes) in the overlapping region of the two radio ranges have little information about the network. They send a message to the two informing them that they chose the same timeslot and

that both must change. Thus, the two choose another timeslot, including the original timeslot as one of the choices.

Example two:

The network is in its operational stages and a newcomer wants to join the network. It notices that if one node changes its timeslot then it will be able to join in on the scheme. It makes a request. The node replies no, it has had that timeslot for many cycles.

Example three:

The network is in its operational stage and the environment changes and a node or several nodes now have more neighbours in their radio range (bidirectional), or that they can hear but cannot transmit to (unidirectional). The “old” neighbours and the “new” neighbours both have had timeslots assigned to them for a long time but now there is a clash and both sets of neighbours are reluctant to change timeslots. Those nodes experiencing the collisions have two decisions:

- 1 Accept that there will be a collision and send out a message that it will ignore those neighbours and not turn on its transmitter for those neighbours' timeslots.
- 2 Send a message out that it is exiting the network, for a period of time.

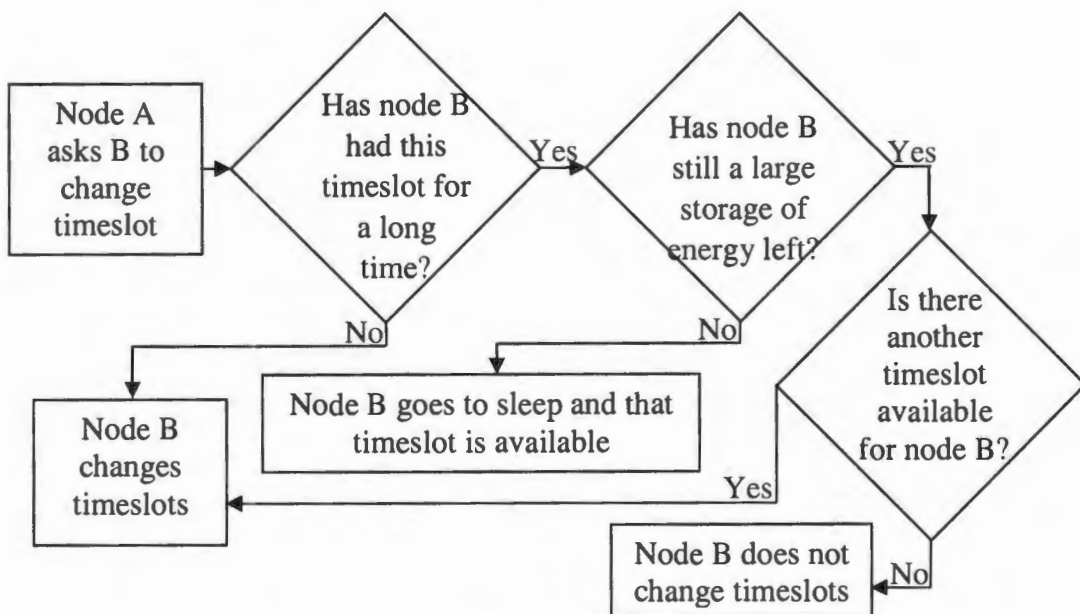


Figure 3.21 Flow Diagram of Choices Node B Computes if it is Requested to Change Timeslots during Scheme Operation

3.2.8 The Colouring Algorithm after Initialisation

When the network is no longer in its initialisation stage, the colouring algorithm does not apply. A node wanting to join the scheme after initialisation follows the steps in 3.3.6 “Special Events and Node Colouring After Initialisation.”

3.2.9 The Hidden and Exposed Terminal Issue

TDMA MAC does not suffer from the exposed terminal problem due to each node having a timeslot to transfer data. The hidden terminal problem is also easily solved: The Colouring Graph Algorithm does not allow a node to choose a timeslot over a two-hop neighbour range.

3.3 The Scheduling Algorithm

The colouring algorithm assigns a timeslot to nodes. The timeslots form part of the TDMA scheme and in this section that schedule will be explored.

3.3.1 The Awake/Asleep Schedule

With regard to the communication needs of a AHWSN node, a node’s scheduling algorithm follows this rule:

A node wakes to transmit when both the following conditions are met:

- The node’s chosen timeslot is active.
- It has data or control information to send.

If these criteria are not met, a node will enter a sleep mode for power efficiency by disabling its transmitters. The node awakens briefly in each neighbour’s timeslot to check if there is data to be received.

3.3.2 The Scheme

The scheduling algorithm can be summarised and explained best by means of a diagram such as Figure 3.22, where it is assumed that the awakened node can instantaneously receive data.

In Figure 3.22, all nodes that have chosen timeslot 1 in the TDMA schedule awaken in the “1’s” dark grey time period. Their neighbours such as “2’s” and “3’s” are all asleep as indicated by the light grey. All “1’s” may use the medium and are called the channel users. If ‘n’ is an integer bounded by the number of timeslots in the TDMA scheme, then channel user ‘n’ corresponds to all timeslot ‘n’ owners. All of the channel user 1s’ neighbours are asleep with their transmitters off. Each neighbour must awaken briefly to check with the channel user if it has data to send them, hence channel user 1s’ timeslot is subdivided into discrete slots in which each neighbour awakens in turn to check if the channel user has a message to send them. This process is represented by the blocks “2”, “3”, “4” down to “β” in Figure 3.22. Data transmission occurs instantaneously and then the neighbour goes back to sleep.

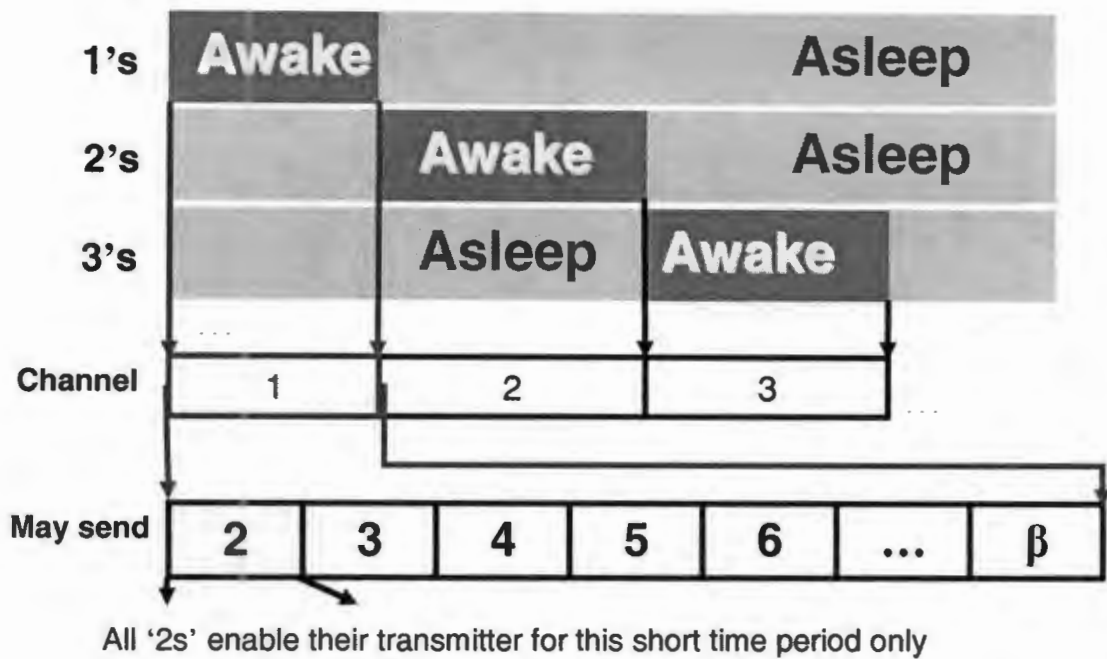


Figure 3.22 The Scheduling Algorithm With Instantaneous Wake Up and Data Transmission

The period when “all ‘2s’ enable their transmitter” illustrated in Figure 3.22, is very short due to instantaneous data transmission. This is applicable to transmitters operating in a high gigahertz range. However, this would not be the case if, for example, the nodes were at the bottom of the sea and the process of sending a message was long and there were many packets. In such circumstances, the scheme must be rearranged as shown in Figure 3.23 below.

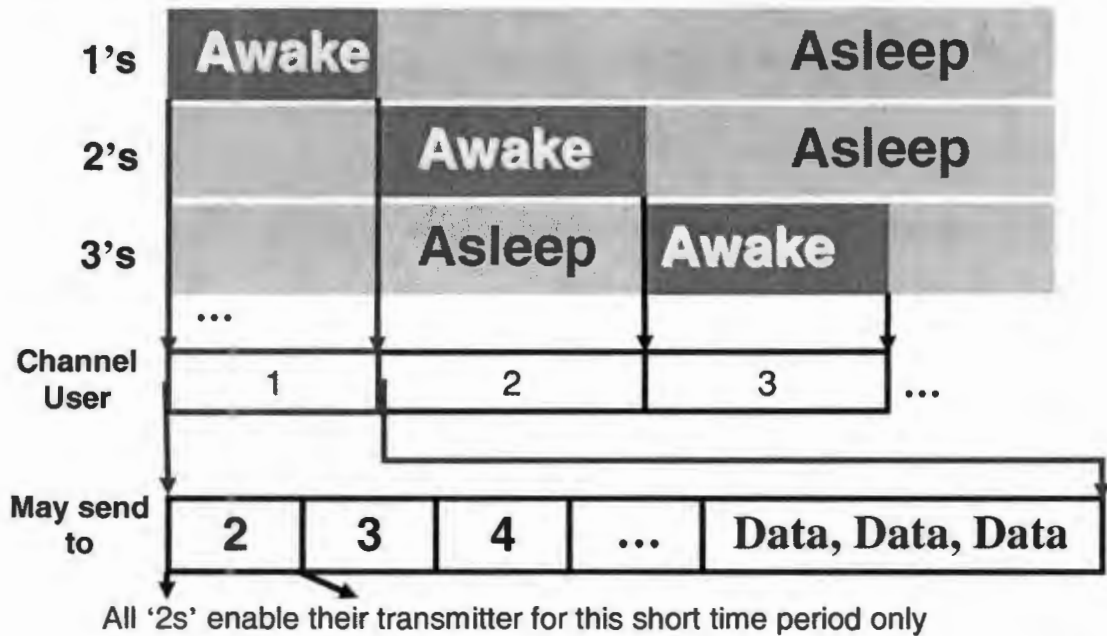


Figure 3.23 The Scheduling Algorithm Accommodating Radio Wake Up and Non-Instantaneous Data Transmission

3.3.3 Broadcast and Unicast

It should be clear that this scheme is useful for unicast traffic where the destination is known and there is only one node such as when channel user 1 sends a message to timeslot owner 3. Allowing one source to talk to one destination keeps the overall network usage low and energy savings high. It is less advisable for broadcast traffic, where all the channel user's neighbours must obtain the same data. Generally, the MAC layer in AHWSN avoids broadcasting because it is expensive in terms of the whole network's energy but it can be used by the higher layers for route discovery.

There is no direct accommodation for broadcasting in Figure 3.23, such as one subdivision where all neighbours enable their radio transmitters at the same time. Thus if a broadcast is required the channel user informs each neighbour in its subdivision that it has a message for it. Channel user's that must do a broadcast and unicast messages, wake up all neighbours and then do the broadcast first. After the broadcast, only the unicast transmissions neighbours stay awake.

3.3.4 Special Events and the Final Schedule

Figure 3.23 is still an incomplete picture of the scheme missing: administrative requirements, radio wake up, and guard times. Figure 3.24 has therefore been prepared to incorporate the requirements which have been left out of the picture.

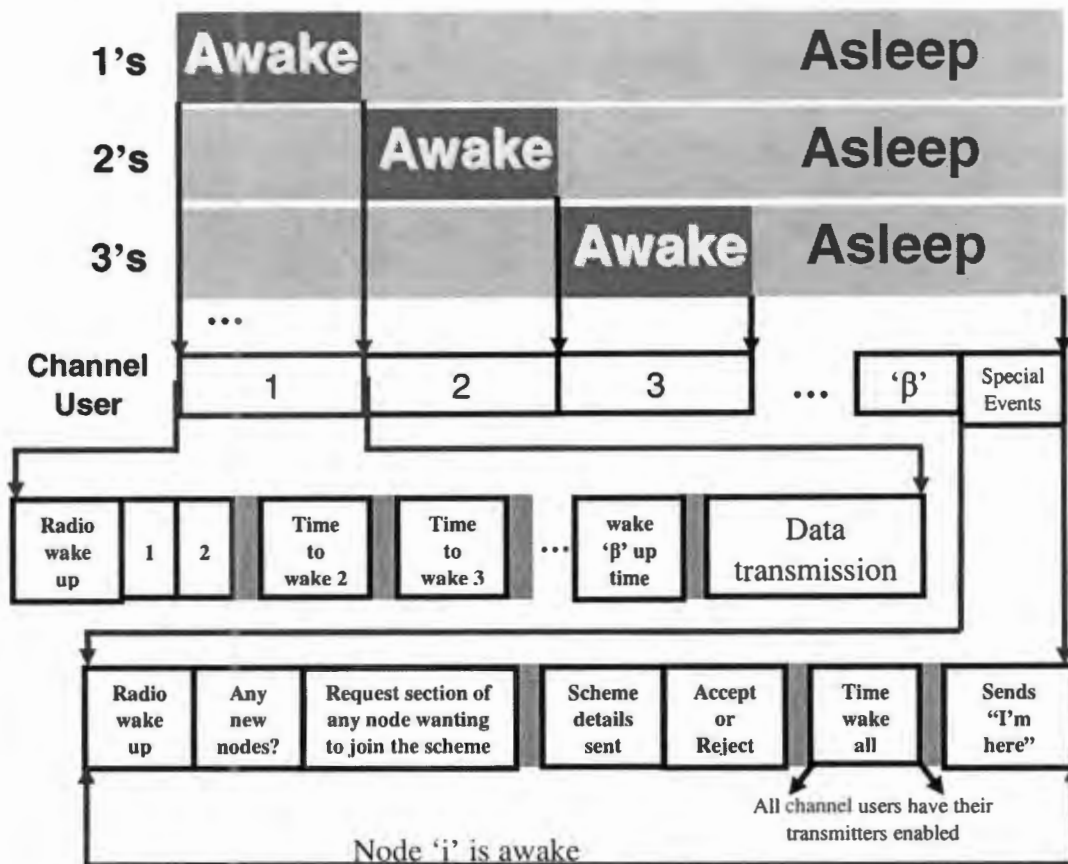


Figure 3.24 The Scheduling Algorithm in Detail

Figure 3.24 is a realistic diagram of the TDMA scheme and it will be discussed in the next few paragraphs. The discussion begins in at the block “Radio wake up” of channel user 1’s timeslot. It then moves block by block left to right until “Data transmission.” Then the discussion moves to the string of blocks below and moves again from left to right. It starts at the special events “Radio wake up” and ends at “Sends I’m here” and briefly mentions the importance of each block

When channel user 1 gains access to its timeslot, it powers up its radio transmitter. Depending on the microprocessor this may take $650\mu\text{S}$ (Nordic Semiconductor nRF9E5) to about 3mS.

After the channel user 1’s radio wake up illustrated in Figure 3.24, there are two blocks labelled ‘1’ and ‘2’. These blocks have two functions:

- 1 To help a node discover if there is a neighbour who is using the same timeslot as itself.
- 2 To assist neighbours that have no timeslot to obtain one.

Periodically but infrequently, the channel user must transmit its table of neighbours and the timeslots the neighbours are assigned. This table is called the neighbour information table. The channel user randomly chooses between the two slots ‘1’ and ‘2’, and submits a control packet during that interval. The control packet informs any ‘listening’ neighbour that the neighbour information table will be sent in the data section. The listening neighbours are the nodes who do not have a timeslot and want to know what is available and what is not. If the channel user ‘hears’ a control packet in the ‘2’ when it sent its control packet in ‘1’ or vice versa then it knows another node is using the same timeslot as itself. This method of discovering a neighbour with the same timeslot is not totally accurate but it can help.

In Figure 3.24, after ‘1’ and ‘2’ is a guard time which is represented by a dark grey block. It is enforced after ‘2’ and before ‘Time to wake 2.’ It appears again between ‘Time to wake 2’ and ‘Time to wake 3’ and so on. This guard time is to ensure that the channel user does not send a wake up when one neighbour’s transmitter is turning off and another is turning on.

The blocks ‘Time to wake 2,’ ‘Time to wake 3,’ up to ‘Wake ‘ β ’ up time’ are times when each neighbour, in turn, is turning on its radio transmitter. They are listening if the channel user will send it a control packet to wake up because it has data for it.

During the data transmission slot, the channel user sends data to the neighbours it has awakened. If it is to send its neighbour information table then this is what the table consists of:

- The timeslot the node submits in
- The timeslots its neighbours submit in, and
- The timeslots its neighbours' neighbours submit in.

The string of blocks below channel user 1's timeslot belong to the 'special events' in Figure 3.24. The reasons for having the 'special events' timeslot are:

- 1 for nodes to communicate that they are still participating in the scheme and to have a timeslot allocated to them
- 2 to assist neighbours without a timeslot to obtain one.

This entire 'special events' interval is conducted by a certain node in the scheme, namely 'i.' ($0 < 'i' \leq \alpha$). Which particular node is awake and is 'i' is discussed in 3.3.7 "The Choice of Node_i." At "Radio wake up" Node_i turns on its transmitter. In the next block "any new nodes", Node_i sends out a packet asking if there are any nodes that want to join the scheme, i.e. require a timeslot. The next block is a contention slot, where all new nodes request to join the schedule; stationary nodes wanting to join the scheme and obtain a timeslot will be called newcomers.

Following the contention block to join the scheme in Figure 3.24, there is a guard time. In the following block, the TDMA schedule is sent, and then there is the block where node_i sends a message to the newcomers to inform them whether they have been accepted or rejected for the scheme.

In the block "time wake all", node_i awakens all its neighbours and in the block "Sends I'm here" it informs its neighbours that it is still participating in the TDMA scheme. Those neighbours then set a parameter to say that that neighbour will be there for another set number of duty cycles and that its timeslot is in use. If neighbours do not periodically receive an "I'm here" message from a neighbour, they assume that that neighbour has completely depleted its energy source and they set that 'dead' neighbour's timeslot as available for newcomers. More details on node_i appear later in subsections 3.3.6 and 3.3.7.

This concludes the exploration of Figure 3.24. The next section presents an example of one of the neighbourhood information tables presented in this subsection. One node transmits the table so that newcomers and neighbours know which slots are available and which are not.

3.3.5 The Neighbour Information Table Example

Newcomers and neighbours are listening for the neighbour information table. Figure 3.25 shows a Colour TDMA MAC AHWSN where the numbers are the nodes and the timeslot they possess. Figure 3.26 illustrates the same nodes as Figure 3.25 but with their unique identification number, which for simplicity is represented by two alphabet characters. Table IV shows the neighbour information table for node six in Figure 3.25 with identification “HH” in Figure 3.26.

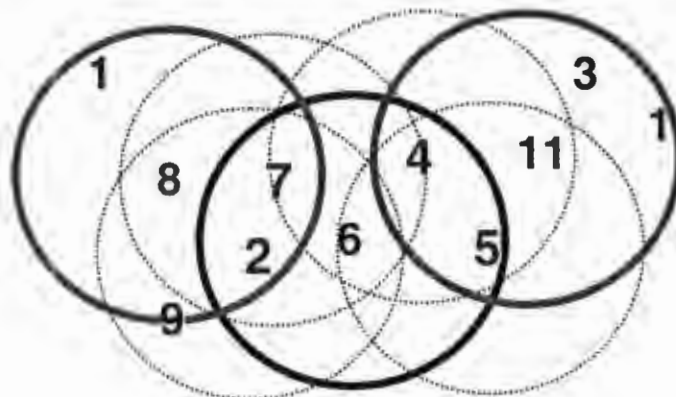


Figure 3.25 Example Colour TDMA MAC AHWSN with a Maximum Limit of Four Neighbours

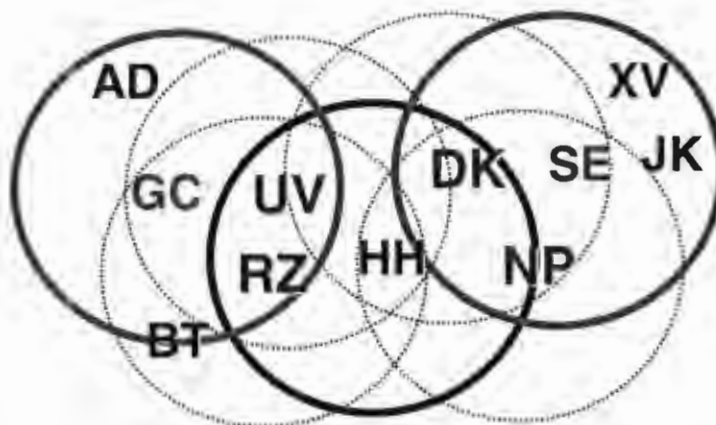


Figure 3.26 Example Colour TDMA MAC AHWSN with a Maximum of Four Neighbours and Their Unique Identification Numbers

TABLE IV

CHANNEL USER 6'S NEIGHBOURHOOD INFORMATION (ID: HH)

Table of HH	One-hop Neighbours	Two-Hop RZ	Two-Hop DK	Two-Hop NP	Two-Hop UV
Timeslot 1					
Timeslot 2	RZ	RZ			RZ
Timeslot 3				DK	DK
Timeslot 4	DK		DK		
Timeslot 5	NP		NP	NP	
Timeslot 6	HH	HH	HH	HH	HH
Timeslot 7	UV	UV	UV		UV
Timeslot 8		GC			GC
Timeslot 9		BT			
Timeslot 10					
Timeslot 11			SE	SE	
Timeslot 12					
Timeslot 13					

Table IV is a sparse table, thus when one node is sending out its neighbour information table, it only sends the filled in entries, rather than the entire table. If the table is dense then a data compression mechanism is used to ensure that the minimum amount of data is sent, thereby saving energy due to less radio transmission.

3.3.6 Special Events and Colouring After Initialisation

Node colouring during initialisation was discussed in 3.2 "The Colouring Algorithm." After initialisation, if a newcomer would like a timeslot, it is up to the schedule to assist them and not the colouring algorithm. This is one of the two purposes of the 'special events' timeslot (see figure 3.24). The two functions of 'special events' are further examined in this subsection.

The colouring scheme has its own individual solution for each node and that node's neighbourhood. Once nodes have been assigned a timeslot, if a node changes its position in the network and does not change timeslots, it will interfere with another

node's timeslot. Hence, in the "Assumptions" in Section 1.5, the nodes are assumed to be stationary. If a new stationary node is added to the network, it may ask for its own timeslot, but first it must establish to whom it must send a request. The Colour TDMA MAC scheme does not employ cluster-heads for a newcomer to talk directly to, but it does have a node 'i' (node_i) which it can ask for help.

The addition of a new node to the scheme is a special event, just like a node leaving the network is a special event. Special events do not happen often, therefore nodes taking precautions for special events in every single duty cycle is a waste of energy. Thus, Colour TDMA MAC does not allow special events to happen round after round. Colour TDMA MAC groups the special events together and it has a small section of the TDMA schedule reserved for them (see Figure 3.27, which is an extract of Figure 3.24). Special events consist of the two procedures of nodes entering and exiting the schedule and they are intertwined to save energy. The node, which wants to inform its neighbours that it is still alive and that it will continue to use its timeslot, is the same node that checks if there are any newcomers.

Colour TDMA MAC makes the newcomer wait with its radio in receive mode until the network is ready to serve it. This makes it an energy efficient solution to the network (but not to the newcomer wanting a timeslot).

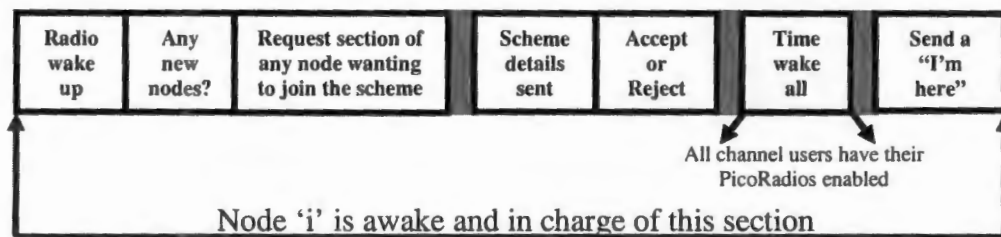


Figure 3.27 The Special Events Section of the Colour TDMA MAC Scheme

Every λ duty cycles, a node participating in the scheme must inform its neighbours that it is still participating in the schedule, so that its neighbours know that they can still communicate with that node and that its timeslot is not available. Thus, it becomes 'node_i' and in charge of the special events section of Figure 3.27. The algorithm to ensure that there is one and only one node_i is presented later in subsection 3.3.7.

Node_i performs two tasks as illustrated in Figure 3.27:

- 1 Informs all its neighbours that it is still participating in the scheme (which is illustrated by the dark section with the white writing of Figure 3.28), and
- 2 Help any newcomers find an available timeslot in the scheme (which is shown by the dark section with white writing of Figure 3.29).

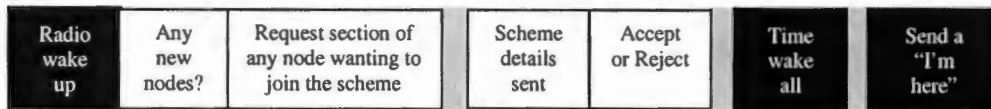


Figure 3.28 Node_i's Message to Tell All its Neighbours it is Still There

Informs all its neighbours that it is still participating in the scheme:

Starting at "Time wake all" from Figure 3.28:

Node_i awakens its neighbours who all have their transmitter on during "time wake all" and in the next block it tells them that it is still part of the scheme. If node_i wants to check that it is still in synchronisation with the other nodes, it asks all neighbours to send a control packet at the beginning of their timeslot (in either blocks '1' or '2' of Figure 3.24) for the next duty cycle. If node_i's neighbourhood information table has any contradictions or problems such as two one-hop neighbours using the same timeslot, then it asks all neighbours to send a control packet during '1' and '2' for more than one duty cycle.

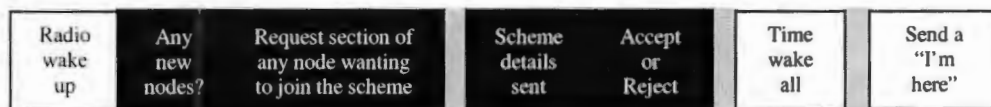


Figure 3.29 Node_i's Helping Section to Any Newcomers

Help any newcomers find an available timeslot in the scheme:

Starting at "Any new nodes?" from Figure 3.29:

Node_i sends out a packet asking if there are any new nodes who want to join the network, to which the newcomer replies. The newcomer then receives a break down of the scheme and all the parameters required to calculate it. Node_i checks if there are any available timeslots from its neighbour table. If there are, it sends a preliminary accept message to the newcomer. This preliminary accept does not guarantee the newcomer will be assigned a timeslot because node_i is not aware of the newcomer's neighbours, which might not have a timeslot for the newcomer.

If a newcomer is preliminarily accepted then:

Node_i sends its timeslot number, its λ , and the current duty cycle number to the newcomer. Node_i also sends the newcomer its neighbour data information in the following duty cycle, in its data transmission timeslot (Figure 3.24).

If a newcomer is rejected then:

Node_i sends the time when the next λ duty cycles will occur. Now the newcomer need not sit around idling. It can put itself to sleep until the next chance to join the network. It may sleep longer than that by multiplying λ by some scalar. If the newcomer is rejected more than once, it can try to lower its radio range and then check if it can see fewer neighbouring nodes.

If a newcomer is preliminarily accepted then node_i asks all its neighbours during the “I’m here” block of Figure 3.28 to send a control packet with their identification number in either 1 or 2 of Figure 3.24 at the beginning of their timeslot for the next three duty cycles. This is so that the newcomer can tell which neighbours it can see and which slots have been taken. It will also be able to hear if it is in an area where two nodes with the same timeslot radios overlap. This will be clear if, in the next three duty cycles, it hears one node in block ‘1’ (see Figure 3.24) and another in block ‘2’ or it hears a collision in one of the blocks.

Node_i might not be in radio range of all of the neighbours of a preliminarily accepted newcomer as shown in Figure 3.24. Hence, the newcomer cannot own a timeslot and become apart of the network until it has heard from all of its neighbours. If the addition of the newcomer leads to the maximum number of neighbours being violated then it will continue to be rejected until the issue is resolved (by a node permanently exiting the network when its energy is exhausted). From the information node_i provided after the accept message, the newcomer can calculate when each of its neighbours will become node_i. This calculation is shown further in subsection 3.3.7. It can then ask each one which timeslots are free in the special events section. It goes through all its neighbours and then calculates which timeslot it can use and it becomes node_i, when no other nodes are node_i, and it announces the timeslot it is using.

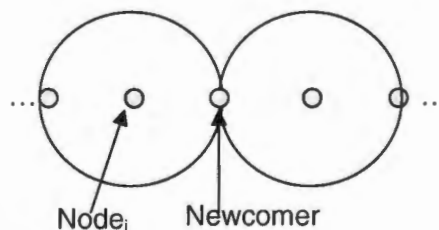


Figure 3.30 Node_i Not being in Radio Range of All of the Newcomer’s Neighbours

The reason why there are slots for ‘1’ and ‘2’ in Figure 3.24 is because the newcomer might be in radio range of two nodes who have the same timeslot but the radios do not overlap sufficiently for them to realise it. Thus, there is a chance that the new node will hear a collision of the signals. If each neighbour randomly selects between 1 and 2 to send their identification, then the chance of the two neighbour’s packets not colliding is 50% per round, if there is a possibility of collision in the area of the newcomer. The neighbours are requested to do this over a number of rounds.

3.3.7 The Choice of Node_i

This section outlines the algorithm to ensure there is only one node_i at the end of a duty cycle.

Every λ cycles it is a node's turn to be node_i. λ is a variable of the protocol but it stays the same throughout the network. The nodes add their timeslot number to λ :

Channel user one adds 1 to λ , channel user two adds 2 to λ , channel user three adds 3 to λ , etc. At the end of $(\lambda + 1)$ duty cycles, channel user one will be node_i. After another $(\lambda + 1)$ duty cycles it will be node_i again. At the end of $(\lambda + 2)$ duty cycles channel user two will be node_i. After another $(\lambda + 2)$ duty cycles it will be node_i again, and so on.

3.3.8 Nodes Exiting the Colour TDMA MAC Schedule

When a node uses up all of its energy, it exits the scheme. Nodes are unaware when exactly this will happen, so every λ duty cycles, a node informs its neighbours that it is still in the scheme. If it does not do this, the neighbours assume the node has exited the network and its timeslot is free.

3.3.9. Network's TDMA cycle

This is a brief summary of the TDMA schedule which concludes this Section. The TDMA cycle has one definition, which exists across the entire network, i.e.

- Number of timeslots in a TDMA cycle (β)
- Length of the timeslots
- Beginning of the TDMA cycle
- Guard times
- Allocated radio wake up times
- Two neighbour information slots
- Special Events timeslot
- Which timeslot the master node just sent that information
- Maximum number of neighbours (α)
- λ

These are the same throughout the network and every node is aware of this information. Multiple nodes across the network may use the same timeslot in the TDMA cycle only if the radio connectivity of one node using timeslot 'x' does not overlap with that of any neighbour also using the same timeslot 'x' with a node in that common region.

If any node consumes all of its available energy, it will not stop the TDMA scheme from continuously cycling through.

3.4 Synchronisation and Neighbour Discovery

Section 3.2 “The Colouring Algorithm” mentioned a two part initialisation. It concentrated on the second part, which assigns a node a timeslot. It assumed that the first part of initialisation was complete, that is that:

- the nodes are synchronised
- they do not have more than α neighbours
- they know the TDMA scheme but have not yet chosen timeslots.

This section will focus on the details of the first part of initialisation – synchronisation and neighbourhood discovery. It will not look at the method of the evaluation of the first part of initialisation; this will be discussed in Section 3.5 “Initialisation.”

3.4.1 Synchronisation

This section outlines the synchronisation technique:

‘If each and every node in the network is synchronised with its neighbours then for any given node’s radio range the network is synchronised; though the entire network may not be synchronised end-to-end’ [34].

The nodes communicate only with their neighbours. There is no need for direct communication from one end of the network to the other. Global synchronisation is not required, only local.

It is assumed that synchronisation between any neighbouring nodes does not have a dramatically large clock drift. Discrepancies are constrained by guard times. Thus there will be no situation of one node needing to be a bridge between two largely differing synchronised nodes.

During execution of the Colour TDMA MAC scheme, a node (node_i) checks for clock drift by asking all its neighbours to send out a packet at the beginning of their timeslot and then aligning themselves in the next duty cycle when they send the control packets. If a channel user awakens just a bit too early, the degree of inexactness is compensated for by the size of the guard bands.

3.4.2 Number of Neighbours is More Than Maximum and Cannot Be Reduced

Every protocol has specific restrictions and limitations. For example, S-MAC has a twenty node limit in a given radio range [1]. Colour TDMA MAC has α as the

maximum restriction of how many neighbours any given node may have⁴. If there are more than α , then some nodes must sleep. In Figure 3.31, a scenario is shown where if node 'A' were to go to sleep, the network would be split in two. This must not occur.

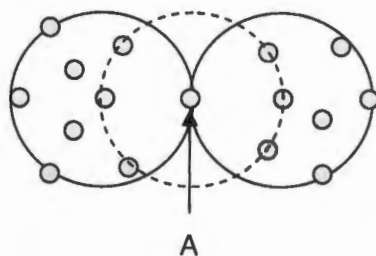


Figure 3.31 Node 'A' is a Gateway Node and if it were to Leave the Network it Would Split it in Two.

Any further solutions, besides going to sleep, are considered future work, such as:

- 1 Half the nodes go into a dormant stage where they sleep for a fixed amount of time and then swap with nodes that were awake.
- 2 Make timeslots become contention based, where nodes who have data to send contend for timeslots with their neighbours who have data to send. The nodes without data to send just sleep.
- 3 Increase α for the whole network.
- 4 Time sharing – where for every even duty cycle one set of nodes run their TDMA schedule and for every odd duty cycle another set of nodes run their own TDMA schedule. It stems from the analogy of time sharing for houses, where you own a house for one week of the year. Another week it belongs to someone else. Other weeks it belongs to other people.

If the routing algorithm can not find a path to the information sink then the nodes must switch off for a long sleep time, after which they try again to re-establish a path.

⁴ In Chapter 5 Results, it is shown that the maximum number of nodes S-MAC and Colour TDMA MAC can function with, is directly influenced by the sensor period and amount of data forwarded per node.

3.5 Initialisation

Initialisation occurs at the start of the network. It is a once off, two-stage process that

- synchronises the network
- instructs the nodes how the TDMA scheme will work
- ensures they have the right amount of neighbours
- assigns them a timeslot.

For maximum energy efficiency, as much as possible is undertaken in the initialisation phase, so that the continued execution of the scheme uses as little energy as possible.

Colour TDMA MAC uses a crystal technique for initialisation. This is where one point in the network starts the process and initialises its neighbours. Certain newly initialised nodes spread the process among their neighbours who in turn initialise their neighbours, and so the process continues.

3.5.1 Initialisation Overview of Colour TDMA MAC

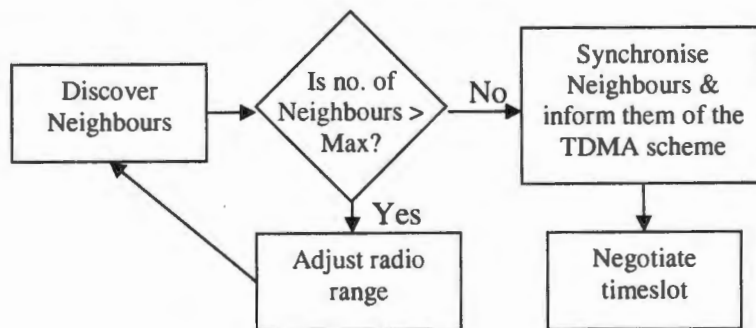


Figure 3.32 Initialisation Overview of Colour TDMA MAC

The following numbered steps explain Figure 3.32:

- 1 Discover the neighbours. Each node identifies all the other nodes within its radio range.
- 2 Adjust radio range if number of neighbours \geq maximum neighbours allowed. The number of neighbours is limited.
- 3 Synchronise with neighbours and let the nodes know what the exact TDMA definition is and when it starts.
- 4 Negotiate the timeslots. Once a node has chosen a timeslot, no other nodes within its radio range or its neighbour's connectivity may choose that same one.

Initialisation is a two part process. It consists of two outward spreading processes:

- Clock synchronisation, neighbour discovery and TDMA cycle information which will be called ‘the synchronisation process’ (points 1 to 3 above)
- Choosing a timeslot according to the colouring algorithm which will be called ‘the timeslot allocation process.’ (covering point 4 above)

The initialisation of Colour TDMA MAC is more energy costly than a contention based MAC.

3.5.2 The Synchronisation Process

The synchronisation process consists of:

- nodes clock synchronising
- discovering their neighbours, and
- being informed of the TDMA cycle information.

This is illustrated by Figures 3.33, 3.34 and 3.35. In Figure 3.33, the central black node clock synchronises all the nodes in its radio range and it gives them the TDMA cycle information. The nodes then start to discover their neighbours through a contention based scheme. Those with too many neighbours reduce them as described in subsection 3.4.2 “Discover Neighbours and Reduce Radio Connectivity if Necessary.” In Figure 3.34, the synchronisation, the TDMA schedule and the neighbour discovery process are moving outwards with certain nodes co-ordinating the process. In Figure 3.35, the process is moving further out.

Nodes that occur in an area of two or more simultaneous synchronisation techniques overlapping one another request that a synchronised node synchronise them, once the simultaneously occurring synchronisation techniques are complete. The request an unsynchronised node makes is by randomly and repeatedly sending out control packets requesting initialisation.

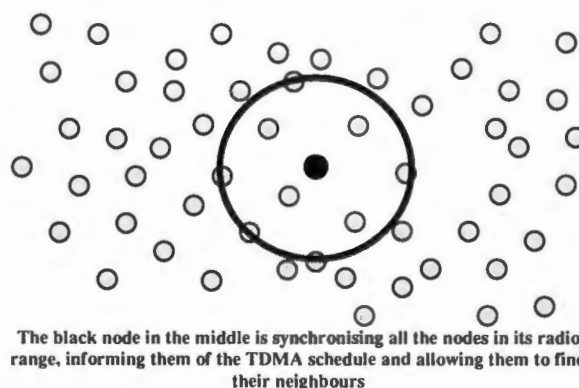
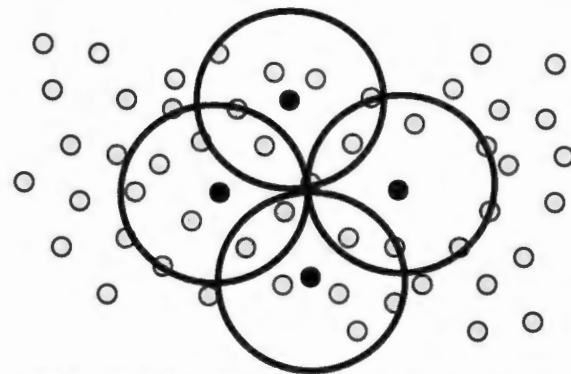
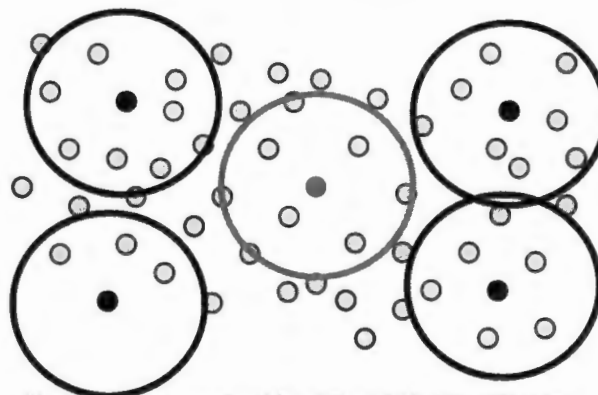


Figure 3.33 Synchronisation, TDMA Schedule and Neighbour Discovery being Propagated Outwards



The black nodes are synchronising all the nodes in their radio range, informing them of the TDMA schedule and allowing them to find their neighbours

Figure 3.34 Later Stages of Synchronisation, TDMA Schedule and Neighbour Discovery being Propagated Outwards



The black nodes are synchronising all the nodes in their radio range, informing them of the TDMA schedule and allowing them to find their neighbours. The orange node is starting the colouring algorithm with its neighbours.

Figure 3.35 Synchronisation, TDMA Schedule and Neighbour Discovery being Propagated Outwards as well as the Colouring Algorithm taking place

Unsynchronised nodes follow their own 'out-of-sync' duty cycle, turning on at the beginning of their awake period, and listening and waiting for a node to synchronise them. If nothing is heard, they go back to sleep until their next awake cycle.

One node starts the chain process of initialisation. That can be either the information sink or the first node to count down a large random number. The information sink has the priority in the initialisation scheme. The information sink is the better option for synchronisation as it will have a better idea what the closest to optimal number of timeslots will be, as well as their length, and so on.

All neighbour nodes are synchronised to its clock; after the clock synchronisation signal, the TDMA cycle information is sent out. It consists of all the numbered points

in 3.3.9. “Network’s TDMA cycle.” Upon nodes receiving this data, they calculate the TDMA scheme from the information provided.

With respect to the synchronisation process, the maximum amount of time allowed for it over an area of nodes is β duty cycles.

3.5.3 Discover Neighbours and Reduce Radio Connectivity where Necessary

The nodes have a few duty cycles during which they can discover their neighbours through control messages. If they have more than the maximum α neighbours, then they reduce radio connectivity. This creates a ripple change – if two nodes reduce their range in the same cycle, this can lead to disconnection.

Nodes have β duty cycles to contend and discover neighbours, rediscover them and change until they satisfy the maximum allowed or fewer neighbours. If they require more time, they may go up to 3β duty cycles which is explained in subsection 3.5.4. The nodes do not reduce their transmission power by half because it then reduces the range by four instead they reduce their sending power by 5%–10%. If the microcontroller is unable to reduce the number of neighbours, then some of those nodes are required to go in to a long power-down state; any better solution is left to future work. The nodes selected to go to sleep must do so either randomly or deterministically. A deterministic method ensures a “bridge node”, as in Figure 3.31, is not put to sleep. Such a method would mean that the node with the most populated neighbour table must not go to sleep.

Figure 3.36 shows one example of what could happen where the black centre is a node that is attempting to reduce its neighbours but cannot. The outside blue circles are black’s neighbours

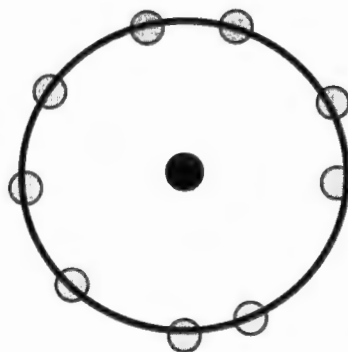


Figure 3.36 Number of Neighbours Can Not be Reduced

This concludes the synchronisation process – the first of the two outward spreading processes. The synchronisation process ensures that

- nodes are clock synchronised
- they have discovered their neighbours, and they do not have more than α
- they know the TDMA cycle information.

After the synchronisation process, a node counts down 3β duty cycles and then starts the colouring algorithm as discussed in the next section.

3.5.4 The Timeslot Allocation Process

Once the synchronisation process has prepared the nodes to choose a timeslot, the colouring algorithm is required so that they can attain one. The colouring algorithm is the timeslot allocation process, which is the second outward spreading process. It runs concurrently to the synchronisation process.

The colouring algorithm begins after a time η . η is a fixed amount of duty cycles after the synchronisation process occurs and it is shown by the mathematical equation 3.3.

$$\eta = (\text{scalar}) \times (\beta) \quad (3.3)$$

β is the maximum amount of time the outward spreading process of synchronisation takes over a group of nodes.

Thus in Figures 3.33, 3.34 and 3.35, the maximum amount of time the synchronisation process takes is β for each figure. In Figure 3.35, if the colouring algorithm were to start at the central orange node, it would not interfere with the outlying black nodes doing the synchronisation process. The colouring algorithm requires β timeslots to colour in the nodes. Thus, a safe minimum value for the scalar would be three, where the two processes would not overlap as shown by the Figures 3.37, 3.38, 3.39 below. Both the synchronisation process and the timeslot allocation process use β duty cycles. This ensures they will never meet.

η is also the number of duty cycles until nodes assume initialisation is over in their area. η duty cycles after the colouring algorithm begins for a group of nodes, they assume initialisation is complete. This is shown by Figures 3.37, 3.38, and 3.39, where the nodes within a dashed large circle have timeslots assigned to them.

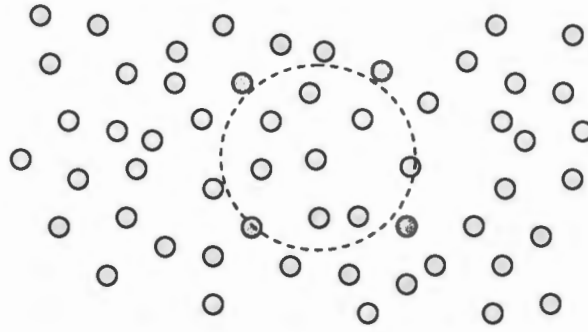


Figure 3.37 Nodes have Timeslots over One-Hop in β Duty Cycles

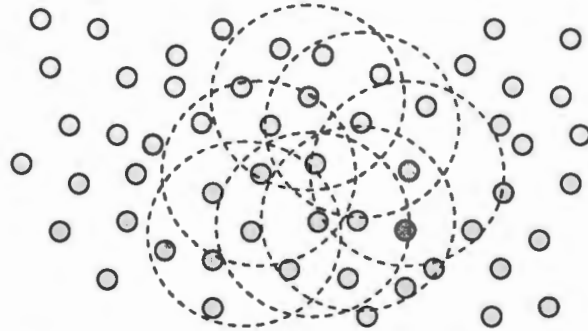


Figure 3.38 Nodes have Timeslots over Two-Hops in 2β Duty Cycles

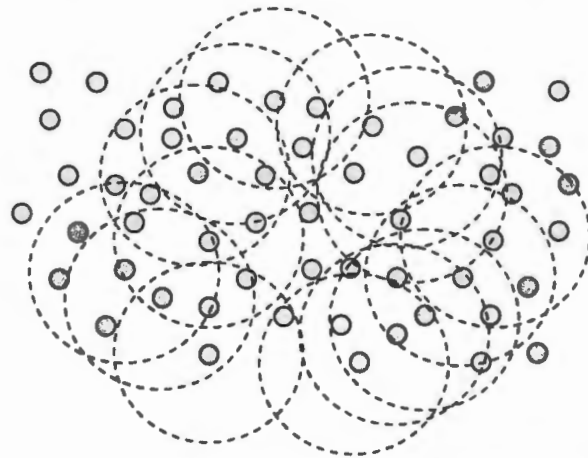


Figure 3.39 Nodes have Timeslots over Three-hops in 3β Duty Cycles

Even if synchronisation is happening on the outskirts in Figure 3.35, it does not affect the inner timeslot allocation. The colouring scheme algorithm takes β duty cycles to assign nodes timeslots. To double that area means all two-hop neighbours are coloured (have a timeslot) as Figure 3.38 shows. To triple it which is illustrated in Figure 3.39 means the algorithm has finished somewhere outside the two-hop neighbour range. Hence 3β , a value that has been used many times in the previous sections.

Nodes that have been through both the processes of synchronisation and timeslot allocation every duty cycle send out a control message at the beginning of their timeslot. This is so that neighbouring nodes know which timeslots are occupied.

Once initialisation is over, the nodes become part of the scheme. The network then moves into the operation stage, in which it remains until the network ends.

3.6 Scheme Operation

In this phase the MAC scheme follows the schedule illustrated by Figure 3.24 and explained in Sections 3.3 “The Scheduling Algorithm” and 3.4 “Synchronisation and Neighbour Discovery.”

There are various scenarios which could occur which upset the operation of the network. An example of this would be if an AHWSN was in a building with a door between nodes with all the nodes in continual execution stage and the door opened revealing a node on the outside, which previously could not be seen, with a radio overlapping the one on the desk. Another example would be if the AHWSN were outside in a forest and it began to rain thereby reducing the radio range. The mechanisms described in Section 3.3 would have to deal with these changes.

The designer must decide on the exact value of α , i.e. the maximum number of neighbours. A maximum of six neighbours is what another MAC scheme which uses colouring suggests [16].

3.7. The PicoRadio

Colour TDMA MAC has been proposed for energy savings in the MAC layer. There is another concept that could be used to further conserve energy and that is the PicoRadio. It is an additional hardware device and it does not fall under the MAC layer but under the physical layer. The Colour TDMA MAC proposal works without it, but incorporating it would reduce energy expenditure.

The PicoRadio is an external low power radio that each node possesses. It works on the principle that if it hears any signal, in the transmission frequency, it awakens the microcontroller attached to it. Thus, if channel user one is awake, while all its neighbours are asleep with their PicoRadios on and their transmitters off, and it sends a message, then all neighbouring nodes awaken.

The reason for using the PicoRadio is that it is more energy efficient than the node's microcontroller and transmitter waking up, checking the channel and going back to sleep if there is nothing to receive. A PicoRadio only awakens the microcontroller

(and hence the radio transmitter) if it is necessary, such as when the channel user has data to send to that sleeping neighbour. The PicoRadio generally uses about 10% or less of the energy than the transmitter uses.

The next chapter investigates the PicoRadio further.

3.8 Advantages and Disadvantages

This scheme is good for unicast traffic where the destination is known and there is only one intended node, but it is less advisable for broadcast traffic. Whether or not this protocol is advantageous depends on the routing scheme and whether it is unicast or broadcast.

Channel usage is deemed inefficient for networks that are not uniformly distributed. An example of this is shown in Figure 3.19, where many timeslots are allocated but only a few nodes actually require timeslots. Also, channel usage is low where nodes have nothing to send. However, part of the colouring algorithm requires that there be unused timeslots which neighbours may use. So, to eliminate internal collisions, channel usage is traded off.

The scheme is slightly complex as far as initialisation is concerned, and it could do with some simplification. A complex scheme is undesirable because a microprocessor has limited resources and it is required to place a whole telecommunications stack in its flash memory, which might only be 4kB⁵. Once the scheme is simplified and the network is established, Colour TDMA MAC provides an energy efficient means of relaying data as the results show.

Colour TDMA MAC is designed for non-mobile AHWSN but new non-mobile nodes may be added to the network at any time. If nodes move they cause internal collisions.

⁵ As in the case of the Nordic Semiconductor nRF9E5.

4 PICORADIO AND MAILBOX

In chapter 3, the proposal uses a PicoRadio to awake a sleeping node. The PicoRadio is not a part of the MAC layer and it is special piece of hardware that does not come standard with the microcontroller. With regard to the International Standards Organisation Open Standards Interface Model, it falls under the physical layer. This proposal is for the MAC layer in AHWSN, but it requires some hardware in the physical layer to assist the scheme. Hence, it is necessary to investigate the physical layer (and thus the PicoRadio).

4.1 Physical Layer Considerations

The following two sections 4.1.1 “Wireless Data Transmission” and 4.1.2 “Transmitter, preamble, modulation, and encoding” are an introduction to wireless data transmission in the physical layer. They explain why Colour TDMA MAC does not require synchronous⁶ wireless data transmission, provided that the receiver’s radio is listening to the channel when the sender is transmitting.

4.1.1 Wireless Data Transmission

If node A has sensor data to send to its nearby (in radio range) neighbour node B, telecommunication between the two follows the steps shown in Figure 4.1 below. Node A knows node B’s address.

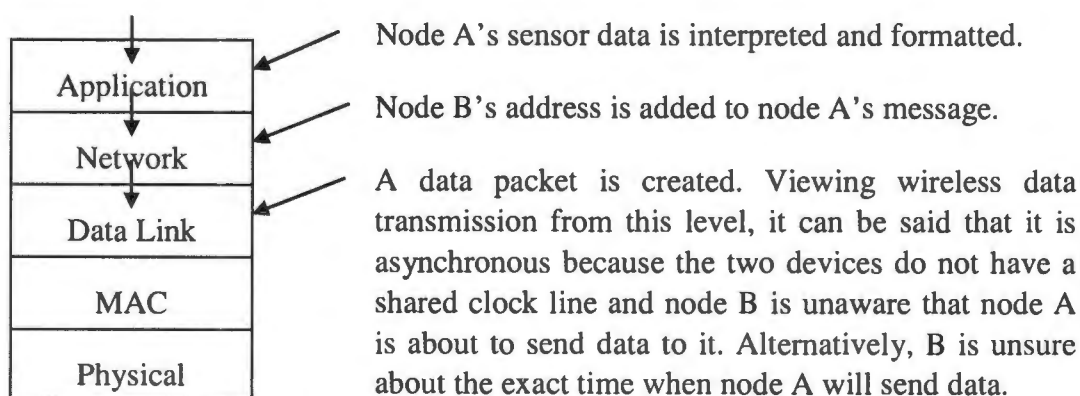


Figure 4.1 Simplified Network Stack Model

⁶ One master synchronising the two nodes that will communicate.

As the process is investigated further, it is shown that:

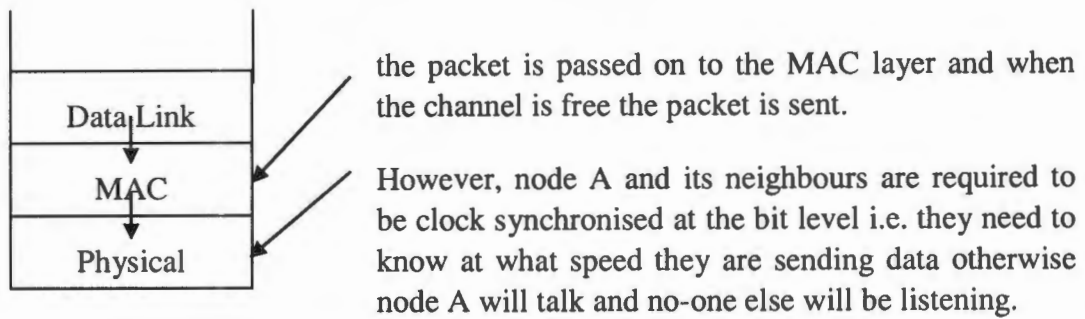


Figure 4.2 Lower Section of the Simplified Networking Stack Model

Hence, even if the data transmission “seems” asynchronous, the devices still need synchronisation at the bit level.

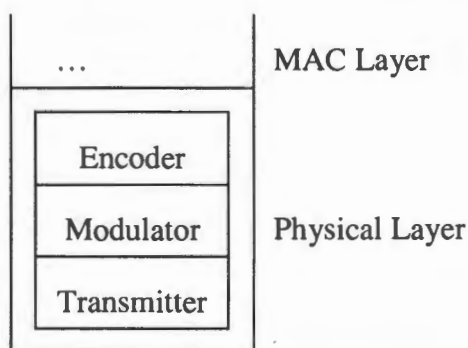
Some MAC protocols [11] use the cluster head to co-ordinate synchronous wireless data transmission. The cluster head is the master (like a base-station) and in turn it maintains the clocks and slots of any two nodes in its cluster (the children) that want to communicate with each other. Colour TDMA MAC does not use cluster heads.

In order to do a bit synchronisation, most wireless AHWSN systems use a phase lock loop (PLL) or an envelope detector as a receiver and the following subsection explains how they work.

4.1.2 Transmitter, Preamble, Modulation, and Encoding

At the beginning of each transmission, nodes send a preamble (a special bit pattern) that tells a PLL where the bit transitions are, so that sampling may occur at the correct time. Or, with an envelope detector, a data slicer (hardware or software) is used to align bits and distinguish a 'one', a 'zero', and two 'ones' or 'zeros' in a row.

“A physical layer transmission stack” (Figure 4.3) is thus created within the physical layer stack to help illustrate the signal that would be sent over the medium.



An incoming frame from the MAC layer enters the Physical layer. It consists of a number of binary digits.

The encoder groups binary bits into symbols. The modulator formats those symbols into transitions that the transmitter must perform when sending the message and a preamble is sent at the beginning of the message.

Figure 4.3 Physical Layer Transmission Stack

Nodes A and B could have clocks that are different i.e. the transmitter of Node A and receiver of Node B could be running at slightly different rates which would cause a drift across them. In this proposal and in the assumptions, packet sizes are limited and are small, only 32 bytes. It is also assumed that the short packet is not long enough to be affected by the clock drift.

Colour TDMA MAC relies on the above technique to transmit between nodes. Part of the reason why the Nordic nRF9E5 microcontroller was chosen in the results section is because it takes care of all these details.

4.2 The PicoRadio

The PicoRadio is an external low power radio that awakens a sleeping microcontroller when a signal is detected in the channel. This proposal discusses PicoRadios but with a timer that enables the radio and disables it. Figure 4.4 is a general design for a PicoRadio.

4.2.1. A General Design of a PicoRadio

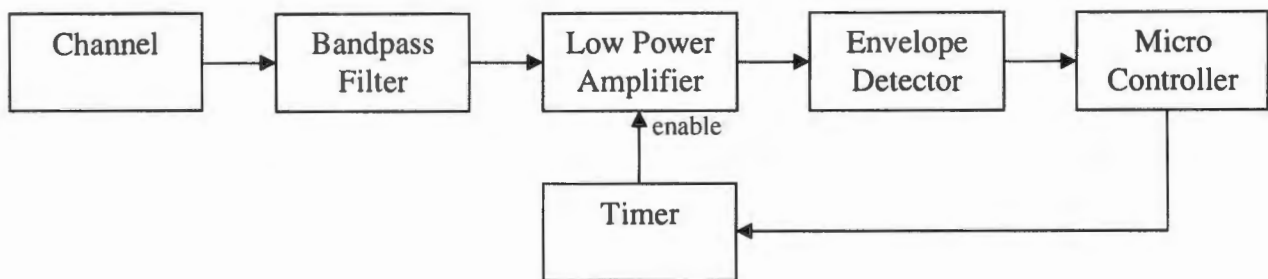


Figure 4.4 Diagram of a PicoRadio and Microcontroller With Integrated Timer

At the beginning of the network, after initialisation, the microcontroller sets the timer for when it must enable the low power amplifier and for when it must disable it. A signal in the sender's frequency range passes through the bandpass filter and enters the low power amplifier. If the amplifier is enabled, it boosts the signal and passes it on to the envelope detector, which detects the presence of a signal and sends an interrupt to the sleeping microcontroller.

The envelope detector has a diode that rectifies the incoming signal, so that it is only positive. An integrator then sums up all these positive curves and if there is more energy than the threshold limit, the microcontroller is triggered

If the Low power Amplifier is disabled, the PicoRadio (being the whole system) will not wake up the microcontroller.

The PicoRadio assists the node by not requiring it to keep its radio transmitter constantly in receive mode i.e. awaiting a signal. The node can go to sleep and rely on the PicoRadio to awaken it. This is a more energy efficient solution, but the best solution would be if the PicoRadio could freeze the incoming signal and give it to the microcontroller when it awakens in its timeslot instead of awakening it immediately to receive a message then going back to sleep. The proposed mailbox is a concept for holding an incoming signal until the microprocessor awakens.

Figure 4.4 illustrates that the component layout suffers from energy inefficiency. For instance, starting with the first device mentioned, the bandpass filter, if the operating frequency is 433MHz then there is a need for a very high Q factor (to eradicate all other frequencies). A passive filter is unable to provide such a high Q factor without consuming large amounts of power. There are two advances in filter technology that would be applicable for low power, low cost and a high Q factor. They are:

- the surface acoustic wave (SAW) filter
- MEMS.

SAW filters have a Q of about 2000 and it is available technology, whereas MEMS has a Q of at least 10000 or more, but it is still in its experimental stages and is not readily available.

4.2.2 A MEMS PicoRadio

The concept of the PicoRadio is not to receive data but to wake the microcontroller when it hears a signal in the channel. The Author believes that PicoRadio could be made just from a MEMS filter and an amplifier, a very narrow bandpass filter and then a signal strong enough to awaken the microprocessor. This is future work but the idea has been conceived.

4.2.3 A Feasible PicoRadio for AHWSN

The concept of a PicoRadio is that it is an external low power radio that awakens the microprocessor and the transmitter when there is data to be received. Its requirements are that it must be more energy efficient than the device it is connected to. AHWSN being concerned about energy conservation tend to use the latest low power radios on the market. Thus, it is difficult to find another radio that uses less power than the AHWSN node and some consider the PicoRadio to be just a concept rather than something that is available. The proposal uses the Nordic Semiconductors nRF9E5 as a reference to what would be placed in a node of an AHWSN. These are cheap microcontrollers with a very low power radio. After searching, however, a device was found that would fit as a PicoRadio to the nRF9E5. The device is [28] "A 0.5V 3.1mW Fully Monolithic OOK Receiver for Wireless Local Area Sensor Network."

The title mentions the highlights of the receiver but if the 0.5V is lowered to 0.45V, the energy used is 1.36mW at -35dBm. The improvement in energy savings compared to the nRF9E5 is a factor of about ten. The downside is that -35dBm is not far transmission distance for the AHWSN if it is sending at 27mW (3V*9mA). However,

the PicoRadio only needs to awaken the microcontroller, hence a signal containing one bit of information should be all the PicoRadio requires to wake up the node. Thus, if one bit was sent at a high power, it could elevate the problem of distance at -35dBm. This is just an illustration of what could be done and thus that the PicoRadio is not just a theoretical idea. Chapter 5 “Results” does not implement [28] but uses a theoretical PicoRadio.

4.3 The mailbox

The Mailbox is a response to the problems experienced with PicoRadios, in particular, the PicoRadio's inability to hold back the incoming signal until the microcontroller wakes up. Waking up the microcontroller when it is asleep is energy wasting hence the idea of introducing a theoretical design for a mailbox which would eliminate the problem altogether. This idea was presented at an international conference held in California, USA in January 2006. (The conference paper is included at the after the appendix).

The transmission of data from one node to another would, in an ideal system, follow the route as depicted in Figure 4.5:



Figure 4.5 Mailbox Position in the System

This received signal is required to be digitalised and demodulated, or demodulated and digitalised, for the microprocessor to interpret the message. Figure 4.6 shows the different choices for dealing with the signal.

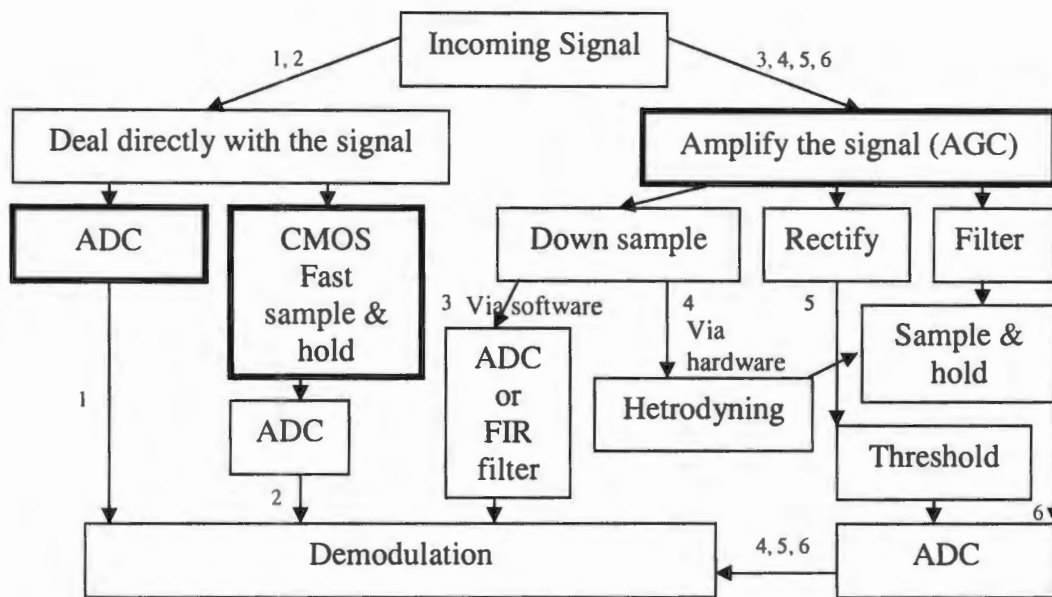


Figure 4.6 Options for Dealing with the Incoming Signal

The power hungry devices have a bold outline. There is no path that leads through Figure 4.6 that does not go through at least one “bolded item”. If there were, then that would be the choice to take over the built-in receiver in the microcontroller of the AHWSN node.

Below follows an investigation into the paths in Figure 4.6 labelled 1, 2, 3, 4, 5 and 6.

- 1 Dealing with the signal directly using an Analogue to Digital Converter (ADC), a 12bit ADC obtains a 70dB Signal to Noise (S/N) ratio⁷. The incoming signal could easily have a greater S/N ratio, thus either a higher bit ADC is required, or the maximum and minimum of the signal need to set the maximum and minimum of the ADC. Even if this is solved there is still the problem that power dissipated by an ADC is proportional to clock speed squared. The chip would therefore be sending a signal at one of the following frequencies⁸: 433MHz, 868MHz, or 915MHz.
- 2 Complementary Metal Oxide Semiconductor (CMOS) technology is based on Field Effect Transistors (FET) technology. FETs can sample at low power at very high frequencies, but the holding of the sample value is where the largest amount of current is drawn. The “hold part” of the circuit is a FET being used as a capacitor. It charges up according to: $i = dv/dt$. If the receiver is near the sender, then the received power will be large, hence the “dv” will be

⁷ The S/N ratio that would be appropriate would be 110dB or higher which is not available. Aug 2006.

⁸ Based on the Nordic Semiconductors nRF9E5 microprocessor and these falls within the ISM band.

considerable while the “dt” will be determined by one of the sending frequencies and be very small and “i” will be large. Another set back is that available CMOS may do sampling up to 2G samples. If the nodes are using 900MHz as their transmission frequency, only 8bits can be used, which results in an unacceptable S/N ratio.

- 3 3, 4, 5 and 6 all use amplification before manipulation. The nodes would be spread out which means the incoming signal must be boosted up considerably. The amplifier is an Automatic Gain Control (AGC). This device varies the amplification, providing large current for small signals and less for larger ones. In receiving, close to as much as the sending power will be used to amplify the incoming signal.

After many attempts to build a mailbox, it was decided, due to time constraints, to stop researching it and to use a PicoRadio in the scheme. The time and effort in attempting to build the mailbox is not reflected in this thesis.

If the mailbox were able to be implemented, it would follow the schedule of Colour TDMA MAC where it would only be enabled for a short period of time and would have a time limit of a certain number of packets. This is to enable the storage capacity of the mailbox to remain low.

5 RESULTS

This section begins by explaining the reasons for comparing a contention based MAC to the proposed Colour TDMA MAC. The explanation also describes the difficulties encountered when comparing the two different protocols. After this, S-MAC is investigated individually and then a comparison of S-MAC and Colour TDMA MAC is explored.

5.1. Comparison of Colour TDMA MAC to S-MAC

Comparing any AHWSN telecommunication protocols is difficult, since the traffic the nodes produce must be investigated. This data must be forwarded on to neighbours and terminate at the information sink. However, this process depends on:

- The number of nodes
- The position of each node
- Their density, or how close they are to each other
- Their applications.

In addition, there is an infinite number of possible AHWSN node configurations.

Comparing any AHWSN MAC protocols also requires an investigation of inter-nodal traffic. The MAC only concentrates on relaying messages from a node to a neighbour. The number of nodes in the network and their position are of lesser importance. The density, i.e. the number of neighbours any given node has, and the application of the network affect the MAC protocols more directly.

An AHWSN MAC can either be compared to a contention based or a non-contention based MAC. S-MAC was selected because it is the most implemented MAC protocol in AHWSN, and following points summarise why:

- Its code is small and not resource intensive.
- It comes in a complete package solution.
- It is a stable program.
- It is free for everyone.
- It is “plug ‘n play.”

AHWSN papers use S-MAC. (See section 2.3.1. “Sensor-MAC”). It is downloadable from the Internet as a complete, stable package that is written in the TinyOS platform. When added to a microcontroller, the only additional item that is required is a routing algorithm.

Comparing S-MAC and Colour TDMA MAC means comparing two completely different entities, but it is preferable to compare the proposal to a mainstream solution rather than ten different proposals which are not being implemented in microcontrollers. For the reason that they are different, S-MAC will be investigated individually first and then compared to Colour TDMA MAC.

5.2 Microcontroller and MAC Properties Implemented to Obtain Results

The performance of a MAC protocol in AHWSN is dependant on many factors. One of them is the traffic and another is the microcontroller implemented. Depending which microcontroller is used, different “energy” graphs are obtained. This is because their properties change from manufacturer to manufacturer. The microcontroller used will scale the output energy in graphs, but if both MAC schemes being compared use the same chip, then the microcontroller may be seen as a scaling factor of the output energy. Thus one processor is chosen (Table V “The Nordic nRF9E5 Microcontroller”) and used to find the graphs in the subsections of this chapter 5.

TABLE V

THE NORDIC NRF9E5 [17]

Supply Voltage	3V
Supply current in transmit @ -10dBm output power	9mA
Supply current in transmit @ 10dBm output power	30mA
Supply current in receive mode	12.5mA
Supply current for μ -controller 4MHz @ 3volt	1mA
μ -controller running at	4MHz
Data rate	50KbpS
Sensitivity	-100dBm
Supply current in power down mode	2.5 μ A
Radio Wake up	650 μ S

-100dBm is only for line of sight and because it is desirable to save energy by transmitting at 9mA (-10dBm), the maximum distance the signal covers in theory is

10 metres. If the sensors are in an office environment, this distance changes as doors open and close, or if it rains and the sensors are outside

The microcontroller was chosen because of its availability as well as it being one of the newer, more energy efficient microcontrollers on the market. Electronic stores in South Africa supply a limited range of microprocessors and controllers. They generally only supply what is popular and cheap overseas. The Nordic nRF9E5 is one of the new energy saving microcontrollers, incorporating the latest in technology, that are available in this country . It has a fast radio wake up of 650 μ S, where other microcontrollers may need as much as 3mS to wake up their radio, which is energy consuming. The supply current (9mA) in transmit at -10dBm is energy efficient, even though the distance covered is not far. The data rate is one of the fastest for competing microcontrollers. The current consumption (1mA) at low voltage (3V) and high frequency (4MHz) is very energy efficient.

The nodes in the S-MAC AHWSN employ the parameters shown in Table VI.

TABLE VI
S-MAC PROPERTIES

Node duty cycle	1150mS
10% awake time	115mS
90% asleep time	1035mS
Sensor period	12000mS
Control packet size	10 bytes
Data packet size	32 bytes

The S-MAC contention consists of the following properties:

- 15 Contention windows for sync
- 31 Contention windows for RTS and CTS.

Except for the sensor period and packet size, these parameters stem directly from [1].

5.3 S-MAC Mathematical Contention Analysis

This section investigates the energy lost in contention. As an introduction, a brief overview of the S-MAC scheme is presented.

5.3.1 The S-MAC Schedule

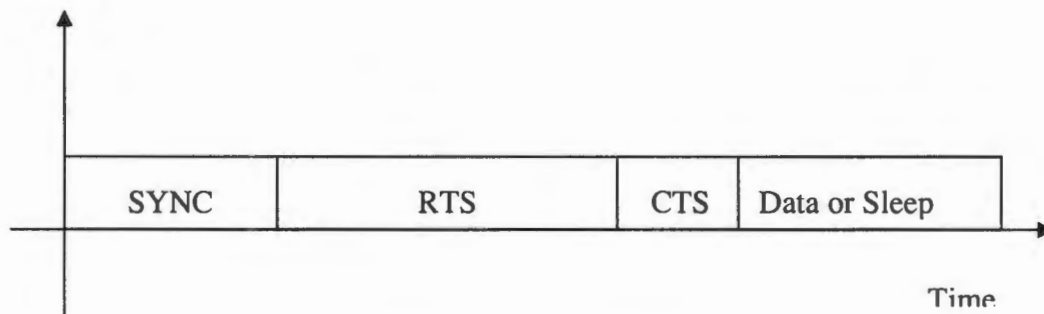


Figure 5.1 S-MAC Schedule

The SYNC period comprises fifteen contention windows. If a node is to send a SYNC message, it chooses one of the slots and sends the message. There are no acknowledgements from any of the neighbours, so the node does not know if the outcome was a success or a failure.

The RTS and CTS section are combined and have thirty-one contention windows for contenders. The nodes send their request to use the channel in the RTS slot. They hopefully receive an acknowledgement (CTS) that they can send the data during the 'data or sleep' stage. If no CTS is received or a node does not receive a RTS intended for itself then it goes to sleep during the 'data and sleep' period.

5.3.2 Contenders

There are thirty-one contention windows for contenders in the RTS and CTS section. Figure 5.2 with a summary below it explains what contenders are:

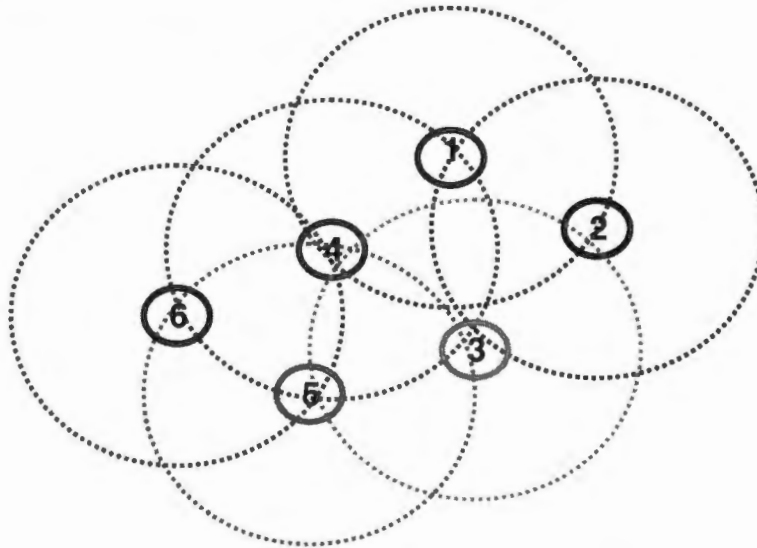


Figure 5.2 Illustrations of Contenders

Example One:

- Node 2 is to send two messages to its neighbours 1 and 3.
- Node 3 is to send one message to its neighbours 2, 4, and 5.
- Node 2 and 3 are to send 'RTS' control packets to each neighbour they have data for. The neighbours are to reply with a 'CTS' control packet. Nodes listening to the channel cannot interpret control packets sent simultaneously.
- Node 2 chooses two different times to send the two RTS messages to 1 and 3. Likewise, node 3 chooses three different times to send its three RTS messages. Node 2's two RTS messages can collide with node 3's RTS messages if they choose to send a RTS at the same time. Thus there are:
 - 2 Nodes in contention
 - 7 Data messages to be sent
 - 10 CONTENTERS (5 RTS, 5 CTS) in node's 2 and 3's radio range
- The number of RTS and CTS messages is the number of contenders.

Example Two:

- 1 node sending 10 Messages becomes 2 contenders because there is one RTS and one CTS.
- The number of RTS and CTS messages is the number of contenders.

5.3.3 Probability of Collisions

Figure 5.3 shows the probability of there being at least one collision among a number of contenders in one duty cycle of an S-MAC scheme. Figures 5.2 and 5.3 are not related.

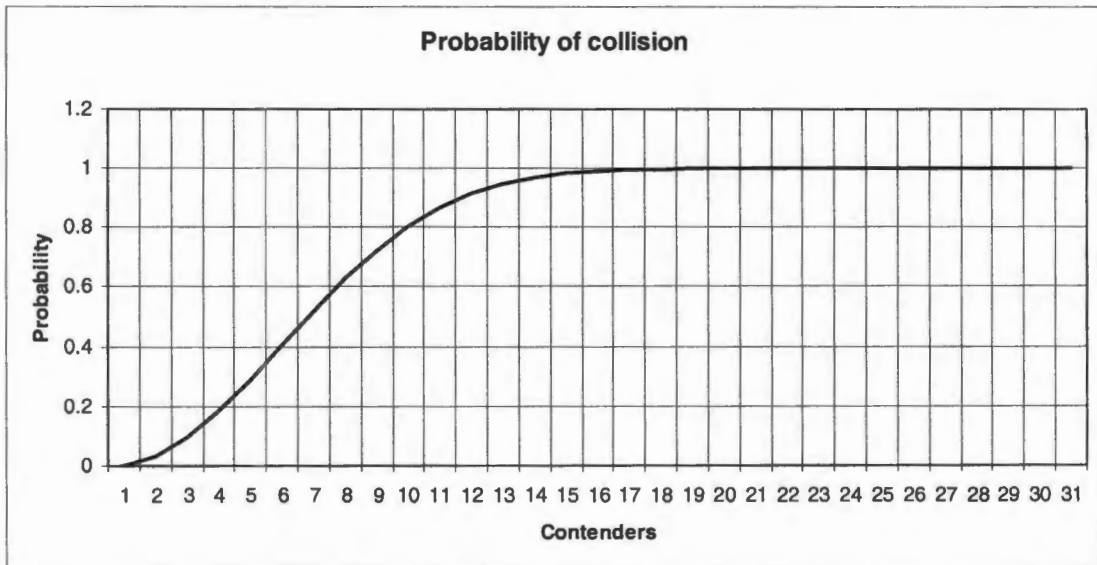


Figure 5.3 Probability of At Least One Collision in the RTS and CTS Section of S-MAC Given a Set Number of Contenders

The graph was calculated using discrete mathematics, which is to be found in the Appendix 9.1.2 “Collisions in Contention and the Birthday Algorithm.” The probability never goes over the value 1.

The graph provides the probability of at least one collision but energy lost in conflict is different depending on the number of nodes affected by the collision. Three nodes requiring packet resubmission is more energy consuming than two nodes requiring packet resubmission. The following graphs, Figures 5.4 to 5.9, show the probabilities of exactly one collision, exactly two collisions, exactly three collisions, etc. They have all been calculated using the combination mathematics in Appendix 9.1.3 “Combinations of the Birthday Algorithm.”

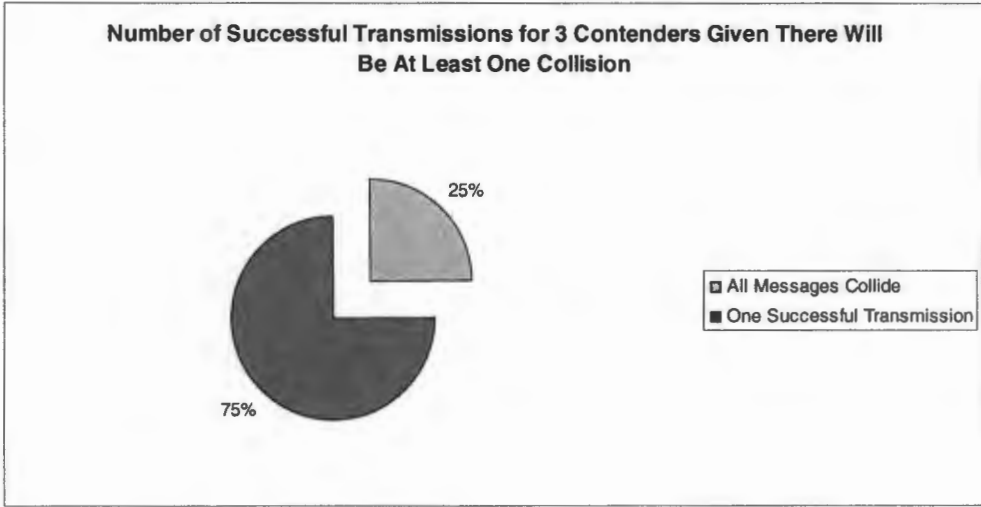


Figure 5.4 Number of Successful Transmissions for 3 Contenders Given There Will Be At Least One Collision

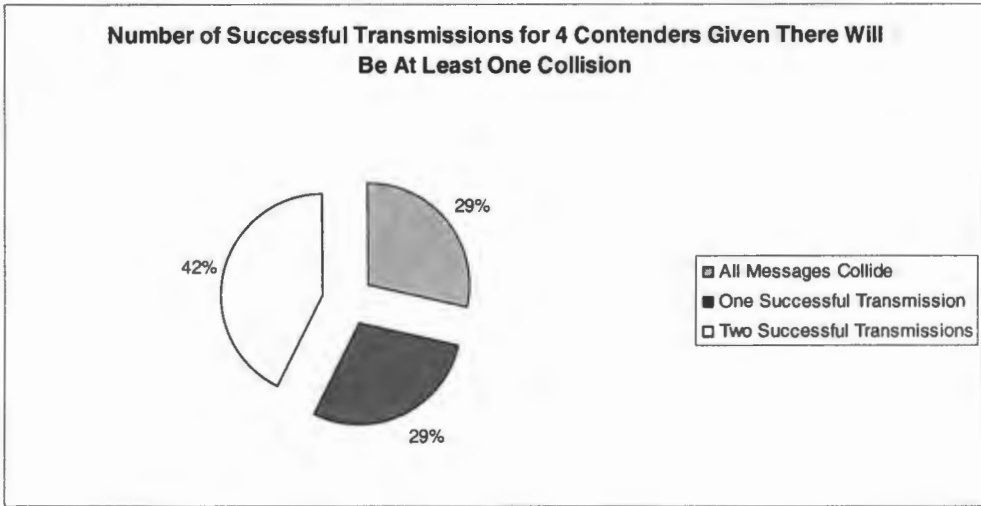


Figure 5.5 Number of Successful Transmissions for 4 Contenders Given There Will Be At Least One Collision

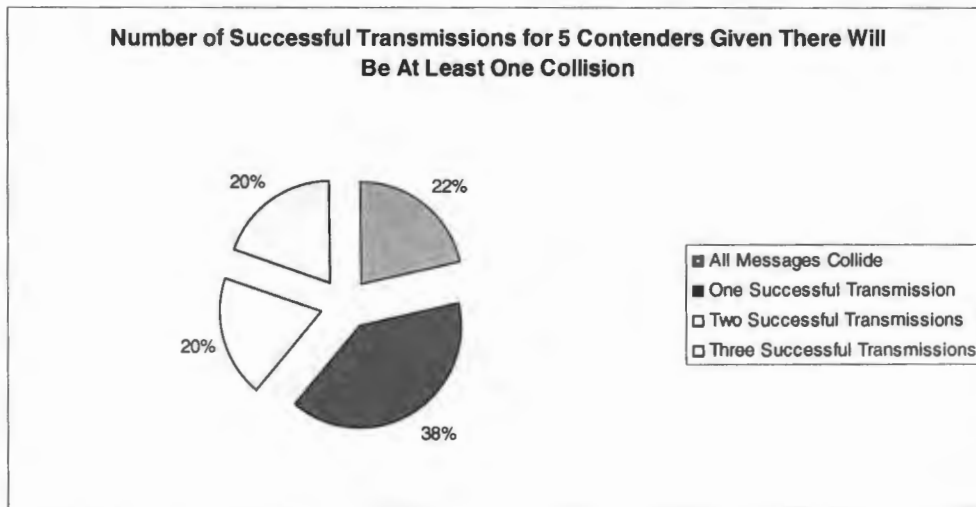


Figure 5.6 Number of Successful Transmissions for 5 Contenders Given There Will Be At Least One Collision

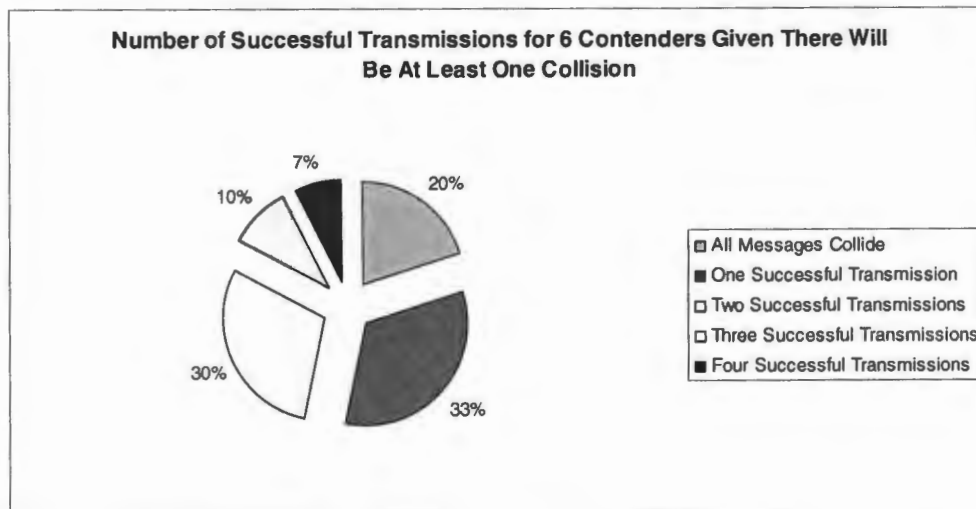


Figure 5.7 Number of Successful Transmissions for 6 Contenders Given There Will Be At Least One Collision

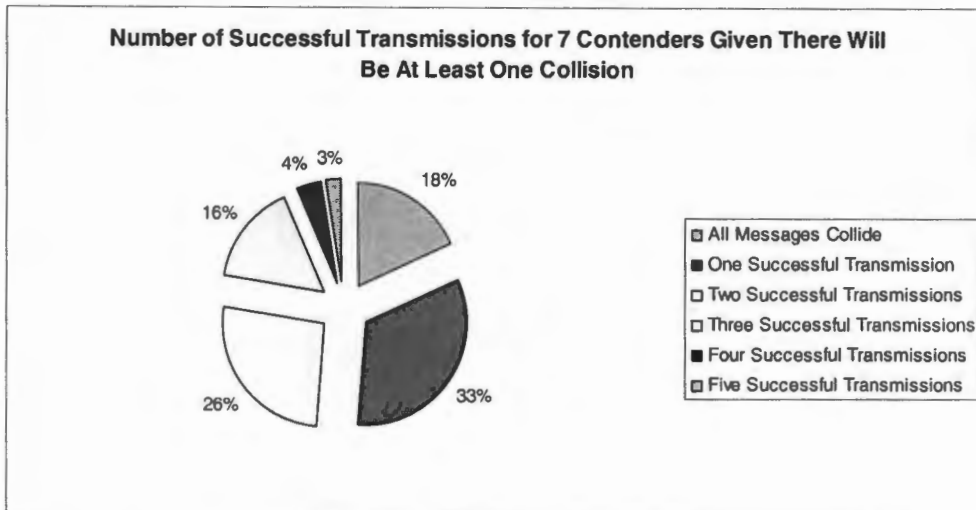


Figure 5.8 Number of Successful Transmissions for 7 Contenders Given There Will Be At Least One Collision

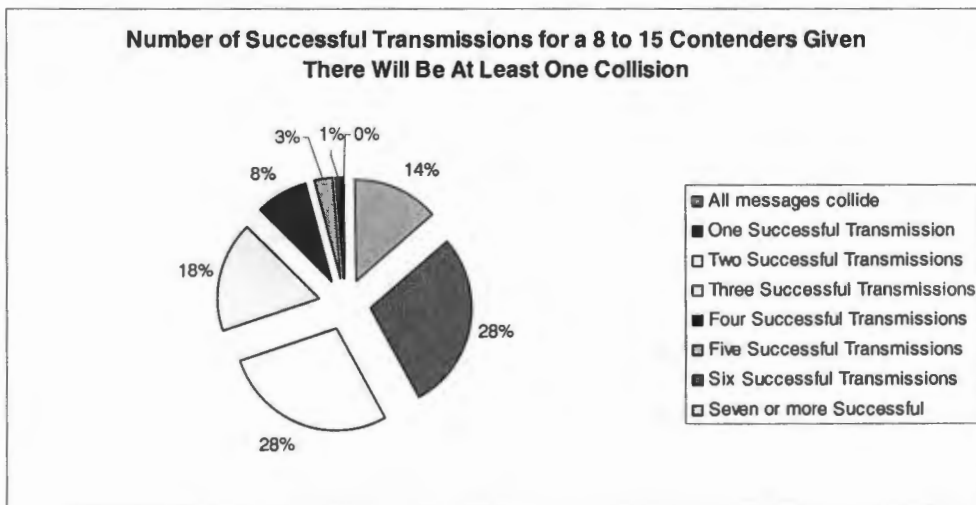


Figure 5.9 Number of Successful Transmissions for 8 to 15 Contenders Given There Will Be At Least One Collision

The graphs show that if there are five contenders and there is a collision, then there is an 80% chance that zero, one, or two transmissions will be successful. However, when there are between eight and fifteen contenders, it is a 74.5% chance that zero, one or two transmissions will be successful.

If there are one or more collisions in this round, then the same messages must be re-sent in the next duty cycle. The probability of a collision in this round affects the probability of collision in the next round. S-MAC [1] does not mention how it deals with reducing that probability in the original paper but the S-MAC code is constantly updated. In their TinyOS S-MAC implementation, the S-MAC code allows for back-off algorithms to be implemented. Using an algorithm such as the binary exponential

back-off, the change of probability in the next round can be assumed to be small and thus it is possible to use a simple Markov Chain (Figure 5.10) to obtain Table VII.

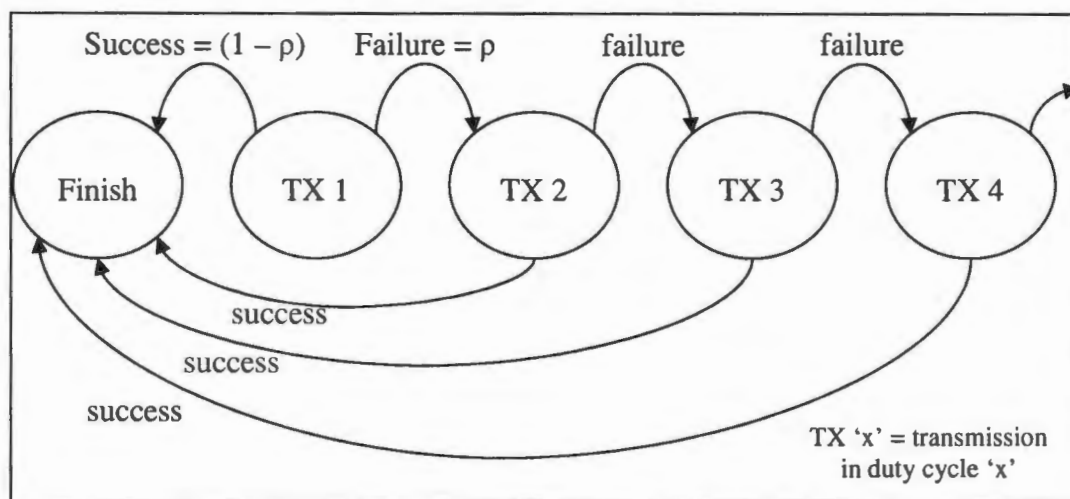


Figure 5.10 Markov Chain to Find the Number of Retries until Success

TABLE VII

NUMBER OF NODES IN CONTENTION AND THEIR ASSOCIATED NUMBER OF ADDITIONAL DUTY CYCLES UNTIL MESSAGE IS SUCCESSFUL

No. of Nodes in Contention	No. of Additional Duty Cycles Until the Message is Successful	Probability of collision in 1 duty cycle for the given number of contenders
2	1	0.032258064
3	1	0.094693028
4	2	0.18230338
5	4	0.287812621
6	6.5 rounded up to 7	0.402681553

Therefore, in S-MAC there is great energy wastage due to collisions and resubmissions. Colour TDMA MAC does not suffer from internal collisions, so theoretically all the energy wasted by collisions in S-MAC is saved by Colour TDMA MAC. Colour TDMA MAC also does not suffer from the control overhead messages (RTS and CTS). Colour TDMA MAC does, however, suffer from low channel efficiency, whereas with a large number of contenders S-MAC's channel efficiency is low as shown by the results above.

5.4 Colour TDMA MAC Properties and the Schedule

The length of the timeslots, guard times and other factors of the scheme used to obtain results are presented in this section. Figure 3.24 is the scheme and the two Figures 5.11 and 5.12 discussed below are extracts from it.

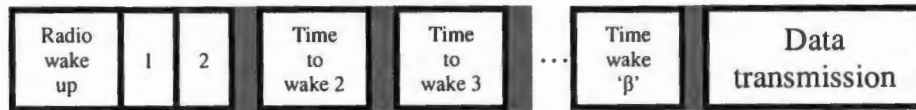


Figure 5.11 One Timeslot of Colour TDMA MAC

What follows is an investigation of Figure 5.11, block by block from left to right:

- Radio wake up – This is the amount of time a radio requires wake up. The precise time is 650uS (for the Nordic nRF9E5, Table V). Radio wake up does not necessarily happen at that block. If a control packet is to be sent in 1 or 2 then the radio wakes up in the first block. If the radio is only required to wake up channel user four (and nothing before) then the radio may wake up in the guard time between blocks “Time to wake 3” and “Time to wake 4.”
- 1 and 2 are of equal length – Each one should accommodate one control packet. Thus 1 and 2 are each 1.6mS as per the discussion in the next point.
- The guard times depend on the length of time it takes to transmit the control packets in the blocks “Time to wake ‘n’” ($2 \leq n \leq \beta$). A control packet is ten bytes long, so it is assumed that that ten bytes includes preamble, addresses and the cyclic redundancy check. Ten bytes divided by fifty thousand bits per second is 1.6mS.
- The “Time to wake ‘n’” should be 1.6mS as per the discussion in the previous point.
- Data transmission should be able to accommodate the node sending its neighbour information table which [9] sets at a maximum of five hundred and twelve bytes. This takes 82mS to send.
 - The duration of Figure 5.11 is: $(85.85 + 3.2*\beta)$ mS
 - The 85.85mS is the radio wake up, ‘1’, ‘2’, and data transmission lengths.
 - $3.2*\beta$ is the 1.6mS Guard time plus 1.6mS time to wake a certain neighbour up.

Figure 5.11 is for one timeslot in the TDMA scheme Figure 3.24. There are β timeslots in Figure 3.24. Thus the length of the TDMA schedule of Figure 3.24 excluding the special events slot is: $((85.85 + 3.2*\beta)* \beta)$ mS.

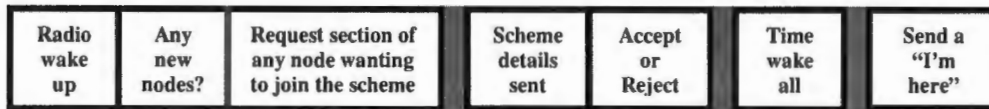


Figure 5.12 The Special Events Timeslot of Colour TDMA MAC

What follows in an investigation of Figure 5.12, block by block from left to right:

- Radio wake up is 650µS long.
- Sending out a request to node who would like to join the scheme is broadcasting a control packet, which is ten bytes thus the block "Any new nodes?" is allocated 1.6mS.
- The contention for the new nodes wanting to join is estimated at 32mS.
- The Guard time is 1.6mS (as per the guard times discussed above).
- Scheme details sent has a maximum of sixty-four bytes, thus it requires 5.12mS but that is grouped with the "Accept or Reject" block, which is a control packet of 1.6mS. Therefore 6.72mS is provided for both blocks.
- "Time wake all" is set at 1.6mS.
- "I'm here" is also a control message hence 1.6mS is given to that slot.

The overall time of the special events timeslot is: 48.97mS. Thus the overall TDMA time of Figure 3.24 is

$$(((85.85 + 3.2 * \beta) * \beta) + 48.97) \text{mS}$$

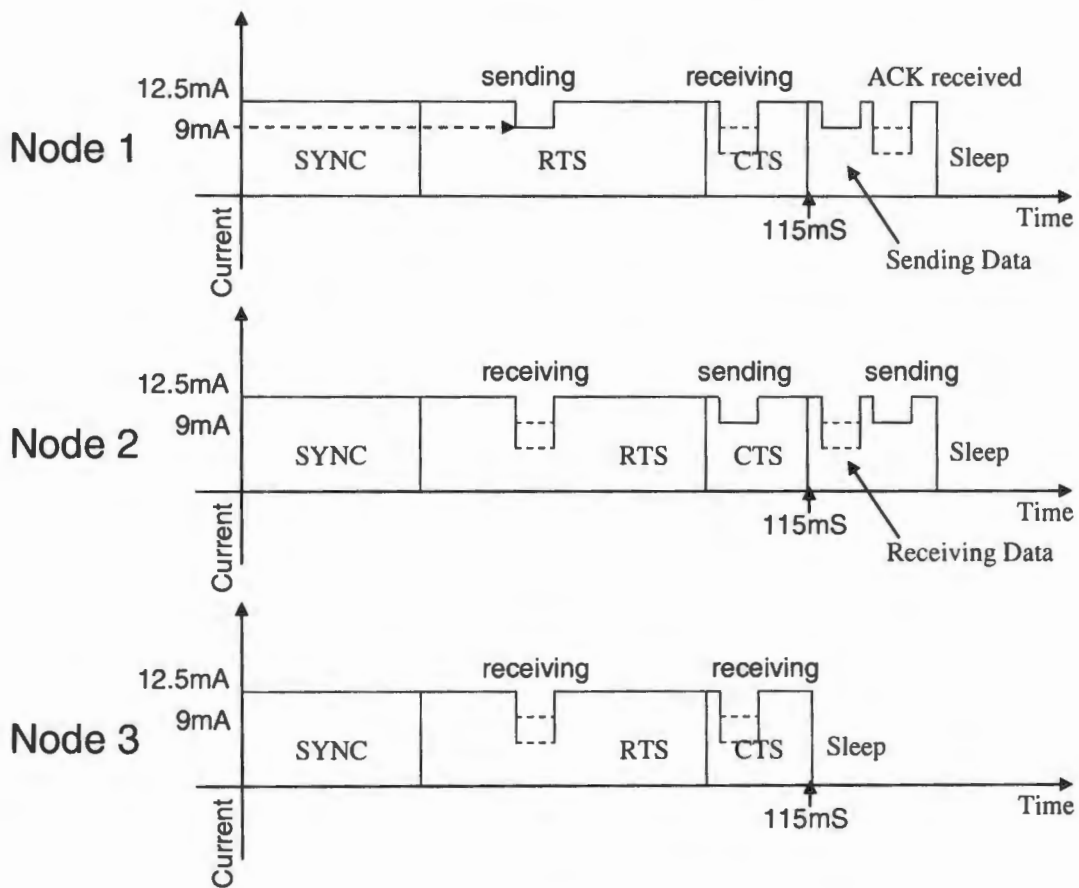
These figures provide the time allocations for the Colour TDMA MAC using the nRF9E5 microprocessor. The sections that follow use these numbers.

5.5 S-MAC versus Colour TDMA MAC

5.5.1 One Example of S-MAC versus Colour TDMA

In the simple scenario of three nodes, with node one sending a data message to node two and the graphs are plotted for the radio usage of S-MAC (Figure 5.13) and the radio usage for Colour TDMA MAC (Figure 5.14). Comparing these graphs gives a value for how much more energy efficient one scheme is compared to the other, with respect to radio utilisation. The transmitter is the biggest energy consumer as shown in Table V, where the radio transmitter may be using 12.5mA whereas the microcontroller uses 1mA only.

The sending current is 9mA at -10dBm, and the receiving current adjusts according to how much amplification is required: the further the node, the greater the amplification that is required. In the diagrams, it is assumed that the receiving signal current consumption is 9mA for both schemes, so that distance may be ignored.



**Figure 5.13 S-MAC Three Nodes Energy Utilisation when
Node 1 is Sending Data to Node 2**

The energy used in idle listening by node three is clearly a waste as shown in Figure 5.13. As long as the radio is in receive mode, it uses the maximum current because it assumes the worst case, where the signal is the smallest and maximum amplification is required.

Another issue is cross talk. Even though it is not illustrated in Figure 5.13, if there are any collisions, it actually reduces the power consumed by idle listening. While idle listening, the radio sits at 12.5mA but when it receives something whether it be a valid data message or junk (due to collisions) the transmitter uses 9mA.

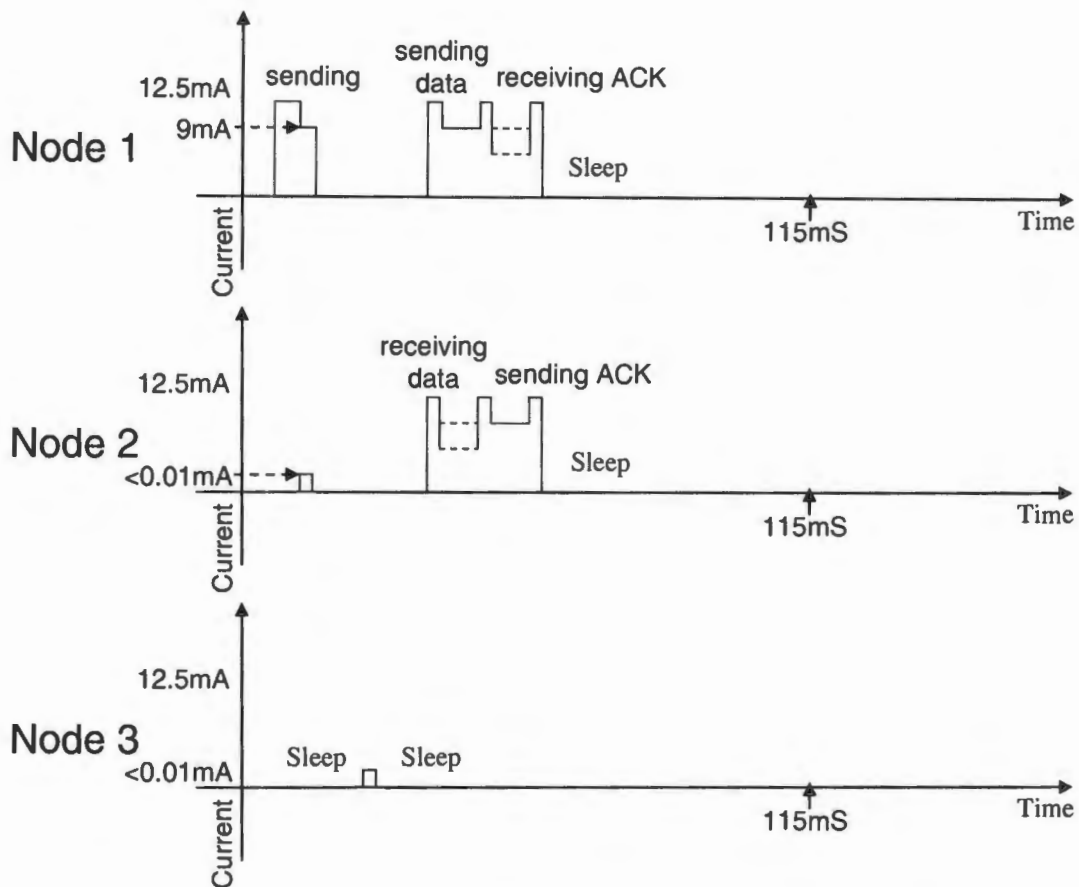


Figure 5.14 Colour TDMA MAC Three Nodes Energy Utilisation when Node 1 is Sending Data to Node 2

From the Figures 5.13, 5.14 and the data in Tables V and VI, it transpires that the energy consumed in Figure 5.13 is 13.21848mJ⁹, whereas in Figure 5.14 only 0.4149mJ¹⁰ is consumed. In the latter case, the length from the beginning of a timeslot to the point it sends data is 32mS and the time period during which a PicoRadio is required to be enabled is 2mS.

The increase in power savings in the radio transmitter is 314%. This is only one figure for one particular situation, however, an all encompassing number is desired.

5.5.2 Simulations

A year and a half was spent working on, and coding simulators specifically to compare these two MAC protocols. What became evident was that simulations were not the best measure for AHWSN MAC protocols. Either that or they proved to be too difficult to install and run on a system as the same network setup would provide

⁹ Total = Sending (Node 1) + Receiving (Node 2) + Idle Node (Node 3).

Sending = $3 \cdot 0.0125 \cdot 0.115 - 2 \cdot 3 \cdot (0.0125 - 0.009) \cdot ((10 \cdot 8) / 50000) + 3 \cdot 0.009 \cdot (((32 + 10) \cdot 8) / 50000)$ J

Receiving = Sending. Idle Node = $3 \cdot 0.0125 \cdot 0.115 - 2 \cdot 3 \cdot (0.0125 - 0.009) \cdot ((10 \cdot 8) / 50000)$ J

¹⁰ Total = Sending (Node 1) + Receiving (Node 2) + Idle Node (Node 3).

Sending = $3 \cdot 0.0125 \cdot 0.001 + 0.009 \cdot ((10 \cdot 8) / 50000) + 3 \cdot 0.009 \cdot (((32 + 10) \cdot 8) / 50000)$ J

Receiving = $3 \cdot 0.00001 \cdot 0.002 + 3 \cdot 0.009 \cdot (((32 + 10) \cdot 8) / 50000)$ J. Idle = $3 \cdot 0.00001 \cdot 0.002$ J

different values using the same MAC scheme. Different simulators provided different results. Also, these simulation results did not provide the factors that greatly influenced the energy savings and it was difficult to prove that what the simulators were computing was correct. Thus many hours of work had to be discarded and in order to obtain results and submit within the required time, it was decided to resort to mathematical analysis which is how all the results were obtained.

5.5.3 Plotting the Schemes versus Each Other in Equality

There is a need to be able to plot the two schemes on the same graph so that they can be compared directly, even though they are very different from each other.

The MAC schemes, being an inter-nodal protocol, can only be directly compared if the number of nodes over a given radio range is equal.

The output is always the energy consumed by the two protocols. They are taken to execute the same command. There are scaling factors that affect the output and these must be identified and then kept the same for both MAC protocols.

The following are the scaling factors:

- Microcontroller used
- Distance between nodes

Thus, on one X axis, there is the number of nodes in a given radio range and on the other Z axis the energy used, as Figure 5.15 illustrates:

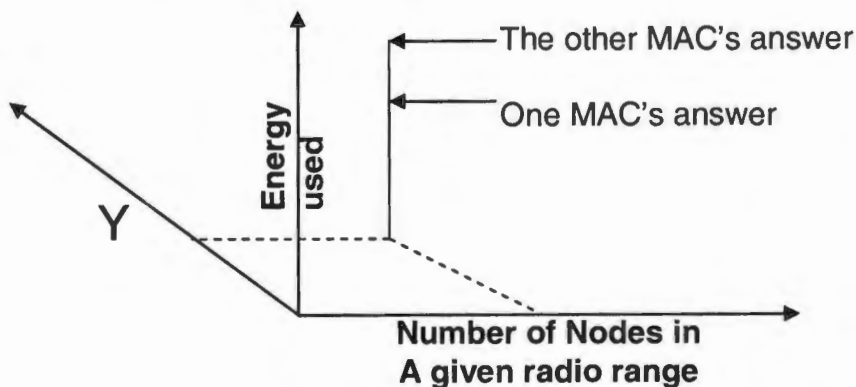


Figure 5.15 Basic 3D Graph to Compare S-MAC and Colour TDMA MAC with no Variable for the Y-Axis

Different variables can be put on the Y axis in order to compare the two MAC protocols. One such measurement is: "The number of messages that need to be sent in one duty cycle." This would provide the sending and receiving energy of two schemes in one duty cycle. It would be necessary to take in to account the collisions of S-MAC. If a collision does occur (in S-MAC) then the node will spend a cycle idle listening.

A more pertinent measurement would be: “Number of messages that need to be successfully sent.” This would include the energy wasted in collisions and it would be a more accurate figure for the comparison. However, obtaining it would be difficult as S-MAC would require a number of duty cycles to send a given set of messages, whereas Colour TDMA MAC would only require one (providing the data is less than five hundred and twelve bytes).

There is another issue which is illustrated in Figure 5.16 where there are five nodes in a given radio range with four messages that need to be successfully sent. Figure 5.16 does not cover all the possibilities. The small circles are the nodes and the arrows are the messages that originate from the sender and point to the receiver.

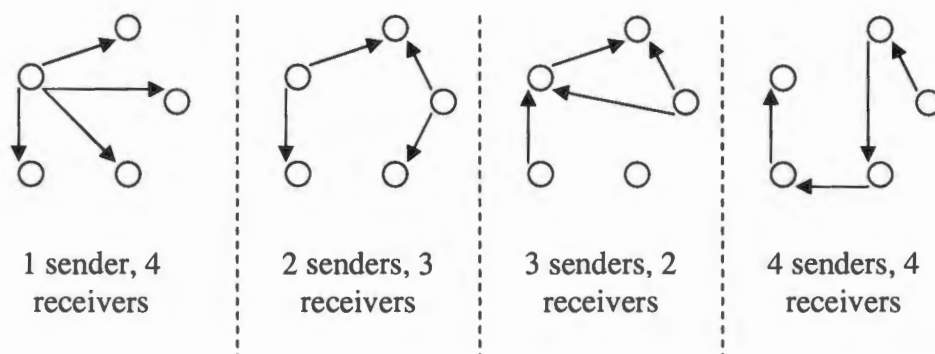


Figure 5.16 Number of Different Senders and Receivers and Their Energy Consumption

To resolve the issue of how much energy is used by each scheme, if there are five nodes and four messages, all the possibilities for that specific situation are required to be considered but not all configurations are equally likely.

Thus for one point shown on Figure 5.15 with “Number of messages that need to be successfully sent” as its Y axis, there can be more than one answer.

Lastly, there is a need to place boundaries on the X and Y axis values. Small values are investigated before moving to larger values. A small number for the X axis would be five. This is advantageous to Colour TDMA MAC because bandwidth use decreases rapidly as the number of nodes in a given radio range increases. The $X = 1$ plane in Figure 5.15 will not be considered because that doesn't constitute a network. A small number for the Y axis would be ten. A low number it is advantageous to S-MAC because as the number of contenders rises in S-MAC, the number of successful transmissions decreases rapidly.

What follows is an examination of the boundary figures.

5.5.4 Boundary Values – Two Nodes and Two Messages

Four issues are investigated in this subsection:

- 1 The different combinations of senders and receivers.
- 2 The energy used in one duty cycle by the radio and the microcontroller according to those combinations.
- 3 The number of successful transmissions in that duty cycle
- 4 The time taken to send the successful transmissions.

Combinations:

This is a trivial data transmission scenario, where the receiver is always the node not sending. Either the one node will send both messages or each node will send one message to its neighbour as shown in Table VIII.

TABLE VIII
CONFIGURATION OF TWO NODES AND TWO MESSAGES TO BE SENT

	Node A sending	Node B sending
Node B receiving	2	0
Node A receiving	1	1

Energy:

Where one node is sending two messages to its neighbour it gives the graph of the radio energy utilised by S-MAC as illustrated in Figure 5.17. The microprocessor is on the entire time from the beginning of the graph to when data transmission completes. That microprocessor energy is added into the equation but it is not illustrated in the Figure 5.17. Figure 5.18 shows the Colour TDMA MAC graph of radio utilisation, where the microprocessors awake time is not shown in the figure. The microprocessor and radio usage result depends on the maximum number of neighbours there are.

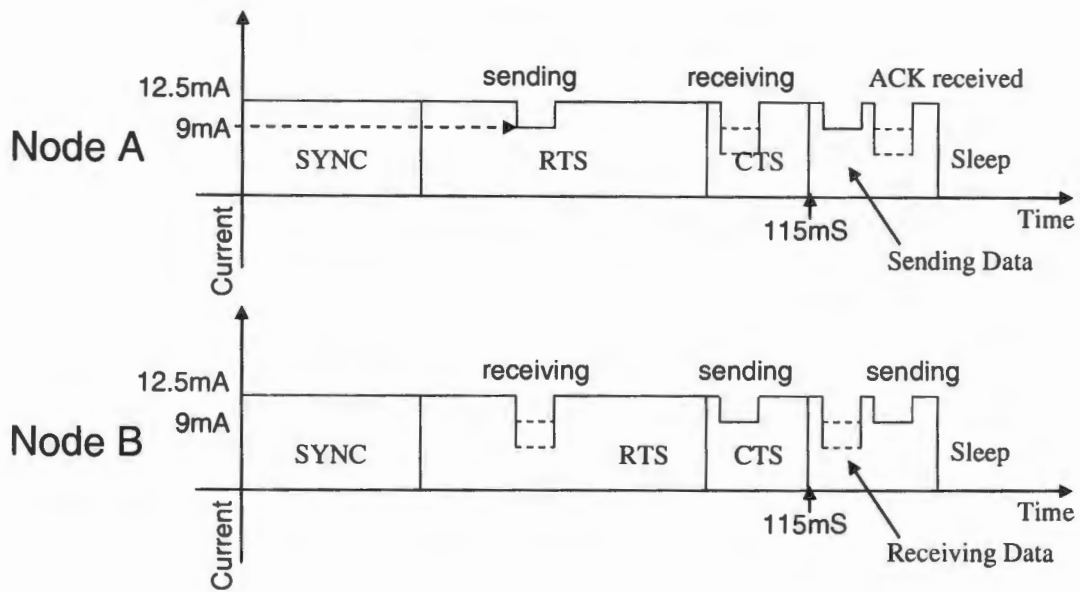


Figure 5.17 S-MAC Two Nodes, One Node Sending Two Messages to its Neighbour

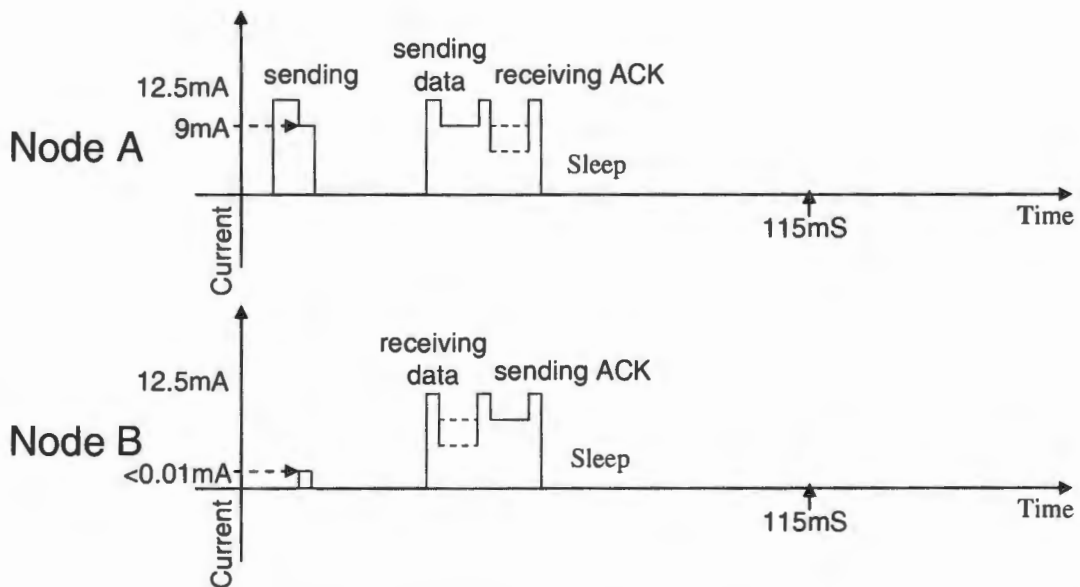


Figure 5.18 Colour TDMA Two Nodes, One Node Sending Two Messages to its Neighbour

Even before investigating the energy values it can see how Colour TDMA MAC is close to the ideal and how far away S-MAC is from the ideal.

The energy used by a scheme in one duty cycle is equal to the sum of the following power utilisations:

- The sender's microprocessor
- The sender's transmitter
- The receiver's microprocessor

- The receiver's transmitter
- The neighbour's microprocessor (neither sending nor receiving)
- The neighbour's transmitter (neither sending nor receiving)

Table IX shows the energy usage of SMAC (from the above mentioned six points) for the first situation in Table VIII, where two messages are being sent by one node.

In Table IX and in the tables for the rest of this chapter: μC represents the microcontroller and x-mitter is a shortened name for the transmitter.

TABLE IX

S-MAC ENERGY USAGE FOR TWO NODES WHERE ONE NODE SENDS TWO MESSAGES

Sender (Node A)		Receiver (Node B)		Neighbours		Total
μC	X-mitter	μC	X-mitter	μC	X-mitter	
0.38532mJ	4.64178mJ	0.38532mJ	4.64178mJ	0mJ	0mJ	10.0542mJ

Node A's microcontroller (μC) equation in Table IX is described below. First, it is explained in words that relate to Figure 2.2 and then the mathematics is presented below the worded explanation. The values used in the mathematics are from Tables V and VI, Section 5.3 "S-MAC Mathematical Contention Analysis" and 5.4 "Colour TDMA MAC Properties and The Schedule."

Time awake during SYNC, RTS and CTS plus time sending data.
(See Figure 2.2)

$$3\text{V} \cdot 0.001\text{A} \cdot 0.115\text{S} + 3\text{V} \cdot 0.001\text{A} \cdot (((2 \cdot 32 + 2 \cdot 10) \cdot 8) / 50000) = 0.38532\text{mJ}.$$

Node A's radio transmitter (X-mitter) in Table IX:

Awake during SYNC, RTS and CTS plus sending one RTS and receiving one CTS plus sending two messages and receiving two ACK. (See Figure 5.17)

$$3\text{V} \cdot 0.0125\text{A} \cdot 0.115\text{S} - 2 \cdot 3\text{V} \cdot (0.0125\text{A} - 0.009\text{A}) \cdot ((10 \cdot 8) / 50000) + 3\text{V} \cdot 0.009\text{A} \cdot (((2 \cdot 32 + 2 \cdot 10) \cdot 8) / 50000) = 4.64178\text{mJ}.$$

Node B's energy utilisation in Table IX is the same as node A's from the Figures 2.2 and 5.17.

There are no neighbours because there are only two nodes.

Table X shows the energy usage of SMAC for the second situation in Table VIII, where each node sends one message. It assumes that there are no collisions.

TABLE X

S-MAC ENERGY USAGE FOR TWO NODES WHERE EACH NODE SENDS ONE MESSAGE

Node A		Node B		Neighbours		Total
μC	X-mitter	μC	X-mitter	μC	X-mitter	
0.38532mJ	4.60818mJ	0.38532mJ	4.60818mJ	0mJ	0mJ	9.987mJ

Node A's microcontroller equation in Table X:

Time awake during SYNC, RTS and CTS plus time sending one data message plus time receiving one data message with ACK. (See Figure 2.2)

$$3\text{V} \cdot 0.001\text{A} \cdot 0.115\text{S} + 2 \cdot 3\text{V} \cdot 0.001\text{A} \cdot \left(\frac{(32+10) \cdot 8}{50000} \right) = 0.38532\text{mJ}.$$

Node A's radio transmitter in Table X:

Awake during SYNC, RTS and CTS plus sending one RTS and receiving a RTS and sending a CTS and receiving one CTS plus sending one messages and receiving one ACK and receiving one message and sending one ACK.

$$3\text{V} \cdot 0.0125\text{A} \cdot 0.115\text{S} - 4 \cdot 3\text{V} \cdot (0.0125\text{A} - 0.009\text{A}) \cdot \left(\frac{10 \cdot 8}{50000} \right) + 2 \cdot 3\text{V} \cdot 0.009\text{A} \cdot \left(\frac{(32+10) \cdot 8}{50000} \right) = 4.60818\text{mJ}.$$

Node B's energy utilisation in Table X is the same as node A's.

There are no neighbours because there are only two nodes.

Table XI shows the energy usage of Colour TDMA MAC (with $\alpha = 2$) for the first situation in Table VIII, where two messages are being sent by one node.

TABLE XI

COLOUR TDMA MAC ENERGY USAGE FOR TWO NODES

WHERE ONE NODE SENDS TWO MESSAGES

Case:	Sender (Node A)		Receiver (Node B)		Neighbours		Total
	μC	X-mitter	μC	X-mitter	μC	X-mitter	
Best	0.085mJ	0.454mJ	0.049mJ	0.387mJ	0mJ	0mJ	0.9775mJ
Worst	0.085mJ	0.454mJ	0.069mJ	0.387mJ	0mJ	0mJ	0.9967mJ

The best and worse cases are in Table XI because depending which timeslot the receiving node chooses, its microcontroller will be on for a shorter or a longer period. The best case values always stay the same because the receiver always awakens just before data transmission.

Node A's microcontroller equation in Table XI:

Time awake during timeslot plus time sending one data message plus time receiving one ACK message. (See Figure 3.24)

$$3V \cdot 0.001A \cdot (5.45\text{mS} + 3.2 \cdot \beta)\text{mS} + \text{"time to send data and receive ACK"}$$

$$3V \cdot 0.001A \cdot (0.00545 + 0.0032 \cdot 3) + 3V \cdot 0.001A \cdot (((2 \cdot 32 + 2 \cdot 10) \cdot 8) / 50000) = 0.08547\text{mJ}.$$

It can be seen that Colour TDMA MAC microcontroller usage is about four and a half times smaller than S-MAC with respect to node A / sender.

Node A's radio transmitter in Table XI:

Awake in "Node B's PicoRadio enabled period" and send a "wake up node B" control message + radio wake up and send the two data messages and receive two ACK. (See Figure 5.17)

$$2 \cdot 3V \cdot 0.0125A \cdot 650\mu\text{S} + 3V \cdot 0.009A \cdot ((10 \cdot 8) / 50000) + 3V \cdot 0.009A \cdot (((2 \cdot 32 + 2 \cdot 10) \cdot 8) / 50000) = 0.45483\text{mJ}.$$

It can be seen that Colour TDMA MAC radio usage is about ten times smaller than S-MAC with respect to node A / sender.

Node B's microcontroller equation in Table XI:

Best case scenario: (node B timeslot number is 3), then the energy used is:

$$3V \cdot 0.001 \cdot (3.2 \text{mS}) + \text{"time to send data and receive ACK"} \text{ which is:}$$

$$3V \cdot 0.001 \cdot (0.0032) + 3V \cdot 0.001A \cdot (((2 \cdot 32 + 2 \cdot 10) \cdot 8) / 50000) = 0.04992 \text{mJ}.$$

Worst case scenario: (node B timeslot number is 1), then the energy used is:

$$3V \cdot 0.001 \cdot (3.2 \cdot \beta) \text{mS} + \text{"time to send data and receive ACK"} \text{ which is:}$$

$$3V \cdot 0.001 \cdot (0.0032 \cdot 3) + 3V \cdot 0.001A \cdot (((2 \cdot 32 + 2 \cdot 10) \cdot 8) / 50000) = 0.06912 \text{mJ}.$$

It can be seen that Colour TDMA MAC microcontroller worst case usage is about five and a half times smaller than S-MAC with respect to node B / receiver.

Node B's radio transmitter in Table XI:

PicoRadio enabled + radio wake up and receive data and send ACK.

$$3V \cdot 0.00001A \cdot 0.0032S + 3V \cdot 0.0125A \cdot 650\mu S +$$

$$3V \cdot 0.009A \cdot (((2 \cdot 32 + 2 \cdot 10) \cdot 8) / 50000) = 0.387375 \text{mJ}.$$

It can be seen that Colour TDMA MAC radio usage is about twelve times smaller than S-MAC with respect to node B / receiver.

Table XII shows the energy usage of Colour TDMA MAC (with $\alpha = 2$) for the second situation in Table VIII (where each node sends one message).

TABLE XII

COLOUR TDMA MAC ENERGY USAGE FOR TWO NODES

WHERE EACH NODE SENDS ONE MESSAGE

Case:	Node A		Node B		Neighbours		Total
	μC	X-mitter	μC	X-mitter	μC	X-mitter	
Best	0.095mJ	0.479mJ	0.095mJ	0.479mJ	0mJ	0mJ	1.1487mJ
Worst	0.114mJ	0.479mJ	0.114mJ	0.479mJ	0mJ	0mJ	1.1871mJ

Node A's microcontroller equation in Table XII:

As a sender:

Time awake during timeslot plus time sending one data message plus time receiving one ACK message. (See Figure 3.24)

$$3V \cdot 0.001A \cdot (5.45 + 3.2 \cdot \beta) \text{mS} + \text{"time to send data and receive ACK"}$$

$$3V \cdot 0.001A \cdot (0.00545 + 0.0032 \cdot 3) + 3V \cdot 0.001A \cdot (((32+10) \cdot 8) / 50000) = 0.06531 \text{mJ.}$$

As a receiver:

Same as node B as a receiver.

$$\textit{Sender + receiver (best case) = 0.09507mJ}$$

$$\textit{Sender + receiver (worst case) = 0.11427mJ}$$

Node A's radio transmitter in Table XII:

As a sender:

Awake in "Node B's PicoRadio enabled period" and send a "wake up node B" control message + radio wake up and send the data messages and receive an ACK.

$$2 \cdot 3V \cdot 0.0125A \cdot 650 \mu\text{S} + 3V \cdot 0.009A \cdot ((10 \cdot 8) / 50000) + 3V \cdot 0.009A \cdot (((32+10) \cdot 8) / 50000) = 0.27339 \text{mJ.}$$

As a receiver:

Same as node B as a receiver.

$$\textit{Sender + receiver = 0.479301mJ}$$

Node B's microcontroller equation in Table XII:

As a sender:

Same as node A as a sender.

As a receiver:

Best case scenario: (node B timeslot number is 3), then the energy used is:

$$3V * 0.001 * (3.2mS) + \text{“time to send data and receive ACK” which is:}$$

$$3V * 0.001 * (0.0032) + 3V * 0.001A * (((32+10)*8)/50000) = 0.02976mJ.$$

Worst case scenario: (node B timeslot number is 1), then the energy used is:

$$3V * 0.001 * (3.2 * \beta)mS + \text{“time to send data and receive ACK” which is:}$$

$$3V * 0.001 * (0.0032 * 3) + 3V * 0.001A * (((32+10)*8)/50000) = 0.04896mJ.$$

$$\text{Sender + receiver (best case) = } 0.09507mJ$$

$$\text{Sender + receiver (worst case) = } 0.11427mJ$$

Node B's radio transmitter in Table XII:

As a sender:

Same as node A as a sender.

As a receiver:

PicoRadio enabled + radio wake up and receive data and send ACK.

$$3V * 0.00001A * 0.0032S + 3V * 0.0125A * 650\mu S +$$

$$3V * 0.009A * (((32+10)*8)/50000) = 0.205911mJ.$$

$$\text{Sender + receiver = } 0.479301mJ$$

Table XIII shows the *comparison of energy usage of S-MAC and worst case Colour TDMA MAC* (tables IX, X, XI, XII) with respect to the scenarios in Table VIII.

TABLE XIII

COMPARISON OF S-MAC VS COLOUR TDMA MAC ENERGY USAGE FOR TWO NODES WHERE TWO MESSAGES NEED TO BE SENT

	S-MAC	Colour TDMA MAC	Colour TDMA MAC improvement factor over S-MAC
2 nodes, one sending 2 messages	10.0542mJ	0.9967mJ	10.08748
2 nodes, each one sending 1 message	9.987mJ	1.1871mJ	8.41293

In Table XIII, Colour TDMA MAC out performs S-MAC by a minimum factor of eight for the worst case of Colour TDMA MAC.

This is for when ($\alpha = 2$) which is the correct choice for the network. The energy used by Colour TDMA MAC must be investigated for when the “wrong” α is chosen for two nodes with two messages to send. α affects the length of the TDMA scheme and hence the time the microcontroller is on, but not period that the transmitter is on. (See Figure 5.18).

Node A’s (the sender’s) microcontroller if $\alpha = 2$ then the energy used is:

$$3V \cdot 0.001A \cdot (5.45mS + 3.2 \cdot \beta) mS + \text{“time to send data and receive ACK”}$$

$$3V \cdot 0.001A \cdot (5.45 + 3.2 \cdot 3)mS + 3V \cdot 0.001A \cdot (((2 \cdot 32 + 2 \cdot 10) \cdot 8) / 50000) = 0.08547mJ.$$

If $\alpha = 3$ node A’s microcontroller utilises 0.12387mJ.

If $\alpha = 4$ node A’s microcontroller utilises 0.18147mJ.

If $\alpha = 5$ node A’s microcontroller utilises 0.25827mJ.

If $\alpha = 6$ node A’s microcontroller utilises 0.35427mJ.

If $\alpha = 7$ node A’s microcontroller utilises 0.46947mJ.

If $\alpha = 10$ node A’s microcontroller utilises 0.93027mJ.

It can be seen that α affects the senders microprocessor’s energy usage. What is also clear is that the microprocessor usage at $\alpha = 7$ uses nearly the same amount of energy as the radio transmitter. Thus, once the radio transmitter has been reduced to almost ideal, the next energy efficiency target is the microprocessor.

Successful Transmission:

The probability of collisions in S-MAC is small and negligible. Colour TDMA MAC does not suffer from internal collisions hence it will have a 100% success rate.

Time to Send:

The duration of sending messages in S-MAC is one awake duty cycle plus the time taken for the data transmission. Therefore, it is:

$$115mS + (((2 \cdot 32 + 2 \cdot 10) \cdot 8) / 50000)S = 128.44mS$$

and the next transmission is allowed to happen in (sleep time)

$$1035mS - (\text{data transmission}) 13.44mS = 1021.56mS.$$

The duration of sending messages in Colour TDMA MAC is within one timeslot (and that depends on α) plus the time taken for the data transmission. Thus for $\alpha = 5$ and node B in timeslot 1 it is:

$$(3.2*\beta)mS + (((2*32+2*10)*8)/50000)S = 112.64mS$$

Although this seems faster than S-MAC, the next transmission is allowed to happen in

$$(((85.85 + 3.2*\beta)* \beta) + 48.97)mS - 13.44mS$$

which is in 3249.58mS time. That is about three times slower than S-MAC. If $\alpha = 2$ then a time to send the message is 23.04mS and the next transmission would happen at 325.88mS which is much faster than S-MAC. **Thus ensuring the right α is chosen for the network is very important.**

5.5.5 Boundary Values – Two Nodes and Ten Messages

Four issues are investigated in this subsection:

- 1 The different combinations of senders and receivers.
- 2 The energy used in one duty cycle by the radio and the microcontroller according to those combinations.
- 3 The number of successful transmissions in that duty cycle
- 4 The time taken to send the successful transmissions.

Combinations:

Two nodes, A and B are in radio range of each other. There are between 1 and 10 messages to be sent between the two nodes. The number of possible configurations for sending is many but the number of receivers is always one. 'The node not sending' is always the receiver. Table XIV investigates the number of different configurations for sending between 1 and 5 messages.

TABLE XIV

TWO NODES SENDING BETWEEN 1 AND 5 PACKETS BETWEEN EACH OTHER

1 messages		2 messages		3 messages		4 messages		5 messages	
A	B	A	B	A	B	A	B	A	B
1	0	2	0	3	0	4	0	5	0
0	1	1	1	2	1	3	1	4	1
		0	2	1	2	2	2	3	2
				0	3	1	3	2	3
						0	4	1	4
								0	5

What is first visible from Table XIV is that the energy used by node A sending one message to node B is the same as the energy used by node B sending one message to node A. The same is true for a few of the numbers in Table XIV. Another example is node A sending two messages to node B and node B sending one message to node A, which is the same as, node A sending one message to node B and node B sending two messages to node A.

Discrete mathematics solves this problem and the number of different possibilities is reduced by half. However, this is not required as the next paragraph proves firstly in S-MAC and then in Colour TDMA MAC.

What is evident is whether node A is sending five messages and node B five messages or node A is sending nine messages and node B one message, there will still be two RTS and two CTS by S-MAC and two signals awaking neighbours in Colour TDMA MAC. Following these signals is the data being sent. The radio diagrams of these situations are almost identical to the previous subsection 5.4.6 “Boundary Values – Two Nodes and Two Messages” but in the sending data stage there is more data to send.

Energy:

The overall values for S-MAC and Colour TDMA MAC do not grow too much compared to the last subsection. As shown below for certain cases:

S-MAC, one node sending ten messages: 13.2798mJ, whereas the value for one node sending two messages was: 10.0542mJ.

Colour TDMA MAC with $\alpha = 5$ and node B (the receiver) in timeslot 1, one node sending ten messages: 4.567971mJ. Disregarding $\alpha = 5$, the value for the transmission becomes: 4.1595mJ.

Colour TDMA MAC is far less affected by a change in the number of transmissions compared to S-MAC in this example.

Successful Transmissions:

Again the best case for S-MAC is assumed, where there should be no collisions. Colour TDMA MAC also has no collisions.

Time to Send:

The duration of sending the messages in S-MAC is one awake duty cycle plus the time taken for the data transmission. Thus it is:

$$115\text{mS} + (((10*32+10*10)*8)/50000)\text{S} = 182.2\text{mS},$$

and the next transmission is allowed to happen in (sleep time):

$$1035\text{mS} - (\text{data transmission}) 67.2\text{mS} = 967.8\text{mS}.$$

The duration of sending the messages in Colour TDMA MAC is within one timeslot (and that depends on α) plus the time taken for the data transmission. Thus for $\alpha = 5$ and node B in timeslot 1 it is:

$$(3.2*\beta)\text{mS} + (((10*32+10*10)*8)/50000)\text{S} = 67.2672\text{mS}$$

Although this seems faster than S-MAC, the next transmission is allowed to happen in:

$$(((85.85 + 3.2*\beta)*\beta) + 48.97)\text{mS} - 67.2672\text{mS}$$

which is in 3195.7528mS time. That is slow compared to S-MAC, but if S-MAC experiences any collisions then 967.8mS is doubled to nearly 2000mS. The probability of collision is less than 10% from Figure 5.3

5.5.6 Boundary Values – Five Nodes and Two Messages

This is another trivial case but it is expected that S-MAC energy usage will jump up quite a bit, because there are five nodes and two messages and most of the nodes will be idle listening. Colour TDMA MAC, being close to the ideal, should not suffer from this problem because it will not be waking any neighbours unnecessarily.

Four issues are investigated in this subsection:

- 1 The different combinations of the senders and receivers.
- 2 The energy used in one duty cycle by the radio and the microcontroller according to those combinations.
- 3 The number of successful transmissions in that duty cycle
- 4 The time taken to send the successful transmissions.

Combinations:

It does not matter whether node A is sending two messages to B, C, D or E or whether C is sending two messages to A, B, D, and E. There are many combinations but many share the same result. The important combinations are the ones where one node will send two messages, or two nodes send one message each.

Energy:

The energy used by S-MAC for one node sending two messages is: 23.9259mJ.

Radio Transmitters:

$$5*3\text{V}*0.0125\text{A}*0.115\text{S} - 5*2*3\text{V}*(0.0125\text{A}-0.009\text{A})*((10*8)/50000) + 2*3\text{V}*0.009\text{A}*(((2*32+2*10)*8)/50000) = 22.12026\text{mJ}.$$

Regardless of the number of packets being sent, as the number of neighbours increase in S-MAC, its radio idling increases and the energy wasted involved becomes more evident.

Microcontrollers:

$$5*3V*0.001A*0.115S + 2*3V*0.001A*(((2*32+2*10)*8)/50000) = 1.80564mJ.$$

The energy used by Colour TDMA MAC for one node sending two messages is:

The energy used by Colour TDMA MAC for one node sending two messages and $\alpha = 5$ considering the receiver is in timeslot 1: 1.470345mJ

This is from two nodes, two messages energy used (1.468845mJ) plus the PicoRadios energy consumed:

$$5*5*3V*0.00001A*0.0016 = 0.0000012mJ.$$

Successful Transmissions:

Again it is assumed that collisions in S-MAC are too small to be taken into account. One node is sending one message and another node is sending only one message. Colour TDMA MAC does not have any collisions.

Time to Send:

Looking at the next time a node will be able to send data after this round:

- S-MAC will be able to send the next message in 1035mS time.
- Colour TDMA MAC will be able to send the next message in 3262.9528mS.

5.5.7 Boundary Values – Five Nodes and Ten Messages

Four issues are investigated in this subsection:

- 1 The different combinations of the senders and receivers.
- 2 The energy used in one duty cycle by the radio and the microcontroller according to those combinations.
- 3 The number of successful transmissions in that duty cycle
- 4 The time taken to send the successful transmissions.

Combinations:

This is where obtaining one value for each scheme becomes more difficult. The ten messages can be split up between five nodes in quite a few ways.

There are thirty-one different combinations for the sender shown in Table XV, where the following alphabet characters represent a node:

TABLE XV
FIVE NODES, AND TEN MESSAGES TO BE SENT

A sends	B sends	C sends	D sends	E sends
10	0	0	0	0
9	1	0	0	0
8	2	0	0	0
8	1	1	0	0
7	3	0	0	0
7	2	1	0	0
7	1	1	1	0
6	4	0	0	0
6	3	1	0	0
6	2	2	0	0
6	2	1	1	0
6	1	1	1	1
5	5	0	0	0
5	4	1	0	0
5	3	2	0	0
5	3	1	1	0
5	2	1	2	0
5	2	1	1	1
5	1	1	1	2
4	4	2	0	0
4	4	1	1	0
4	3	3	0	0
4	3	2	1	0
4	3	1	1	1
4	2	2	2	0
4	2	2	1	1
3	3	3	1	0
3	3	2	2	0

3	3	2	1	1
3	2	2	2	1
2	2	2	2	2

NB whether A, B or C sends 10 messages it is the same situation being evaluated.

For each one of the senders (except the last one), there is a combination to which node is the receiver. E.g. Node A is to send ten messages. It can send all ten to one neighbour, or nine to one neighbour and one to another neighbour. Or eight messages to one neighbour and two to another neighbour and so on. Thus the exact number of investigations that is required is a high number. The chance of one node sending many messages is low as shown by the following example. There is a one in thirty-one chance that the sender will send ten messages but then there is another one in thirty-one chance that there will be only one receiver. One in thirty-one is 3% but one in thirty-one squared gives a 0.001% chance of that occurring.

Energy, Successful Transmissions and Time to Send:

With so many combinations, what is clear is that there will be many collisions. As was noted in the last subsection, the number of neighbours greatly influences the amount of energy expended by S-MAC per round. Regardless of whether two or ten messages are sent through, the minimum energy used in the round will be 23.9259mJ for S-MAC, the figure from the last subsection.

Collisions of RTS and CTS, do not consume much of the radio power. 12.5mA goes down to 9mA, whether the node is receiving a signal that is valid, invalid or not destined for it. The fact that the message did not win the channel this round means it must contend in the next round, which means another 23.9259mJ must be spent. If the message is not sent then, it is required to be sent in the round after and so on. Colour TDMA MAC does not suffer from collisions and even though it takes 3263.02mS per TDMA round, it still gets the message through and it does not involve waiting another 3263.02mS. It does this using the minimum number of nodes. The value of five nodes sending ten messages for Colour TDMA MAC is thus estimated to use about the same amount of energy as two nodes and ten messages. This value is 4.567971mJ, where the dominant value was in the number of packets that were required to be sent.

5.5.8 Points in Between and Points Outside the Boundaries

What has been discovered from the boundary results should be consistent through the rest of the graph. Thus choosing a point somewhere in the middle, for example three nodes four messages, or another point such as four nodes three messages, should present something according to the trends learnt at the boundaries. This should also be the same for points of the graph outside the chosen boundaries.

5.5.9 Plotting the Energy Usage

Thus with the data gathered from the boundaries, graphs can be plotted (as shown in Figures 5.19 and 5.20), where the Y axis is number of messages sent in one round.

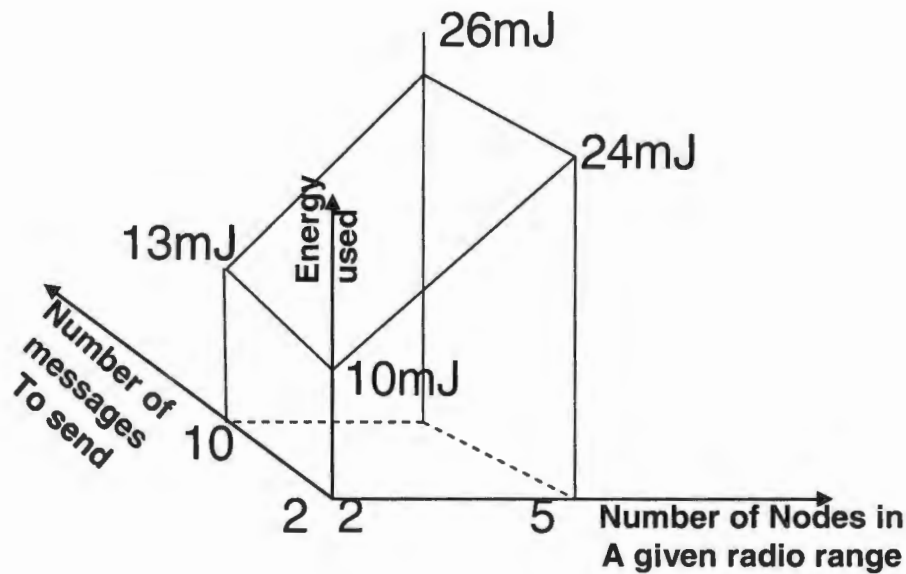


Figure 5.19 S-MAC Results

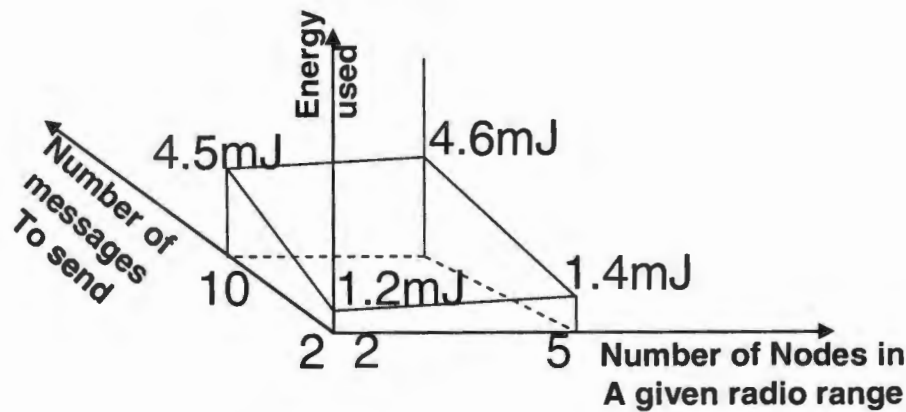


Figure 5.20 Colour TDMA MAC Results

Figure 5.19 indicates that successful transmissions go down rapidly when moving outwards from the origin. The same is not the case for Colour TDMA MAC where successful transmissions are guaranteed. When S-MAC experiences a collision, that data must be sent in the next duty cycle, and two duty cycles add up to 2.3mS and three add up to 3.45mS which adds onto the latency of a packet moving from one node to its neighbour, i.e. hop-to-top delivery times go up.

Figure 5.20 indicates that the delivery time for a message is exponentially increased when moving outwards from the origin along the X = Y line of Colour TDMA MAC graph.

5.5.10 Time to Pass a Message from One Hop to the Next

As the number of messages being sent between nodes increases, S-MAC suffers from collisions and uses more time to deliver messages to its neighbour. Colour TDMA MAC has a set time that depends on the maximum amount of neighbours that are

allowed in its scheme. Figure 5.21 shows the time it takes for a message to be delivered and the values for S-MAC have been derived from Figure 5.3 and Table VII assuming that there will be a collision. The values for Colour TDMA MAC follow the equation:

$$((85.85 + 3.2*\beta)* \beta)mS + 48.97mS$$

It is clear from Figure 5.21 that if there are collisions that Colour TDMA MAC will in fact deliver the message quicker than S-MAC.

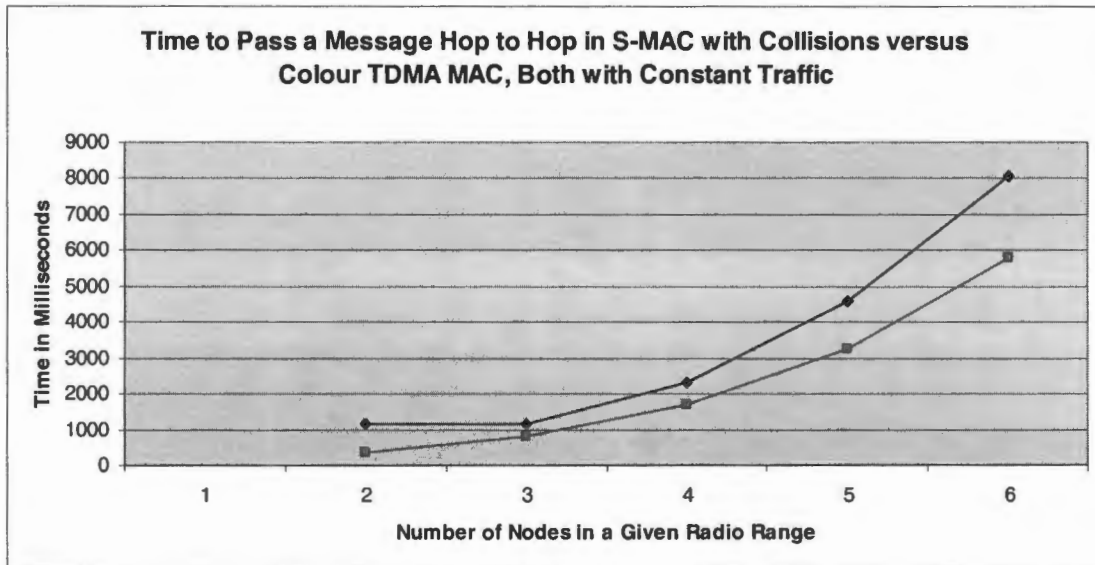


Figure 5.21 Time to Pass a Message Hop to Hop in S-MAC with Collisions versus Colour TDMA MAC, Both with Constant Traffic

5.5.11 S-MAC Initialisation versus Colour TDMA MAC Initialisation

In the operation stage of the scheme, great favour is shown to Colour TDMA as far as energy efficiency goes, but Colour TDMA MAC is not energy efficient during initialisation. S-MAC is far simpler and it uses less energy due to there being no associated administration tasks. However, it is preferable to spend a large amount once-off in the beginning and then spend only the bare minimum whenever it is needed rather than spend an average amount round after round.

5.6 Guideline Results for Non-Contention Based MAC

The TDMA schemes that employ cluster heads or co-ordinating nodes for the administration of the TDMA scheme are energy inefficient. They require that one node from a selection or cluster be awake for an entire duty cycle and to use its radio, the most energy consuming device of an AHWSN node, for at least one duty cycle to co-ordinate fellow nodes. Approximately “(number of timeslots -1)*(number of neighbours)” duty cycles later and it is that node’s turn to co-ordinate again, and normally for a number of rounds. Thus, when comparing Colour TDMA MAC with TDMA MAC protocols that use some form of hierarchy, Colour TDMA MAC should provide outstanding energy improvement results as it did against S-MAC.

6 CONCLUSION

With the results having been presented in the previous chapter, it is necessary to reflect on and highlight some important outcomes and results in this project and in particular the conclusions reached.

6.1 Colour TDMA MAC versus Other MAC Protocols

6.1.1 Colour TDMA MAC versus S-MAC

The energy efficiency of the two schemes:

Colour TDMA MAC is more energy efficient than S-MAC:

- Almost no idle waiting for a data transfer. In S-MAC as the number of nodes in a radio range increases, the energy wasted by idle waiting increases.
- Maximum sleep time, minimum awake time. Sending and Receiving is almost ideal.
- No inter-nodal collisions.
- Radio transmitter utilisation. Sending and Receiving is almost ideal.

Colour TDMA MAC is less energy efficient than S-MAC:

- Initialisation.
- Nodes joining the scheme.

Aspects other than energy efficiency:

Colour TDMA MAC is more efficient than S-MAC:

- Colour TDMA MAC can guarantee when a message will be delivered, the time depends on the number of timeslots. S-MAC cannot guarantee a time for delivery.

Colour TDMA MAC is less efficient than S-MAC at the following when there is a low number of neighbours:

- Bandwidth utilisation. When there are many timeslots in Colour TDMA MAC, less than a quarter are used by a given group of nodes within a radio range.
- Delivery of the message is quicker with S-MAC when a small amount of data (between few neighbours) is being transferred.

However, if there are collisions in S-MAC, Colour TDMA MAC is more efficient at both of the above. If there are three nodes with one sending ten messages and there is a collision in S-MAC, then Bandwidth use will be zero for that round, whereas Colour TDMA MAC guarantees delivery. If there is a collision in S-MAC, the time until successful delivery is longer than Colour TDMA MAC as shown in Figure 5.21

6.1.2 Colour TDMA MAC versus other TDMA MACs

Colour TDMA MAC offers a better TDMA scheme and energy utilisation than other TDMA MACs [4], [8] for AHWSN, due to its unique features:

- It does not require strict timing for the TDMA scheme.
- There is no central point or super node(s) co-ordinating the TDMA scheme.
- The nodes with their neighbours do not form clusters.
- If any node leaves the TDMA sequence, it does not stop the continuous cycle.
- The receiving node does not need to be awake to obtain data.

6.1.3 Colour TDMA MAC Uniqueness in AHWSN

Colour TDMA MAC has the following unique features that are the first of their kind for TDMA MAC protocols in AHWSN:

- If any node fails the TDMA cycle continues uninterrupted. [9] and [23] offer a distributed scheme but there are still nodes in charge of timeslots.
- There is no need for nodes to form clusters or a hierarchy while the scheme is in operation.

6.1.4 Extra Hardware and Costs in Colour TDMA

Colour TDMA MAC requires a PicoRadio which is additional hardware and a cost to the node. A PicoRadio is a simple device that awakens the node to receive data. Its simple structure consists of a few components and it is not a costly item to add to the node. A few cents more is the maximum amount that would be added to the cost of each node.

Thus, there is little doubt that Colour TDMA MAC in an AHWSN is a unique solution when looked at from various perspectives, whether energy use is the critical factor or the hardware costs.

7 FUTURE WORK

There were some ideas left for future work in this dissertation. They were:

- A solution to what could be done if there were more than the allowed maximum amount of neighbours to a given node in a Colour TDMA MAC scheme
- A MEMS PicoRadio
- The Mailbox

Point one and point three are discussed further, where as the MEMS PicoRadio is left open for the reader's mind.

7.1 "The Mailbox" and 7.2 "Time Sharing" 7.3 "The MAC layer and Routing" are to a degree complementary ideas, they could become the subject of the same future investigation and design techniques. Thus they can be researched together by one person.

7.1 The Mailbox

The mailbox concept would allow a move away from the PicoRadio. It envisages a process where a message destined for an asleep node does not require it to awaken and turn on to receive the message but that the message can be stored in a "mailbox" elsewhere while the node remains asleep. Later when the node awakens for its timeslot it can check its mailbox and retrieve "any mail".

7.2 Time Sharing

Colour TDMA MAC does not have a restriction on the number of nodes in the overall network but on how many neighbours any given node may have. The maximum is α . If there are more than α , then a solution is that the nodes go into a time sharing where for every even duty cycle, one set of nodes runs its TDMA schedule and every odd duty cycle another set of nodes runs its own TDMA schedule. Exactly how this would work is left for future work.

7.3 The MAC Layer and Routing

The MAC layer determines which nodes may use the channel when, and the routing protocol determines where the message is headed. The two can help each other. The MAC layer needs to know its entire one and two-hop neighbours. Thus routing and the Colour TDMA MAC could be merged for a really good overall protocol proposal.

8 BIBLIOGRAPHY AND REFERENCES

- [1] Wei Ye, John Heidemann and Deborah Estrin, Medium Access Control With Coordinated Adaptive Sleeping for Wireless Sensor Networks. ACM journal.
- [2] Tijs van Dam and Koen Langendoen, An Adaptive Energy-Efficient MAC Protocol for Wireless Sensor Networks. ACM journal.
- [3] Joseph Polastre, Jason Hill and David Culler, Versatile Low Power Media Access for Wireless Sensor Networks. ACM journal.
- [4] Jing Li and Georgios Y. Lazarou, A Bit-Map-Assisted Energy-Efficient MAC Scheme for Wireless Sensor Networks. ACM journal.
- [5] C. Raghavendra and S. Singh, PAMAS - power aware multi-access protocol with signalling for ad hoc networks," Oct. 1997.
- [6] Rajgopal Kannan, Ram Kalidindi and S. S. Iyengar, Energy and Rate based MAC Protocol for Wireless Sensor Networks. ACM journal.
- [7] W. R. Heinzelman and H. B. Anantha Chandrakasan, "Energy-efficient communication protocol for wireless microsensor networks," Jan. 2000.
- [8] Saâd Biaz and Yawen Dai Barowski, "GANGS": Energy Efficient MAC Protocol for Sensor Networks. ACM journal.
- [9] Venkatesh Rajendran, Katia Obraczka and J.J. Garcia-Luna-Aceves, Energy-Efficient, Collision-Free Medium Access Control for Wireless Sensor Networks. ACM journal.
- [10] L.F.W. van Hoesel and P.J.M. Havinga, Poster Abstract: A TDMA based MAC Protocol for WSNs. ACM journal.
- [11] Edgar H. Callaway, Jr. 2003, Wireless Sensor Networks: Architectures and Protocols, CRC Press LLC, United States of America.
- [12] Ahmed Helmy, CAPTURE: location-free Contact-Assisted Power-efficient Query Resolution for sensor networks. ACM journal. p33
- [13] Robert Szewczyk, Eric Osterweil, Joseph Polastre, Michael Hamilton, Alan Mainwaring, and Deborah Estrin, Habitat Monitoring With Sensor Networks. ACM journal.
- [14] Benjamin J. Culpeppera, Lan Dung, Melody Moh, Design and Analysis of Hybrid Indirect Transmissions (HIT) for Data Gathering in Wireless Micro Sensor Networks. ACM journal.
- [15] Alaa Muqattash and Marwan Krunz, CDMA-Based MAC Protocol for Wireless Ad Hoc Networks.
- [16] Chunlong Guo, Lizhi Charlie Zhong, Jan. M. Rabaey, Low Power Distributed MAC for Ad Hoc Sensor Radio Networks.
- [17] Product Specification Sheet of Nordic Semiconductors, 433/868/915MHz RF Transceiver with Embedded 8051 Compatible Microcontroller and 4 Input, 10 Bit ADC. nRF9E5.
- [18] Tian He, Sudha Krishnamurthy, John A. Stankovic, Tarek Abdelzaher, Liqian Luo, Radu Stoleru, Ting Yan, Lin Gu, Energy-Efficient Surveillance System Using Wireless Sensor Networks.
- [19] Luis Alonso, Ramón Ferrús, Ramón Agustí, MAC-PHY Enhancement for 802.11b WLAN Systems via Cross-layering.
- [20] R. E. Kahn, S. A. Gronemeyer, J. Burchfiel, and R. C. Kunzelman. Advances in packet radio technology. Proceedings of the IEEE, 66(11):1468–1496, Nov. 1978.

- [21] Deepak Ganesany, Deborah Estrin, Alec Woo, David Culler, Complex Behavior at Scale: An Experimental Study of Low-Power Wireless Sensor Networks.
- [22] Cintia B. Margi, Katia Obraczka, Instrumenting Network Simulators for Evaluating Energy Consumption in Power-Aware Ad-Hoc Network Protocols.
- [23] Lichun Bao, J.J. Garcia-Luna-Aceves, A New Approach to Channel Access Scheduling for Ad Hoc Networks.
- [24] Rohit Negi, Arjunan Rajeswaran, Physical layer effect on MAC performance in ad-hoc wireless networks.
- [25] Yalin Evren Sagduyu, Anthony Ephremides, Crosslayer Design for Distributed MAC and Network Coding in Wireless Ad Hoc Networks.
- [26] Joon Yoo and Chongkwon Kim, On the Hidden Terminal Problem in Multi-rate Ad Hoc Wireless Networks.
- [27] Dr. Nguyen, www.discera.com/news_articles/news_01-31-02.htm
- [28] Yu-Tso Lin, Tao Wang, and Shey-Shi Lu, Guo-Wei Huang, A 0.5 V 3.1 mW Fully Monolithic OOK Receiver for Wireless Local Area Sensor Network.
- [29] Injong Rhee, Ajit C. Warriar, Lisong Xu, Randomized Dining Philosophers to TDMA Scheduling in Wireless Sensor Networks.
- [30] Matthias Ringwald, Kay Römer, BitMAC: A Deterministic, Collision-Free, and Robust MAC Protocol for Sensor Networks.
- [31] Zhihui Chen, Ashfaq Khokhar, Self Organization and Energy Efficient TDMA MAC Protocol by Wake Up For Wireless Sensor Networks.
- [32] Sinem Coleri Ergen and Pravin Varaiya, TDMA Scheduling Algorithms for Sensor Networks.
- [33] http://en.wikipedia.org/wiki/Four_color_theorem
- [34] Zoran L. J. Basich, and H. Anthony Chan, SrMember, IEEE, Hardware Mailbox and Energy Efficient Scheduling Algorithm in Ad Hoc Wireless Sensor Area Networks.

9 APPENDICES

$$\lambda = c / \text{frequency}$$

9.1. Probability and the Birthday Algorithm

9.1.1. The Birthday Algorithm

In the discrete world of mathematics there is a phenomenon called the “birthday paradox.” It entails finding the probability of two or more people having a birthday on the same day from a group of people. The amazing surprise is that to find out that probability, one only has to find the probability of them all choosing different days. As illustrated below:

Assuming there are 365 days in a year, and there are two people: the first will choose one day out the year and then the second will choose either the same day or a different day. The first has a choice of 365, the second if he chooses a different day has the choice of 364 days and if he chooses the same day he has chosen the same 1 out of 365 as the first. Thus our Equations are:

Probability of two with the same birthday: $(365/365 * 1/365)$

Probability of two with different birthdays: $(365/365 * 364/365)$

Or:

Probability of two with the same birthday: $1 - (365/365 * 364/365)$

Probability of two with different birthdays: $1 - (365/365 * 1/365)$

The Probability of:

Three people with two or more birthdays on the same day:

$$1 - ((365 * 364 * 363) / (365 * 365 * 365))$$

Four people with two or more birthdays on the same day:

$$1 - ((365 * 364 * 363 * 362) / (365 * 365 * 365 * 365))$$

Five people with two or more birthdays on the same day:

$$1 - ((365*364*363*362*361) / (365*365*365*365*365))$$

The official birthday algorithm formula is:

1- (365 to the power (n) / 365 to the power of n) and (n) means “downwards” factorial

This algorithm may be manipulated to suit contention calculation.

9.1.2. Collisions in Contention and the Birthday Algorithm

The Birthday Algorithm is a simple and powerful tool that can be manipulated to find the number of collision in a contention based scheme.

If we just examine the RTS time in SMAC (Figure 2.2) we see that there is a defined length of time a node may send a RTS. That length divided by the time it requires to send a RTS will produce the number of discrete intervals or “days in a year”. The number of contending nodes is the “number of people” there are.

Thus:

If a node’s duty cycle is 1.2 seconds, the RTS slot length is 50mS, the packet size is 10bytes, the radio transmitter transmits at 100kBps and there are 8 nodes in radio connectivity of each other then the probability of collision is:

$$10B / 100kBps = 0.0001 \text{ seconds or } 0.1\text{mS}$$

$$50\text{mS} / 0.1\text{mS} = 500$$

$$1 - ((500*499*498*497*496*495*494*493) / (500 \text{ to the power of } 8)) = 0.00189054$$

This means that in every 100000 RTS’s there will be 189 collisions.

How many RTS’s are there in a day? $60*60*24 / 1.2 = 72000$

That is 136 collisions per day. Each collision costs energy but the energy consumed depends on the number of collisions. There could be only 2 nodes colliding or there could be 4 nodes colliding at this given time and another 2 at another given time all within the RTS time. Thus one needs to find the combinations all the different possibilities of two or more collisions / birthday’s on the same day...

9.1.3. Combinations of the Birthday Algorithm

Given 3 people and using the birthday algorithm we can find the probability that two or more will have a birthday on the same day but what is the probability of exactly two on the same day and what is the probability of exactly three?

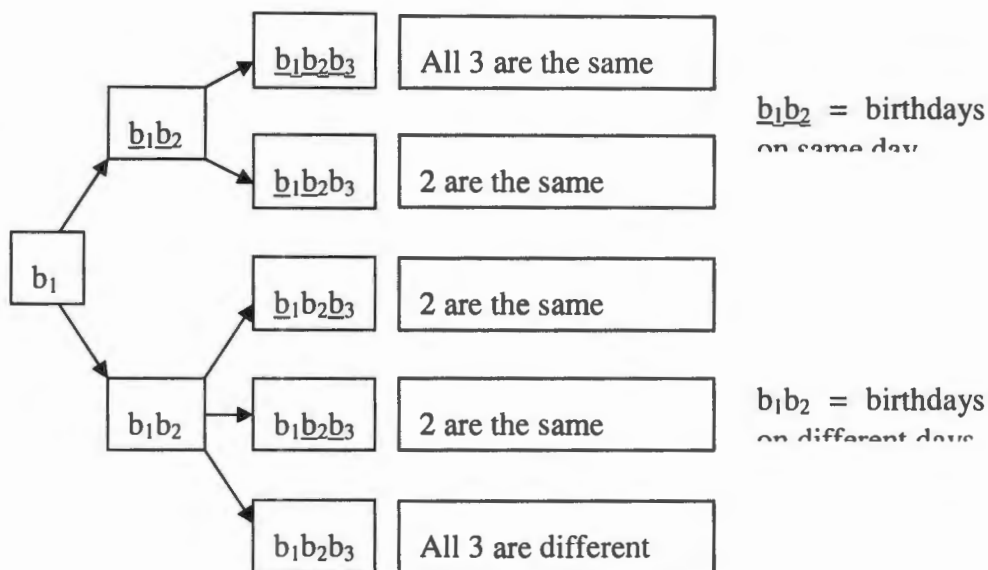


Figure 9.1 The Birthday Algorithm Tree

The figure shows there are 3 different combinations of “2 are the same.” Mathematically we can use “n pick k” to know the number of combinations of there will be. A problem exists where there are 4 or more people. If there are 4 people and 2 choose one day and the other 2 choose the same day too, but not the same day as the first two then the mathematics becomes complicated to say the least. The author could not find the answer to this problem on the internet and if there is a solution what would be its name? So the author coded a program that would work out all the probability for a certain given range.

Solutions to the different combinations of 2 to ‘x’ people:

Same Day Birthdays	Different Day Birthdays	No of iterations
2	0	1
0	2	1

Same BDays	Different BDays	Iterations
------------	-----------------	------------

3	0	1
2	1	3
0	3	1

Same BDays	Same BDays	Diff BDays	Iterations
4		0	1
3		1	4
2	2	0	3
2		2	6
0		4	1

Same BDay	Same BDay	Diff BDay	Iterations
5		0	1
4		1	5
3	2	0	10
3		2	10
2	2	1	15
2		3	10
0		5	1

Same BDay	Same BDay	Same BDay	Diff BDay	Iterations
6			0	1
5			1	6
4	2		0	15

4			2	15
3	3		0	10
3	2		1	60
3			3	20
2	2	2	0	15
2	2		2	45
2			4	15
0			6	1

Same	Same	Same	Diff	Iterations
7			0	1
6			1	7
5	2		0	21
5			2	21
4	3		0	35
4	2		1	105
4			3	35
3	3		1	70
3	2	2	0	105
3	2		2	210
3			4	35
2	2	2	1	105
2	2		3	105
2			5	21

0			7	1
---	--	--	---	---

Etc ...

8,0,1

7,1,8

6 2,0,28

6,2,28

5 3,0,56

5 2,1,168

5,3,56

4 4,0,35

4 3,1,280

4 2 2,0,210

4 2,2,420

4,4,70

3 3 2,0,280

3 3,2,280

3 2 2,1,840

3 2,3,560

3,5,56

2 2 2 2,0,105

2 2 2,2,420

2 2,4,210

2,6,28

0,8,1

9,0,1

8,1,9

7 2,0,36

7,2,36

6 3,0,84

6 2 ,1 ,252
6 ,3 ,84
5 4 ,0 ,126
5 3 ,1 ,504
5 2 2 ,0 ,378
5 2 ,2 ,756
5 ,4 ,126
4 4 ,1 ,315
4 3 2 ,0 ,1260
4 3 ,2 ,1260
4 2 2 ,1 ,1890
4 2 ,3 ,1260
4 ,5 ,126
3 3 3 ,0 ,280
3 3 2 ,1 ,2520
3 3 ,3 ,840
3 2 2 2 ,0 ,1260
3 2 2 ,2 ,3780
3 2 ,4 ,1260
3 ,6 ,84
2 2 2 2 ,1 ,945
2 2 2 ,3 ,1260
2 2 ,5 ,378
2 ,7 ,36
0 ,9 ,1

9.1.4. Code for Combinations

The program consisted of an interface and code:

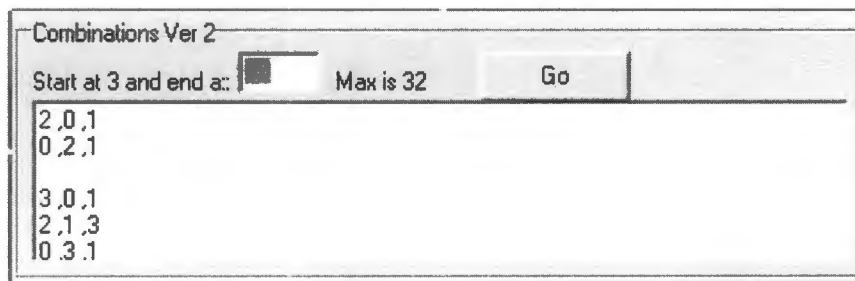


Figure 9.2 The Interface

The code was written in Delphi and consisted of the following lines:

```

beginTime := Time;

for LoopVar := 1 to 1500 do

begin

    NoOfEntries[LoopVar] := 0;

end;

EndInt := StrToInt(Edit11.Text);

GroupSame[1,1] := 2;

GroupSame[1,17] := 0;

NoOfEntries[1] := 1;

GroupSame[2,1] := 0;

GroupSame[2,17] := 2;

NoOfEntries[2] := 1;

startGroupSame := 3;

endGroupSame := 3;

for LoopVar := 3 to EndInt do

begin

    fileName := 'Probability' + IntToStr(LoopVar - 1) + '.txt';

    AssignFile(inputFile,fileName);

    Reset(inputFile);

    fileName := 'Probability' + IntToStr(LoopVar) + '.txt';

    AssignFile(outputFile,fileName);

    Rewrite(outputFile);

    while (Eof(inputFile) <> true) do

        begin

```

```

aChar := 'a';

ReadLn(inputFile,inputStr);

ReadLn(inputFile,tempStr);

count := StrToInt(tempStr);

for innerLoop := 1 to (count + 1) do

begin

    WriteLn(outputFile,(inputStr + aChar));

    if (innerLoop < (count + 1)) then

        begin

            WriteLn(outputFile,count);

        end

    else

        begin

            WriteLn(outputFile,(count + 1));

        end;

        inc(aChar);

    end;

end;

CloseFile(inputFile);

CloseFile(outputFile);

Reset(outputFile);

while (Eof(outputFile) <> true) do

begin

    ReadLn(outputFile,inputStr);

    ReadLn(outputFile,tempStr);

    for anotherLoop := 1 to 68 do

        begin

            tempArrayChar[anotherLoop] := '';

            tempArray[anotherLoop] := 0;

        end;

    for anotherLoop := 69 to 100 do

```

```

begin
  if ((anotherLoop - 68) <= LoopVar) then
    begin
      tempArrayChar[anotherLoop] := inputStr(anotherLoop - 68);
    end
  else
    begin
      tempArrayChar[anotherLoop] := ' ';
    end;
  end;

  //Find the number of numbers with the same numbers e.g. 111122 is 4 numbers the same and 2 numbers the same
  for innerLoop := 1 to LoopVar do
    begin
      for anotherLoop := 1 to LoopVar do
        begin
          if (tempArrayChar[(68 + anotherLoop)] = Chr(innerLoop + 96)) then
            begin
              inc(tempArray[innerLoop]);
            end;
          end;
        end;
      end;
    end;
  LastDigit := 1;
  for anotherLoop := 1 to LoopVar do
    begin
      if (tempArray[anotherLoop] > 1) then
        begin
          tempArray[(32 + LastDigit)] := tempArray[anotherLoop];
          inc(LastDigit);
        end
      else if (tempArray[anotherLoop] = 1) then
        begin
          inc(tempArray[49]); //32 + 17, 32 being the max and 17 being the pos in array where numbers are not the same
        end;
      end;
    end;
  end;

```

```

    tempArray[anotherLoop] := 0;

end;

end;

//Need to do a check upto here
//outputStr := "";
//for anotherLoop := 1 to LoopVar do

//begin

//outputStr := outputStr + IntToStr(tempArray[(32 + anotherLoop)]);

//end;

//outputStr := outputStr + IntToStr(tempArray[49]);

//RichEdit3.Lines.Add(outputStr);

//NOW MUST TAKE THIS AND COMPARE IT TO WHAT'S IN THE GROUPSAME
//ARRAY ALREADY. IF GROUPSAME ARRAY DOESN'T HAVE IT THEN IT MUST BE
//ADDED.

if (startGroupSame = endGroupSame) then

begin

for anotherLoop := 1 to 17 do

begin

GroupSame[endGroupSame,anotherLoop] := tempArray[(32 + anotherLoop)];

end;

inc(NoOfEntries[endGroupSame]);

inc(endGroupSame);

end

else if (startGroupSame = (endGroupSame - 1)) then

begin

for anotherLoop := 1 to 17 do

begin

GroupSame[endGroupSame,anotherLoop] := tempArray[(32 + anotherLoop)];

end;

end;

```

```

inc(NoOfEntries[endGroupSame]);

inc(endGroupSame);

end

else

begin

trueOrFalse := false;

for innerLoop := startGroupSame to (endGroupSame - 1) do

begin

if (trueOrFalse = false) then

begin

for anotherLoop := 1 to 17 do

begin

tempArray[(49 + anotherLoop)] := GroupSame[innerLoop,anotherLoop];

end;

for anotherLoop := 1 to 16 do

begin

trueOrFalse := false;

for yetAnotherLoop := 1 to 16 do

begin

if (tempArray[(32 + anotherLoop)] = tempArray[(49 + yetAnotherLoop)]) then

begin

tempArray[(49 + yetAnotherLoop)] := 0;

trueOrFalse := true;

break;

end;

end;

if (trueOrFalse = false) then

begin

break;

end;

end;

end;

if (trueOrFalse = true) then

```

```

begin
  if (tempArray[49] = tempArray[66]) then
    begin
      trueOrFalse := true;

      inc(NoOfEntries[innerLoop]);
    end
  else
    begin
      trueOrFalse := false;
    end;
  end;
end
else
  begin
    break;
  end;
end;
if (trueOrFalse = false) then
  begin
    for anotherLoop := 1 to 17 do
      begin
        GroupSame[endGroupSame,anotherLoop] := tempArray[(32 + anotherLoop)];
      end;
      inc(NoOfEntries[endGroupSame]);
      inc(endGroupSame);
    end;
  end;
end;
startGroupSame := endGroupSame;
CloseFile(outputFile);
end;

```

```

fileName := 'OutputProb.txt';
AssignFile(outputFile,fileName);
Rewrite(outputFile);
max := 2;
for LoopVar := 1 to (endGroupSame - 1) do
begin
outputStr := "";
for anotherLoop := 1 to 16 do
begin
if (GroupSame[LoopVar,anotherLoop] > 0) then
begin
outputStr := outputStr + IntToStr(GroupSame[LoopVar,anotherLoop]) + ' ';
end;
end;
if (outputStr = "") then
begin
outputStr := '0 ';
end;
if (GroupSame[LoopVar,1] > max) then
begin
max := GroupSame[LoopVar,1];
RichEdit4.Lines.Add(' ');
WriteLn(outputFile,' ');
end;
outputStr := outputStr + ' ' + IntToStr(GroupSame[LoopVar,17]) + ' ';
outputStr := outputStr + ' ' + IntToStr(NoOfEntries[LoopVar]);
RichEdit4.Lines.Add(outputStr);
WriteLn(outputFile,outputStr);
end;
outputStr := ' ';
WriteLn(outputFile,outputStr);
endTime := Time;

```

```
beginTime := endTime - beginTime;
```

```
beginTime := beginTime*24*60;
```

```
WriteLn(outputFile,beginTime);
```

```
CloseFile(outputFile);
```

Energy Efficient Hardware Mailbox and Scheduling Algorithm for the MAC layer in Ad Hoc Wireless Sensor Area Networks

Zoran L. J. Basich, and H. Anthony Chan, *SrMember, IEEE*

Abstract— Ad hoc wireless sensor network's (AHWSN) primary goal is energy efficiency. Theoretically, the most energy efficient MAC protocol for a mesh network would be a TDMA based protocol without the administration, control and exact timing signals that need to be conveyed. This paper presents a hardware solution and a TDMA MAC protocol that is as close to that ideal as possible. The proposal is a low power hardware mailbox that enables nodes to catch messages destined for them while they are asleep. This also allows leeway to the exact timing of the synchronisation needed for a TDMA scheme. In addition the protocol reduces the amount of control messages that needs to be transmitted between nodes and also defines how the nodes need not change their function and role periodically, to coordinate the transmissions of data of other nodes.

Index Terms—AHWSN, WSN, TDMA, CSMA, MAC

I. INTRODUCTION

AN Ad Hoc Wireless Sensor Network (AHWSN) comprises of a number of self-containing electronic devices communicating wirelessly, with the particular purpose of sensing a parameter. Whether it be machine vibrations or temperature changes or many other measurable properties of the environment. The acquired data is then fed to an information sink.

The network is called ad hoc because there is no infrastructure. The devices themselves are the resources of the network, to each other.

The primary goals of AHWSN in order of importance are:

1. Energy efficiency (Prolong the life of the network, thus nodes exercise an awake / asleep duty cycle),
2. Self organising (if individual failure occurs the network still exists),
3. Scalability
4. Reduced cost of each node.

Manuscript received October 31, 2005.

Z. L. J. Basich is a student at the University of Cape Town, South Africa (phone: +2721650-2813; fax: +2721650-3465; e-mail: zoran@skyrove.com).

H. Anthony Chan is professor at the Department of Electrical Engineering, University of Cape Town, Private Bag, Rondebosch 7701, South Africa (e-mail: h.a.chan@ieee.org).

A. The Mailbox

This paper presents a mailbox for the nodes of the network and a TDMA scheduling algorithm for the MAC layer. The mailbox acts as a buffer for the signal being sent from a transmitting node to a node that is asleep. The function of the mailbox is to store detected signals while a node is asleep. The purpose of this new design is that a node could go to sleep i.e. save battery life and still receive other nodes' messages destined for it from other nodes.

The mailbox is hardware that is integrated into each node, eliminating the need of a special node acting as a mailbox relaying the stored messages onto awakening nodes.

An added benefit of the mailbox is that a sending node and a receiving node in a data transfer do not require to be synchronised with each other. Synchronising nodes in an AHWSN is energy consuming.

II. CSMA/CA AND TDMA

MAC protocols for AHWSNs are either:

1. Contention based (CSMA/CA), or
2. Non-contention based (TDMA).

Contention based MAC protocols such as S-MAC [1], T-MAC [2], B-MAC [3], PAMAS [5], sense the channel for activity and if no activity is detected they place a request to use the channel, to which they receive an acknowledgement. If they are granted access to the channel then all other nodes are required to 'keep quiet' during the transmission. Non-contention based MAC protocols such as Bit-Map-Assisted MAC [4], ER-MAC [6], LEACH (with regards to cluster-head transmissions) [7], GANGS [8], TRAMA [9] and E-MAC [10], use a time division scheme to schedule nodes when they may use the channel for transmission.

III. THE TDMA MAC WITH MAILBOXES PROPOSAL

A general TDMA scheme shares a channel in time in round robin form for synchronous data transmission over one medium. This proposal uses the TDMA's time division technique so a node is aware of when it may utilise the channel but the *data transmission is asynchronous*.

One TDMA cycle definition exists across the entire network

- A set number of timeslots.
- The length of the timeslots.
- The beginning of the TDMA cycle

Multiple nodes across the network may use the same timeslot in the TDMA cycle. Only if the radio connectivity of the node using timeslot "x" does not overlap with that of any neighbour also using this same timeslot "x". An overlap is only allowed if there is no node in that common region.

The nodes can assume that the channel is free during their time slot and that no internal network collisions will occur. Bluetooth, frequency division and orthogonal coding are utilised in other networks to reduce collisions but they are highly energy consuming and not suitable for AHWSN, as clearly proven in [11]. Knowing the medium is free ensures that the node need not spend energy sending requests for the medium and require sensing it for current usage. This is energy saved.

Many other MAC protocols [4], [6], [7], [8], [9], [10] enforce the nodes in to clusters. This TDMA MAC with mailboxes *does not form clusters, nor have nodes periodic changing of function and role to a cluster head*. There is a limit to the amount of neighbours a node is allowed to possess thus limiting its transceiver range. The number of timeslots in a TDMA cycle is larger than or equal to the maximum amount of neighbours a node may have.

If any node consumes all of its available energy it will not stop the TDMA scheme from continuously cycling through. This is the only TDMA MAC protocol for AHWSN that can claim such a feature. The nodes will only need to check that their neighbours are there every so many cycles.

A node and its neighbours awake and sleep asynchronously to each other. The node awakes at or near the beginning of its time slot and may go back to sleep when ever it wishes but may only transmit data within its time slot. Only the node whose time slot it is, is awake (all other nodes are asleep) and messages sent to sleeping neighbours are stored in their mailbox.

At the beginning of the timeslot, the node checks its mailbox for any tasks. If there are any, it executes them in that timeslot. Else it goes back to sleep.

Some of the time slots can become contention based, so if there are mobile nodes in the network they can transmit data without asking the current TDMA cycle to configure itself.

The TDMA MAC with mailboxes proposed protocol has two stages:

1. Initialisation
2. Continual execution of the protocol.

"Before order there is chaos". TDMA requires initialisation:

- For nodes to synchronise to each other and,
- the compartmentalising of the channel, and
- Nodes to choose a time slot.

Messages are passed locally only i.e. A node only sends data to its neighbour thus only local synchronisation is required between nodes. 'Hence if each and every node in the network is synchronised with its neighbours then for any given radio range the network is synchronised.' The fact that the ends of network may be out of sync does not affect the message traversing from one side to the other.

The latency of passing a message from one node to another would equal the length of the TDMA cycle. If the TDMA cycle equals the length of the awake/asleep cycle (which is the time taken for a CSMA/CA protocol for AHWSN to send a message) then [13]'s argument that a contention based MAC has a lower latency than a TDMA MAC is not true.

The steps during Initialisation are:

- 1) Discover the neighbours within its radio range.
- 2) Adjust radio range if number of neighbours \geq maximum neighbours allowed. The number of neighbours is limited.
- 3) Synchronise with neighbours. The nodes are required only to synchronise their wake / sleep duty cycle, but not to align bits because data transmission is asynchronous.
- 4) Negotiate the timeslots. Once a node has chosen a timeslot, no other nodes within its radio range may choose that same timeslot.

During the continual execution of the protocol TDMA can guarantee a QoS. The QoS defines the bit rate of transferring data from one node to its neighbour but when viewing the entire network the QoS changes according to where the information is headed. This is due to some nodes being inherently many more hops away from the information sink (the common destination) than other nodes.

In small networks contention based MAC algorithms perform well but scaling up to larger networks the protocol performs very poorly. This is due to the probability of collisions rising rapidly as the number of neighbours of a node rises incrementally¹. The proposed TDMA MAC with mailboxes is for large scale densely packed networks where a TDMA scheme eradicates internal network collisions. 'Studies reveal that energy wastage in existing MAC protocols occur mainly from collision, overhearing, control packet overheads

¹ If α is the number of neighbours a node has plus itself, and β is the time allocated for a node to send a SYNC or a RTS or a CTS, and the exponential $e^{-\alpha/\beta}$, then the probability, converted into a percentage, of a collision

and idle listening.' [6]. A TDMA MAC with mailboxes focuses on minimising all of these energy wastages.

The following Table 1 compares energy consuming factors of CSMA and TDMA with and without a mailbox. The advantages and savings are marked with a star. Table 2 compares other factors of CSMA and TDMA with and without a mailbox. The advantages are marked with a star.

TABLE I
ENERGY CONSUMING FACTORS OF CSMA, TDMA AND TDMA WITH MAILBOX

	CSMA	TDMA	TDMA with Mailbox
Requires initialisation before use:	No*	Yes	Yes
Requires Awake/Asleep synchronisation:	Yes	Yes	No*
Synchronous data transmission:	Yes	Yes	No*
Overhead bits in Asynchronous data:	No*	No*	Yes
Requires co-ordination by "super nodes"?:	No*	Yes	No*
Hidden / Exposed terminal problem:	Yes	No*	No*
Maintains a schedule to avoid collisions:	No	Yes*	Yes*
Sense the medium before transmission:	Yes	No*	No*
Overhead signal for the request and the usage of the channel?:	Yes	No*	No*
Number of *'s highlighting Energy Savings	*** (3)	***** (5)	***** (7)

TABLE II
OTHER ATTRIBUTE COMPARISON OF CSMA, TDMA AND TDMA WITH MAILBOX

	CSMA	TDMA	TDMA with Mailbox
QoS?	No	Yes*	Yes*
Latency? Dependant on ...			
1. Awake/ asleep cycle	1	2	2
2. Length and number of time slots			
Performance in a sparse network?	Good*	Average	Average
Performance in a dense network?	Poor	Good*	Good*
Number of stars:	*(1)	** (2)	** (2)

Thus the TDMA MAC with mailboxes has the greatest number of advantages.

IV. THE HARDWARE DESIGN OF A MAILBOX

The general transmission of data in a network without a mailbox is channel, demodulator then chip. Looking at the mailbox design below in Figure 4.1 'The Mailbox', The Bandpass filter and envelope detector consumes very little energy, less than 10µA, but both are continuously 'on'. The mailbox replaces the demodulator with a low power A/D converter (like an eight level comparator). The A/D is only powered when the envelope detector detects a signal. The maximum and minimum of the A/D is set by the size of the incoming signal and no pre-amplification of the signal is performed. Then the signal is plotted out two-dimensionally. Thus it is saved as an impulse train. Later the chip when it awakes, it can interpret / demodulate the signal via software.

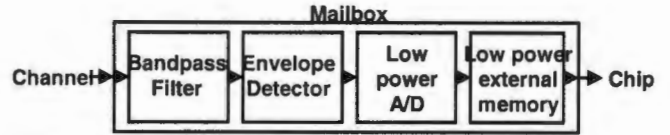


Figure 4.1 The Mailbox.

V. RESULTS

A. Energy Efficiency

A simulator comparing a contention based S-MAC to the above presented TDMA MAC with mailboxes is currently in the coding process but some rough estimate results can be calculated. Using some simple calculations and comparing the two MAC approaches with respect to energy utilisation.

TABLE III
ENERGY CONSUMING ATTRIBUTES

	S-MAC	TDMA with mailbox
Initialisation and synchronisation before use:	Worst case is that the node must send and receive packets that amount to the number of neighbours	Worst case is that the node must send and receive packets that amount to the number of neighbours
(Energy utilised in demodulation in S-MAC is greater than the mailbox's energy consumed)		
Demodulator	Yes	No
Hidden terminal / exposed terminal problem?	Adds to the probability of collisions	No
Collisions? (Within the network)	Yes and the probability may be calculated	No / Almost none
Requires sensing the medium before transmission?	RTS, CTS packets sent back and forth	No
Idle listening? (Waiting for data to come through)	50% - 100% of energy spent during receiving of data [14].	No
Mailbox?	No	Continual energy consumption

Both of these MAC protocols use a random technique to discover their neighbours in initialisation and in the worst case both of them will send the same amount of packets. If a fixed packet length is enforced with each different packet (RTS, CTS, data of S-MAC, data of TDMA MAC with mailboxes) then when comparing these two different MAC protocols, the TDMA MAC with mailboxes will be predominantly more energy efficient in the initialisation stage than the S-MAC protocol due to less packets sent.

The receiving of packets in the two schemes differ and the mailbox utilises less energy in decoding a message than S-MAC. A demodulator may easily use twice the energy than the chip, so moving the demodulation to software can be a great reduction in power consumption.

Collisions consume great deals of energy. If a node's duty cycle is one second then in a day eighty six thousand four

occurring during the SYNC or RTS or CTS stage of the S-MAC awake cycle is: $(1 - (1 - (\alpha / \beta))^n) * 100$

hundred cycles will occur. If there is a small probability of collisions occurring, for example eight percent probability of a collision in each cycle, then clearly a great deal of energy would be wasted in one day's worth of collisions. The hidden terminal problem increases the probability of collisions. S-MAC endures these collisions but TDMA MAC with mailboxes does not.

Packet control overhead such as RTS and CTS in S-MAC is required to be sent each time a node wants to send data. In TDMA MAC with mailboxes there is no such requirement thus when inter-nodal data transmission occur the packet ratio (if all packets are of a fixed length) is three for S-MAC to one for TDMA MAC with mailboxes.

Once a node has received a confirmation to send data in S-MAC, the sender and receiver wait for the other neighbours to go to sleep and then the transfer of data may occur. The idle time where the nodes stay awake and wait wastes energy.

The mailbox continuously utilises energy in the TDMA MAC with mailboxes protocol. The complete circuit of the mailbox does not require to be powered all the time. Only the first two stages: The Bandpass filter and the envelope detector need to be powered constantly. The envelope detector may switch on the A/D and memory when a signal is detected. Thus the permanent power utilised is approximately a mere 10 μ A and when a signal is being stored that increases by the amount that the A/D and memory use.

B. Cost of hardware for mailbox

The Bandpass filter and envelope detector would consist of two resistors, a diode and two capacitors, which hardly amount to any greater cost to the design. The eight level comparator would consist of a few resistors and op-amps and again that would hardly amount to any significant increase in cost. The low power external memory would be a low power flash memory consisting of a few kilo bytes of data. This would make a significant difference in cost, but with bulk production the price would be reduced.

VI. CONCLUSION

A. TDMA MAC with Mailboxes versus S-MAC

The TDMA MAC with mailboxes is more energy efficient than S-MAC in:

- Demodulation
- Amount of packets sent in a data transfer by 3:1
- Having no inter-nodal collisions
- Not having an idle listening stage between two nodes transferring data

The TDMA MAC with mailboxes is less energy efficient than S-MAC in:

- Requiring the mailbox to be powered at all times

B. TDMA MAC with Mailboxes versus other TDMA MACs

The TDMA MAC with mailboxes offers a better TDMA scheme and energy utilisation than other TDMA MACs [4], [6], [7], [8], [9], [10], [14] for AHWSNs due to its unique features:

- It does not require strict timing for the TDMA scheme
- No central point or super nodes co-ordinating the TDMA scheme
- The data transfer is asynchronous
- The nodes with their neighbours do not form clusters
- If any node leaves the TDMA sequence, it does not stop the continuous cycle
- The receiving node does not need to be awake to obtain data

C. TDMA MAC with Mailboxes' cost versus S-MAC's cost

Energy efficiency is the primary goal of AHWSN but its secondary is the cost of each node. With the addition of a mailbox costs are incurred but at the same time the node no longer requires a demodulation circuit (if it was separate to the modulating one) which would reduce costs. The two would not cancel each other out. Further investigation is required for the actual increase in cost per node.

REFERENCES

- [1] Wei Ye, John Heidemann and Deborah Estrin, Medium Access Control With Coordinated Adaptive Sleeping for Wireless Sensor Networks. ACM journal.
- [2] Tijs van Dam and Koen Langendoen, An Adaptive Energy-Efficient MAC Protocol for Wireless Sensor Networks. ACM journal.
- [3] Joseph Polastre, Jason Hill and David Culler, Versatile Low Power Media Access for Wireless Sensor Networks. ACM journal.
- [4] Jing Li and Georgios Y. Lazarou, A Bit-Map-Assisted Energy-Efficient MAC Scheme for Wireless Sensor Networks. ACM journal.
- [5] C. Raghavendra and S. Singh, PAMAS - power aware multi-access protocol with signalling for ad hoc networks," Oct. 1997.
- [6] Rajgopal Kannan, Ram Kalidindi and S. S. Iyengar, Energy and Rate based MAC Protocol for Wireless Sensor Networks. ACM journal.
- [7] W. R. Heinzelman and H. B. Anantha Chandrakasan, "Energy-efficient communication protocol for wireless microsensor networks," Jan. 2000.
- [8] Sa'ad Biaz and Yawen Dai Barowski, "GANGS": Energy Efficient MAC Protocol for Sensor Networks. ACM journal.
- [9] Venkatesh Rajendran, Katia Obraczka and J.J. Garcia-Luna-Aceves, Energy-Efficient, Collision-Free Medium Access Control for Wireless Sensor Networks. ACM journal.
- [10] L.F.W. van Hoesel and P.J.M. Havinga, Poster Abstract: A TDMA based MAC Protocol for WSNs. ACM journal.
- [11] Edgar H. Callaway, Jr. 2003, Wireless Sensor Networks: Architectures and Protocols, CRC Press LLC, United States of America.
- [12] Ahmed Helmy, CAPTURE: location-free Contact-Assisted Power-efficient Query Resolution for sensor networks. ACM journal. p33
- [13] Robert Szewczyk, Eric Osterweil, Joseph Polastre, Michael Hamilton, Alan Mainwaring, and Deborah Estrin, Habitat Monitoring With Sensor Networks. ACM journal.
- [14] Benjamin J. Culpeppera, Lan Dung, Melody Moh: Design and Analysis of Hybrid Indirect Transmissions (HIT) for Data Gathering in Wireless Micro Sensor Networks. ACM journal.