An Experimental Wireless Mesh Network Node Based on AVR ATmega16 Microcontroller and RFM12B Radio Module

Remigiusz Olejnik

West Pomeranian University of Technology, Szczecin
Faculty of Computer Science and Information Systems
ul. Żołnierska 49, 71-210 Szczecin, Poland
r.olejnik@ieee.org

Abstract. The paper describes a simple node designed to act as part of an experimental wireless mesh network. The node is based on Atmel AVR ATmega16 microcontroller and Hope Microelectronics RFM12B radio module. Along with technical details of the node, the network layer (addressing scheme and routing algorithms) is presented. Fundamental principles on wireless mesh networks are also presented. Experimental results summarize the paper and prove limited usability of the project.

Keywords: wireless mesh networks, routing protocol, network layer.

1 Introduction

Wireless mesh networks (WMN) are quite new way of building data communication systems. It's main goal is to achieve higher level of reliability, which is done mainly through special routing protocols. WMNs are still in development stage – there are not enough simple ways of building such networks for masses, due to lack of software support [1]. The field seems to be still in experimental stage. Following that direction, this paper presents an Experimental Wireless Mesh Network (EWMN) node: its hardware and software architecture, including network layer.

The paper consists of three main parts. First part presents basics of wireless mesh networks. Second part describes details of the EWMN node: its hardware and software, including network addressing scheme and routing protocol. Third part shows experimental results. Summary containing final conclusions and possible future works ends the paper.

2 Wireless Mesh Networks' Basics

2.1 Wireless Mesh Networks

Unlike in most common network topologies – bus or star – in mesh topology data is forwarded between the nodes, until it will reach destination, without

A. Kwiecień, P. Gaj, and P. Stera (Eds.): CN 2010, CCIS 79, pp. 96–105, 2010.

help of central node (hub/switch/access point). Such communication is direct and depends on radio range of the nodes – if destination node is not within radio range, it is forwarded according to routing tables. The nodes of mesh networks act as routing devices, which decide about path of data [2]. An example of mesh network is depicted on Fig. 1.

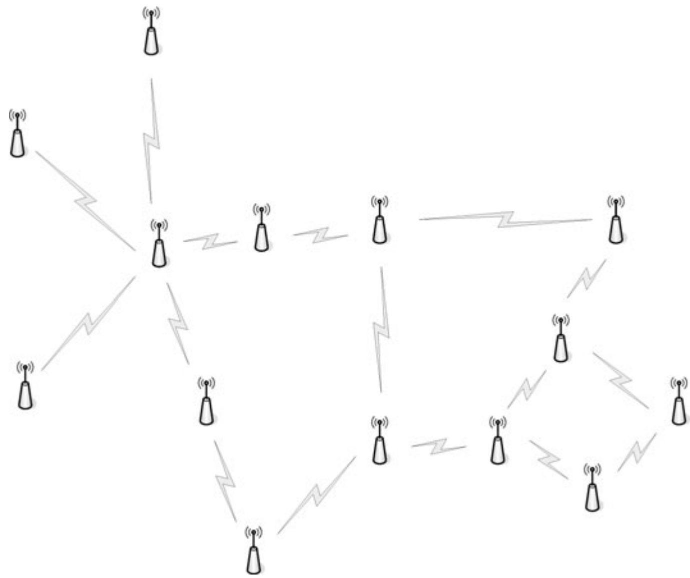

Fig. 1. An example of mesh network

Advantages of mesh networks are numerous: higher redundancy, easy reconfiguration capability, flexibility and easy upgrade possibility. Dynamic reconfiguration and network self-healing may be useful in case of network fail or nodes' location change. The network have to retain its main goal – forwarding data between pairs of source and destination nodes.

2.2 Routing in Wireless Mesh Networks

Performace of wireless mesh networks highly depends on routing techniques. There are many known approaches, that can be classified either as proactive (they maintain paths for every pair of nodes) or reactive (the route paths are discovered in "on demand" way) [2,3]. In this paper one of the modified traditional algorithms is used – B.A.T.M.A.N. (**B**etter **A**pproach **T**o **M**obile **A**dhoc **N**etworking) [4,5] behaviour is inspired by termites, which leave feromonal traces along optimal paths to the food. None of the nodes keep full topological information about the network. Every node has only information about its neighbouring nodes, which could handle data for destination node, associated with some kind of metric. Active neighbouring nodes are discovered periodically by sending HELLO datagram in broadcast manner.

3 An Experimental Wireless Mesh Network Node

3.1 Architecture of the Node

Experimental Wireless Mesh Network (EWMN) has been built of quite simple nodes. The node's architecture is based on cheap and wide known Atmel AVR ATmega16 microcontroller [6] as CPU. Communication between the nodes is provided in one of ISM (industrial, scientific, medical) bands with Hope Microelectronics RFM12B [7] radio module. External communication with PC is done via MAX232 chip and serial port of the computer. Schematic of the node is shown on Fig. 2 and physical design of the node is shown on Fig. 3.

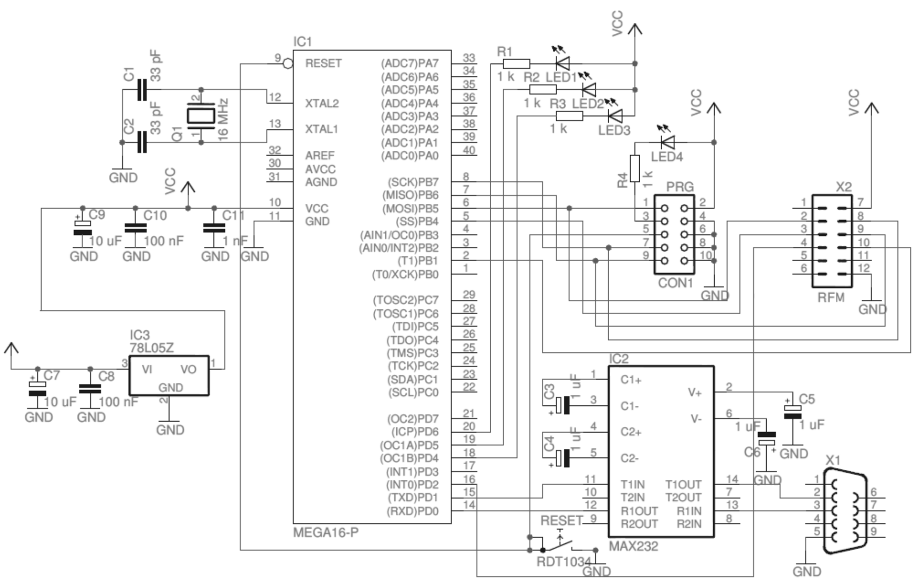

Fig. 2. EWMN node schematic

3.2 Network Layer

Addressing Scheme. Addresses of the nodes have length of one octet. Excluding 0x00 (empty address/address not assigned) and 0xFF (broadcast address) capacity of address pool is 254. The node has random address after initialization. Moreover default destination address value is set to broadcast address 0xFF. Broadcast mode is always local, within radio range of the node, so the datagrams are not forwarded by neighbours to the rest of the network.

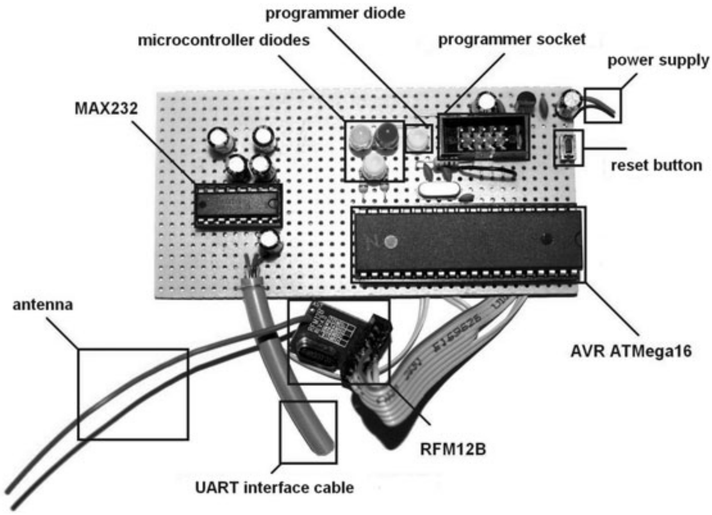

Fig. 3. EWMN node physical design

Datagram Specification. Datagrams have fixed length of 64 bits. The header is 24 bits long and data part is 40 bits long. Structure of the datagram is depicted on Fig. 4.

type	src address	dst address	data	data	data	data	data

Fig. 4. Datagram structure

The header consists of three fields, each of 8 bits length:

– **type** – with value `HELLO` for broadcast datagram or `REGULAR` in case of ordinary data;
– **source address** – with correct value in range `0x01`–`0xFE`;
– **destination address** – with correct value in range `0x01`–`0xFF`.

Broadcast datagram `HELLO` uses two data fields carrying originator address and sequential number. Three remaining data fields are unusued at this time. Structure of broadcast datagram is depicted on Fig. 5.

HELLO	src address	dst address	originator	seq no.	*	*	*

Fig. 5. `HELLO` datagram structure

Routing Algorithm. Routing algorithm is based on the B.A.T.M.A.N. algorithm [4,5] and implements addressing, splitting data into datagrams, broadcasting and data forwarding between the nodes to provide routing routines. Key concept of the B.A.T.M.A.N. algorithm is to inform other nodes of node's presence in cyclical manner. `HELLO` datagrams allow building a routing table at the level of every node. Creating new datagrams is rather simple, but decision mechanism used after datagram's arrival is more complex. That algorithm is shown on Fig. 6.

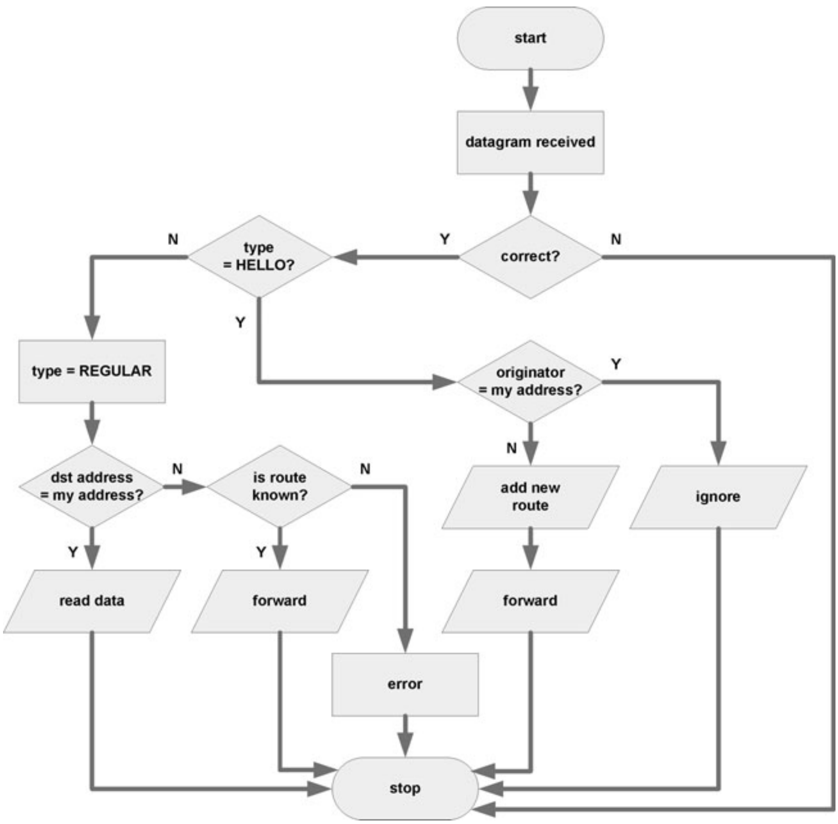

Fig. 6. Decision algorithm after datagram's arrival

At first, datagram type is checked against its correctness. If type is not valid (`HELLO` or `REGULAR`) it is discarded. In case of `REGULAR` datagram, destination address is checked. If it is equal to the node's own address, data fields are used by higher layer. If the datagram is not directed to the node, the routing table is checked to find a route to destination node. If the route is succesfully found, the datagram is forwarded using that route. In case of `HELLO` datagram, originator address is compared with the node's own address – if they are equal, datagram is ignored because it came back to its source; if originator address is not equal to the node's own address, new route (or update) is added to the routing table and datagram is forwarded do all of the neighbouring nodes.

Routing Table. Routing table consists of four fields: destination address, intermediate node address, metric (number of hops) and sequential number. Every node knows only directions (neighbouring nodes), which could be used as routing paths. Data will be forwarded through them until it will reach its destination. The metric is used to find most reliable path and sequential number is used when updating routing tables. Example of routing table is presented in Table 1.

Table 1. Example of routing table

Destination	Direction	Metric	Seq No.
0x4C	0x4C	1	17
0x2A	0x3C	2	6
0x5D	0xFA	5	95
0xD4	0x12	3	25
0xD4	0xFA	2	104

Routing Table Update. Routing table update is done in two stages. First stage is preparing temporary table with new routes. During second stage, synchronously with sending `HELLO` datagrams, new routes are added to routing table. Algorithm of preparing new routes is presented on Fig. 7.

3.3 Program Architecture

Program code of the node has been divided into functional blocks (Fig. 8):

- **main**: main part of the program;
- **BATMAN**: routing algorithm implementation with routing tables and mechanisms;
- **RFM12B**: radio module's routines;
- **RS232**: serial interface communication;
- **datagram**: datagram coding and decoding;
- **tools**: auxiliary routines.

3.4 Terminal Interface – Communication with PC

The node is connected to external computer using node's RS232 MAX232 interface with 9600 bps 8N1 settings. Eight terminal based commands are hidden in ATMega16 program and they are combined of ~ and abbreviated commands:

- (**a**)ddress of the node (view/change),
- (**d**)estination address (view/change),
- (**r**)outing table (view),

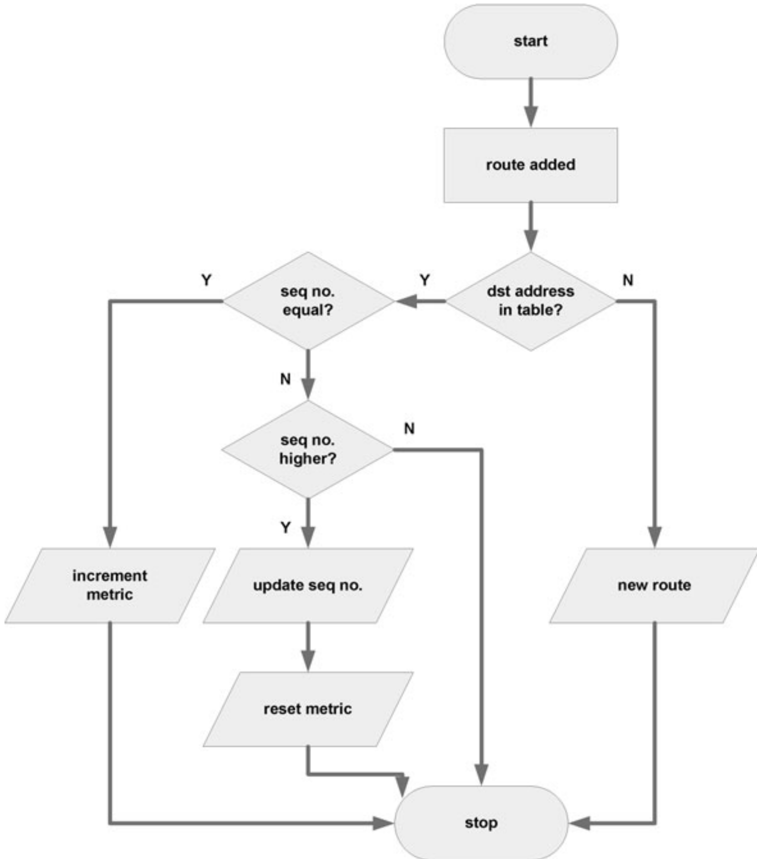

Fig. 7. Algorithm of preparing new routes for routing table

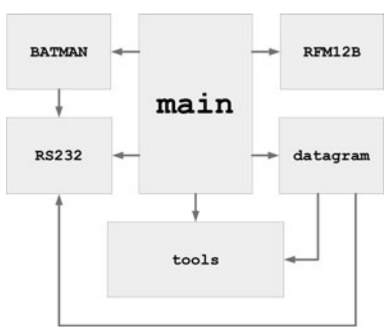

Fig. 8. Functional blocks of the node's program code

– (**n**)ode state toggle (Fully Online, Locally Online, Offline),
– (**p**)ower of TX change (min/med/max),
– (**v**)erbose mode on/off toggle,
– (**c**)lear terminal screen,
– (**e**)cho on/off toggle.

Typical information in verbose mode are as follows:

```
# Hello msg from: 0x71; origin: 0x71; seq num: 0x1B
# New Route to 0x71 via 0x71
# Sending Hello
# Regular datagram from 0x71
# Data field: dzien
# Regular datagram from 0x71
# Data field: dobry
# Sending Hello
# Hello msg from: 0x71; origin: 0x47; seq num: 0x1C
# My own hello returned
```

4 Experimental Results

After initial design of Experimental Wireless Mesh Network hardware and software, some preliminary experiments have been conducted.

At first, simple test of datagram transmission has been made. 10 datagrams has been sent 10 times to find possible data loss ratio for one node working in Fully Online state. Results are shown on Fig. 9.

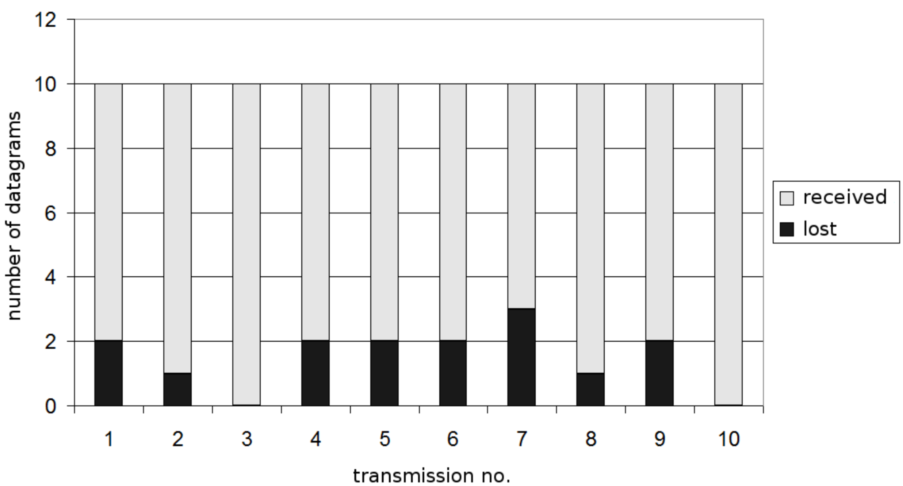

Fig. 9. Lost datagrams

Unfortunatelly, data loss ratio is rather high, about 15 %, so it can be said that data integrity is not assured at this time. More nodes mean more hops and possible multiplication of data loss ratio. Probable reason of such high datagram loss ratio is long main program loop and insufficient length of FIFO queue of RFM12B radio module (only 16 bits long). When the node had been in Locally Online state, data loss ratio was measured at 1 % level. Possible solution for that issue is redesign of main program loop and/or add network layer that would facilitate communication between nodes.

Theoretical network range, that can be computed using Friis equation [8], has been also confirmed. Theoretically the range is less than 150 m in free space [9]. It has been practically proven than network's capability of proper communication is kept at distances 120–130 m, even in office environment, without direct line-of-sight requirement met and with very simple antennas, supplied by radio modules manufacturer.

5 Summary

Presented Experimental Wireless Mesh Network node has been successfully deployed as hardware and software design using Atmel AVR ATMega16 microcontroller and Hope Microelectronics RFM12B radio module. The paper presents details on the design, especially network layer and other network-related issues. Experimental results prove limited usability of that system, although general specific aspects of wireless mesh networks (such as selfconfiguring capability and auto-updating of routing tables) and regular data transmission between nodes has been shown.

Possible enhancements could be done with:

– redesigning of main loop of the node program,
– adding external RAM allowing bigger routing tables (1 KB SRAM yields limit of 32 records now),
– longer addresses allowing bigger network (only 254 usable addresses now),
– adding local keyboard and display to the nodes allowing direct text messages exchange,
– changing flooding algorithm (used for `HELLO` datagrams propagation) into edge forwarding algorithm,
– using more efficient antennas helping radio communication reliability,
– increasing network throughput (RFM12B modules allow maximum throughput 115.2 kbps),
– designing higher layers, especially network layer, helping datagram loose problems.

Acknowledgments. I would like to thank my graduate student, Marek Korniowski, who developed my initial idea of the network as his master dissertation [10]. Parts of his work are used here with author's permission.

References

1. Olejnik, R.: Przegląd rozwiązań sprzętowo-programowych pozwalających na budowę bezprzewodowych sieci o topologii kratowej (in Polish). Metody Informatyki Stosowanej 1(19), 69–74, Komisja Informatyki PAN, Szczecin (2009)
2. Zhang, Y., Luo, J., Hu, H. (eds.): Wireless Mesh Networking: Architectures, Protocols and Standards. Auerbach Publications, Boca Raton (2007)
3. Hossain, E., Leung, K. (eds.): Wireless Mesh Networks: Architectures and Protocols. Springer, London (2008)
4. BATMAN routing protocol, http://www.open-mesh.net/
5. Johnson, D., Ntlatlapa, N., Aichele, C.: A simple pragmatic approach to mesh routing using BATMAN. In: 2nd IFIP International Symposium on Wireless Communications and Information Technology in Developing Countries (WCITD 2008), Council for Scientific and Industrial Research, Pretoria (2008)
6. Atmel AVR ATmega16 datasheet, http://www.atmel.com/dyn/resources/prod_documents/doc2466.pdf
7. Hope Microelectronics wireless FSK transceiver module RFM12B, http://www.hoperf.com/rf_fsk/rfm12b.htm
8. Golio, M. (ed.): The RF and Microwave Handbook. CRC Press, Boca Raton (2001)
9. Olejnik, R.: A concept of emergency communication system based on ISM bands. In: Węgrzyn, S., Czachórski, T., Kwiecień, A. (eds.) Contemporary Aspects of Computer Networks, vol. II, pp. 275–280. Wydawnictwa Komunikacji i Łączności, Warszawa (2008)
10. Korniowski, M.: A concept of a failsafe computer network based on the licence-free bands (in Polish). Master thesis, West Pomeranian University of Technology, Szczecin (2009)