UNIVERSITY OF VAASA

FACULTY OF TECHNOLOGY

TELECOMMUNICATION ENGINEERING

Niklas Wik

WEATHERLAN – A LOCAL AREA NETWORK FOR MONITORING AND CONTROL

Master's thesis for the degree of Master of Science in Technology submitted for inspection in Vaasa 12th of October 2009.

Supervisor	D. Sc. (Tech) Mohammed Elmusrati
Instructor	M. Sc (Tech) Reino Virrankoski

ACKNOWLEDGMENTS

First I will thank Mohammed Elmusrati and Reino Virrankoski for their work and support leading me to the completion of this Master's Thesis.

I am grateful for Lab. Engineer Veli-Matti Eskonen and Lab. Engineer Juha Miettinen for their support with necessary tools to finish this thesis.

A special thanks goes to Lab. Engineer Jani Ahvonen for valuable help with the work related to electronics.

TABLE OF CONTENTS

SYMBOLS AND ABBREVIATIONS

3G	Third Generation
3GPP	The 3rd Generation Partnership Project
AES	Advanced Encryption Standard
API	Application Programming Interface
APN	Access Point Name
BSC	Base Station Controller
BSD	Berkley Software Distribution
BSS	Base Station Subsystem
BSSGP	Base Station Subsystem Gateway Protocol
BTS	Base Station Tranceiver
CPU	Central Processing Unit
CSMA-CA	Carrier Sense Multiple Access – Collistion Avoidance
CTS	Clear To Send
DC	Direct Current
DES	Data Encryption Standard
DS	Differentiated Services
ECN	Explicit Congestion Notification
EIR	Equipment Identity Register
FFD	Full Function Device
FMI	Finnish Institute of Meteorology
GGSN	Gateway GPRS Support Node
GPIO	General Purpose Input/Output
GPRS	General Packet Radio Service
GSM	Global System for Mobile communication
GTP	GPRS Tunneling Protocol
HLR	Home Location Register
IC	Integrated Circuit
ICMP	Internet Control Message Protocol
IEEE	Institute of Electrical and Electronics Engineers
IETF	Internet Engineering Task Force
IHL	Internet Header Length

IMEI	International Mobile Equipment Identity
IP	Internet Protocol
IPC	Inter Process Communication
IPsec	Internet Protocol security
ISO	International Organization for Standardization
ITU-T	International Telecommunications Union – Telecommunications sector
MAC	Medium Access Control
MD5	Message-Digest algorithm 5
ME	Mobile Equipment
MS	Mobile Station
MSC	Mobile Switching Center
MT	Mobile Termination
NSAPI	Network Service Access Point Identifier
O-QPSK	Offset-Quadrature Phase Shift Keying
OS	Operating System
OSI	Open Systems Interconnection
PAN	Personal Area Network
PDN	Packet Data Network
PDP	Packet Data Protocol
PDU	Protocol Data Unit
PHY	Physical
QoS	Quality of Service
RF	Radio Frequency
RFC	Request For Comments
RFD	Reduced Function Device
RTS	Ready To Send
SAP	Service Access Point
SICS	Swedish Institute of Computer Science
SIM	Subscriber Identity Module
SGSN	Serving GPRS Support Node
SMC	Surface Mounted Component
SoC	System-on-Chip

SPI	Serial Peripheral Interface
SSL	Socket Secure Layer
TCP	Transport Control Protocol
TE	Terminal Equipment
UART	Universal Asynchronous Receive and Transmit
UDP	User Datagram Protocol
UE	User Equipment
WPAN	Wireless Personal Area Netowork
WSN	Wireless Sensor Network

UNIVERSITY OF VAASA
Faculty of technology
Author: Niklas Wik
Topic of the Thesis: WeatherLAN – A local area network for monitoring and control
Supervisor: Mohammed Elmusrati
Instructor: Reino Virrankoski
Degree: Master of Science in Technology
Major of Subject: Telecommunications Engineering
Year of Entering the University: 2005
Year of Completing the Thesis: 2009 **Pages:** 64

ABSTRACT

Monitoring of nature behaviours is a crucial part in many applications. The need for monitoring is in fact unavoidable in systems where independent operation of a system is needed. On locations where no cabled infrastructure is available it is necessary to use wireless link to interconnect the location with the Internet. GPRS is a cheap solution for transferring data over such areas where cables are not available by using operator public cellular network.

In this thesis a wireless sensor network is integrated with a GPRS module to support multiple measurement points and GPRS link as backbone connection to remote location. Security issues related to embedded systems and the use of public networks is investigated and one possible solution presented. Vaisala WXT520 weather transmitter is added to the system to measure the weather at the network location which would be needed to remotely support the distributed energy production by wind turbine generator, solar panels and backup diesel generator. The system prevent to be one solution that would enable remote control of the local energy production.

KEYWORDS: WSN, 802.15.4, GPRS

1. INTRODUCTION

For a long time the energy production have consisted of large centralized production units. The discussion about global warming have rapidly pushed research into reusable energy forms, with decentralized energy production such as wind turbine generators and solar panels. However, these kinds of energy production technologies require more and more automation to make them operate robustly. The use of such renewable energy sources as solar and wind also implies that further knowledge is needed about the surrounding weather on the production site. Additionally small scale energy production units should work automatically without constant human intervention.

There exist several remote weather monitoring applications. One outstanding example is Helsinki Test Bed maintained by Finnish Meteorological Institute (FMI) and Vaisala. Vaisala WXT weather transmitters are used in the test bed to measure weather phenomenons and collect them to a database. Long haul connections are made with cellular network, since most of the weather transmitters are placed at cellular network base stations (FMI, 2009).

In this thesis a wireless sensor network with control application have been investigated and tested for remote monitoring and control of energy production. Thus, a regular solution, in which communication is done with one module is extended by introducing a wireless sensor network (WSN), that may collect different kinds of data. Since the sensor nodes are equipped with microcontrollers and memory, data sensed by the nodes and by the weather monitoring station can be processed locally. Only that part of the information which is needed in the centralized control will be transmitted over the long distance link.

In this Master's Thesis work the main goal was to design and build an experimental set up which was deployed at Meteoriihi museum and observatory located at Söderfjärden, Vaasa. The system targets to integrate a Vaisala WXT520 weather transmitter with a WSN, which will be further supported by a general packet radio service (GPRS) module. The module gives access to Internet for two-way communication. A server applica-

tion which converts the measured and transmitted data to database and support software for system remote control.

The rest of the thesis is organized such that first different network protocols, their functionality and integration with each other is investigated. Then data aggregation and the methods how to exploit the measured data are analysed. After the theoretical introduction detailed specifications of the developed system are given. Finally, we describe the experimental set up, explain the results and point some directions for future work.

2. NETWORK INTEGRATION

Nowadays there are many different network technologies in use. Therefore a good understanding of each one of them is necessary to be able to integrate different communication systems with each other. In this chapter different communication systems that are used in this thesis work are discussed in theoretical point of view.

The OSI-model (Open System Interconnection) defined by ISO (International Organisation for Standardization) will work as a guideline in this chapter. OSI model is a seven layer stack, where each layer can be independently developed. The interfaces between layers should be well defined to support the principle that each layer can be developed independently (Stallings 2007: 42-45).

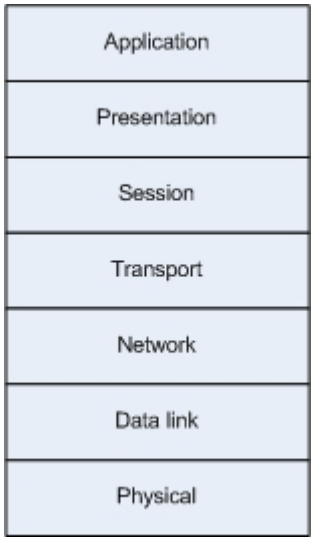


Figure 1. OSI-model structure.

2.1. Internet Communication

In practice, the OSI model works as a reference and the de facto standard protocol stack which is used to communicate over the internet is the Transport Control Protocol (TCP) on top of the Internet Protocol (IP). This combination of TCP/IP is usually called TCP/IP protocol suite, which is a modification of the OSI model structure (Stallings,

2007:34-38). IP and TCP were defined and published at the same time in IETF RFC documents 791 and 793, in September 1981 (IETF RFC 791. 1981).

For comparison with the OSI model an overview of TCP/IP protocol suite is given in Figure 2. In TCP/IP suite the three highest layers are all stubbed into one application layer. As a result, a programmers interface to the protocol stack will be with the transport layer. In TCP/IP protocol suite the programming interface is done by sockets. A socket is defined as a pair of TCP or user datagram protocol (UDP) port and an IP address. Examples in transport layer are TCP and UDP , in the internet layer IP and Internet Control Message Protocol (ICMP) are common protocols (Stallings 2007:34-42).

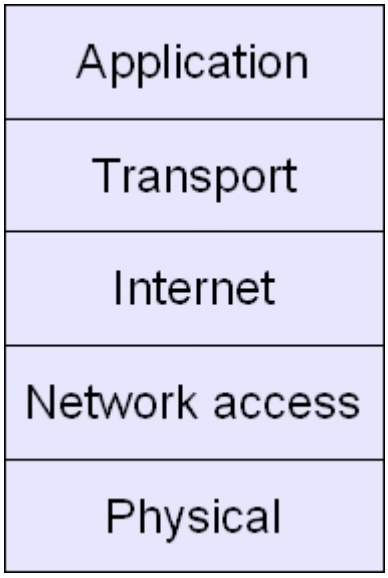


Figure 2. TCP/IP protocol suite.

2.1.1. TCP/IP Protocol Suite

The IP protocol header structure for version 4 is presented in Figure 3. As shown in the figure, a minimum length of the IPv4 header is 20 octets. The first 32-bit word Starting with an integer of the version number, which is naturally 4 in IPv4, continuing with Internet header length (IHL), which specifies the length of the header including options

and paddings field. The differentiated services (DS) 6-bit field provides a classification how packets in the network should be forwarded, with a default value of best-efford. Further explanation of the DS field is given in RFC 2474 published by IETF. Explicit Congestion Notification field is still under development and are not necessarily used. A field total length specifies the length of the complete PDU (Stallings 2007: 578, 636-641).

In the second word, an identification field is used to uniquely identify a specific IP PDU in the Internet, and its living time is specified in the third word. The time to live field measures the time a packet can travel the Internet in seconds, although since every router must decrease the value with one it may be used as a hop count as well. A fragment field is necessary, since the PDU may travel in such networks that do not support the original or intermediate sizes of the packets. The fragment field specifies an offset in multiples of 64-bit units. One can already see from this that a maximum data payload can not exceed a specified size, which is 65535 octets for IPv4, including the headers. (Stallings 2007: 578-579), (Postel 1981)

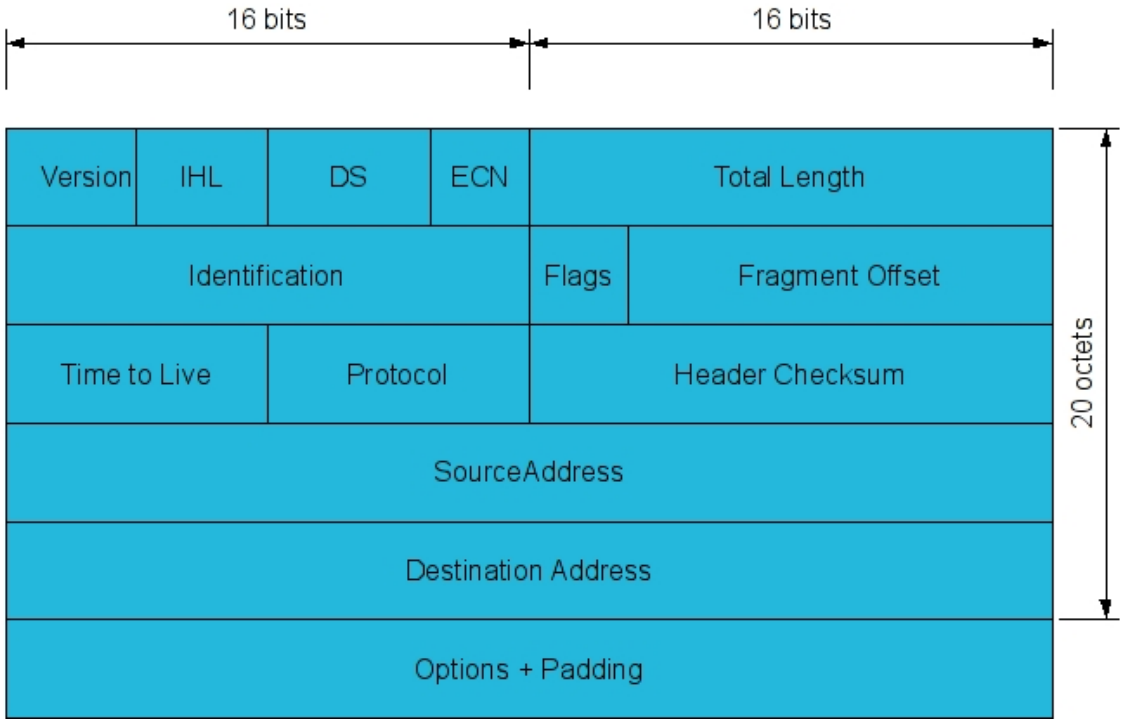
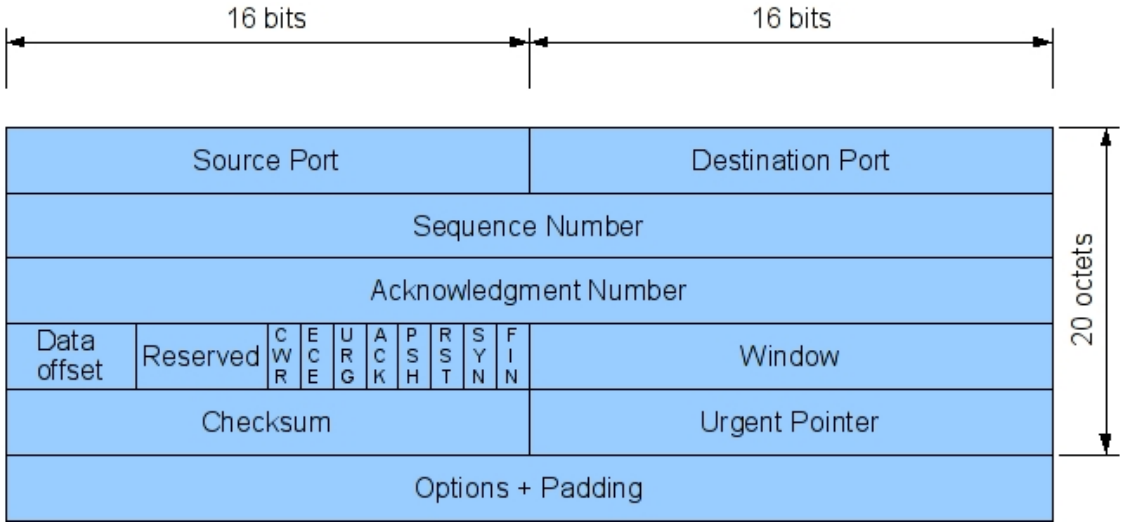


Figure 3. IPv4 protocol header.

If IP protocol is applied it is necessary that all Internet hosts participating to the network have an IP module. In the case of network routers the module shall participate to the routing decisions so that a packet reaches its destination. The routing decisions are made with addresses specified in the protocol header. There are two standards available, in the Internet today, internet protocol version 4 (IPv4) and internet protocol version 6 (IPv6). (Stallings 2007: 39-40)

2.1.2. TCP Protocol

The TCP header format is illustrated in Figure 4. TCP provides reliable communication , since it is a connection-oriented protocol. In other words, a connection between two hosts is established by a three way handshake. In order to establish a connection one of the hosts must be in listen state or a simultaneous SYN must be sent, otherwise the connection is not established. In the client-server architecture the normal procedure is that a server sets up a socket in listen state. In the Unix environment this is done by the socket-API, where a socket is first requested from the kernel, then bind to a specific port and set into listen state. A client may then initiate a connection to the server by setting the SYN flag and the destination port equal to the servers listen port. Upon reception of the SYN the connection is moved to SYN received state, where the listening application must call accept to acknowledge and finalize the three way handshake. When the handshake is completed data may be exchanged between hosts. (Postel 1981b)



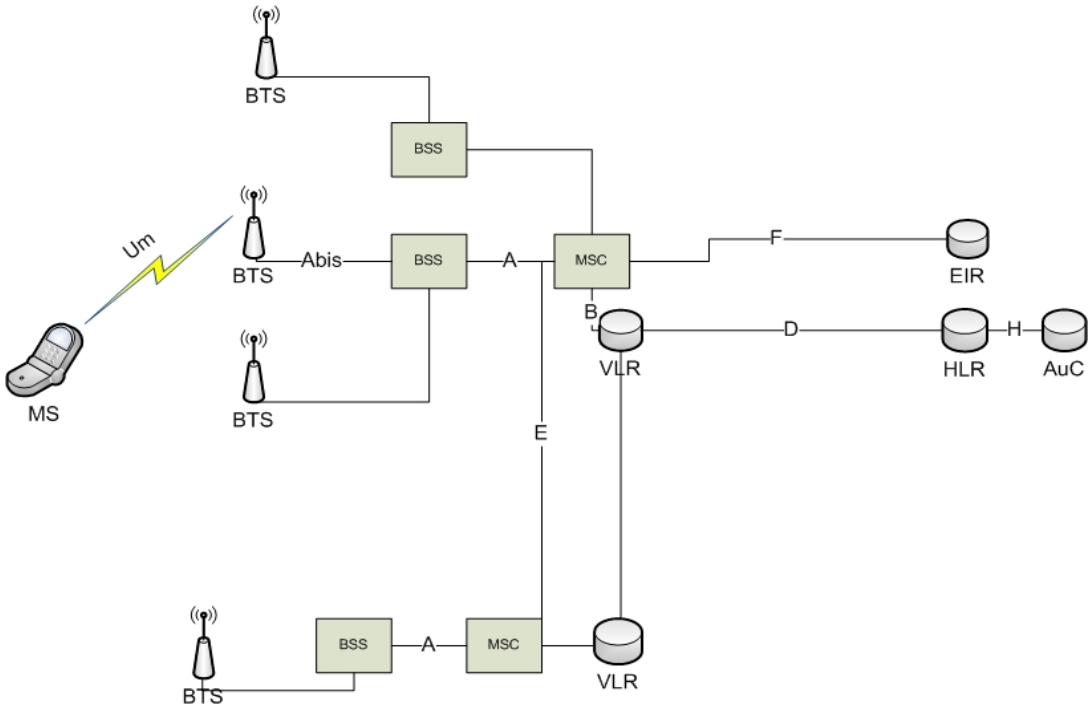
2.2. Cellular Network

Cellular networks are widely spread today. In Finland, existing cellular networks are built according to the Global System for Mobile communication (GSM) and Third Generation (3G) cellular network standards.

In this thesis the part of the GSM network applied to create the long distance link is GPRS. Therefore only some limited things of the GSM standard are covered to understand GPRS. 3G network is not discussed further, although 3G network services may be used in future work based on the work done in this thesis.

2.2.1. Network Architecture

A schematic of the general GSM network is presented in Figure 5. A Mobile Station (MS) consists of the Mobile Equipment (ME) and the Subscriber Identity Module (SIM) (Eberspächer, Vögel & Bettstetter 2001:35-36).



The Mobile Station (MS) is communicating with the Base Station Transceiver (BTS) over the Um interface. The ME is further divided into blocks of a Mobile Termination

(MT) and a terminal equipment (TE), where the MT is performing all functions related to GSM standard. In GSM the radio interface is a combined TDMA/FDMA (Time Division Multiple Access)/(Frequency Division Multiple Access), with eight time slots mapped onto 200kHz frequency bands (Eberspächer et. 2001:63-73, 209).

The core network of GSM offers a single data channel with data rates of 300-9600 bit/s for data services and 13kbit/s for voice. A MS may employ any higher layer protocol on top of the GSM protocol stack. Due to the complexity of the GSM network a detailed explanation can not be given in the context of this thesis, readers should refer to other sources for more information about GSM, see for example (Eberspächer et. 2001).

2.2.2. GPRS

General Packet Radio Service (GPRS) is an evolution of the GSM standard, although it uses the same network with two nodes added, Serving GPRS Support Node (SGSN) and Gateway GPRS Support Node (GGSN). GPRS adds the support to use packet switched communication in the GSM network, without occupying a complete channel at all the time, instead the channel is released when not used and other MSs may use it. Therefore, it is possible to interact with other packet switched networks with a MS node, using the concept of client-server technology with reduced costs. In comparison, if GSM data call is made, it occupies a channel during the whole connection even though no data is sent (Seurre, Savelli & Pietri, 2003).

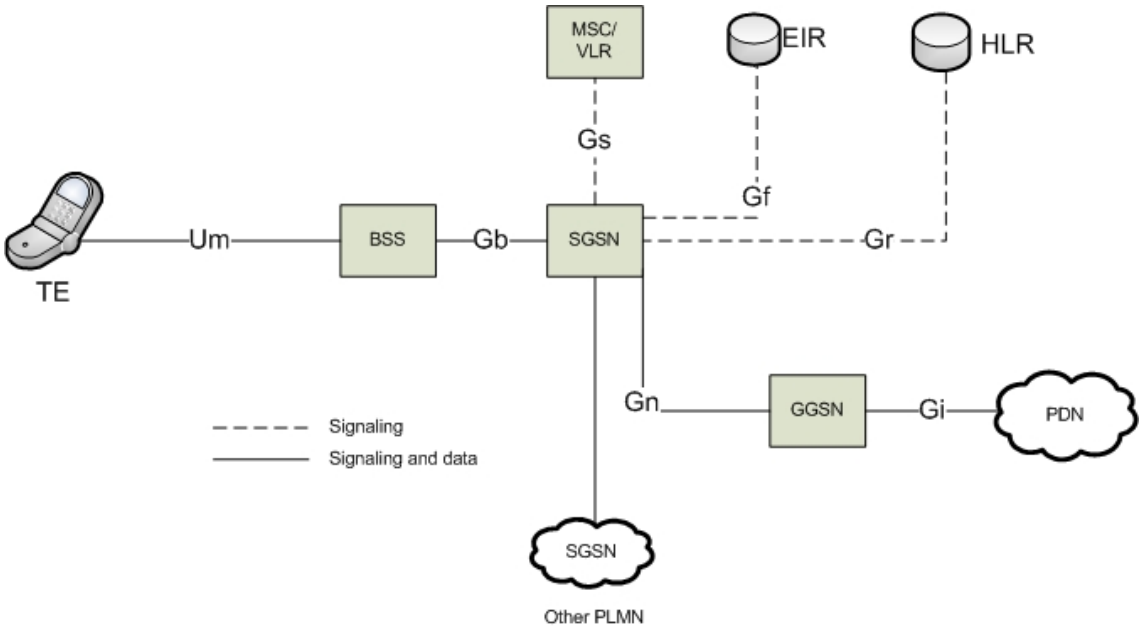


Figure 6. GPRS core network.

There are two new nodes added to support GPRS: Serving GPRS Support Node (SGSN) and Gateway GPRS Support Node (GGSN), see Figure 6. Additionally there are new logical interfaces added, all named with first letter G. Original GSM Base Station Controller (BSC) is connected with the GPRS core network over Gb interface. From the SGSN there are logical interfaces to mobile switching center (MSC), equipment identity register (EIR) and home location register (HLR). These logical connections handle the information flow needed to access the MS. (Seurre, et 2003:61-64)

The GGSN is responsible for connections to outgoing PDNs (Packet Data Network). Thus, it forwards incoming and outgoing packets between the SGSN and the used PDN. Figure 7 illustrates the GPRS transmission protocols.

Inside the GPRS core network, several different protocols are used to enable the transparency for the MS, such that it appears as it has direct connection to a PDN, for example towards the Internet. An application that communicates with the Internet proceeds as follows. An IP packet is generated from the application data, this one is handled by the sub-network dependent convergence protocol (SNDCP), which is the direct access between MS and SGSN. From the SNDCP the packet is further segmented into logical link control (LLC) frames, in which the radio link control (RLC) and MAC

layer blocks are made which and then mapped onto physical channel for transmission over Um interface. The role of SNDCP is to deliver network layer payload from the MS to the SGSN and vice versa (Eberspächer et. 2001: 252-255).

BSS GPRS (BSSGP) application protocol is defined between the base station subsystem (BSS) and SGSN. The purpose of BSSGP is that it take care of the MS mobility. Also other information, such as quality of service (QoS) information is transferred over this protocol. User payload from the MS is unpacked from the RLC frame and inserted into BSSGP payload and transferred to the SGSN (Eberspächer, ec 2001:256-257).

At the SGSN the MS packet is unpacked all the way to SNDCP, see Figure 7. The SGSN performs an address translation from the IP packet originating from the MS, where the IP source address of the MS is translated into a network service access point identifier (NSAPI) and temporary link logical identifier (TLLI) pair which is unique for that part of the core network. Therefore, a MS can be uniquely identified inside the network. From the SGSN the MS IP payload is further encapsulated into a GPRS tunnelling protocol (GTP) frame which is transferred over an UDP/IP connection inside the core network to the GGSN. At the GGSN the IP packet from MS is unpacked and sent to the PDN network over usual TCP/UDP over IP as described previously. A packet originating to the MS travels the same path in the core network until it reaches the MS. An incoming packet for the MS is always routed to the GGSN, since it is configured with an IP address that has the same network prefix as the MS IP address. In GGSN a translation is made to TLLI, sent to SGSN which identifies the MS and forwards the message further to the MS by using BSS. (Eberspächer, etc. 2001)

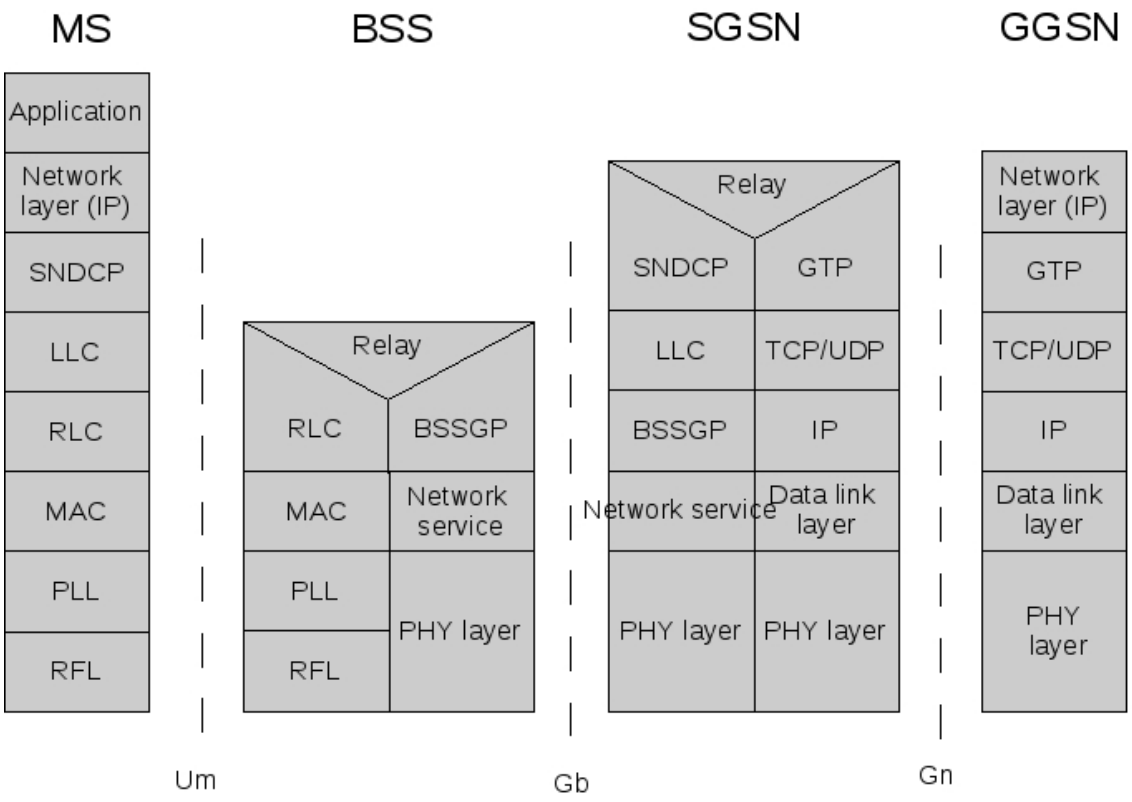


Figure 7. GPRS core network transport.

A GPRS connection in packet mode is specified by a PDP (Packet Data Protocol) context. The PDP context contains information about the APN (Access Point Name), which is used by the GGSN serving the connection. NSAPI and LLC service access point identifier, specifies additional information to handle the traffic flow inside the GPRS network. PDP address is the MS address for the PDP connection. QoS, which is negotiated with the network, is specified in the GPRS standard. Radio priority is used by the MS to specify the order of transmission in lower layers. Protocol configuration options provide additional information about the external data protocol used (Seurre, et 2003:338-339).

The PDP address and QoS are important in the user's point of view. The PDP address specifies the address of the MS for that PDP connection, which is IP address in case of IP based networking. A PDP context can be initiated by both the network and the MS. The command issued to the MT to configure a PDP context by applications is AT+CGDCONT, where at least the type of connection and APN is specified. There is also a

possibility to set the PDP address space and compression flags, where the compression type is the V.42 bis. After a +CGDCONT command the connection may be set online with AT+CGATT, which will try to perform a combined IMSI attach request for the MT from the network (Seurre, et 2003:50-56), (3GPP, 1999).

2.2.3. GPRS Operation

The MSs supporting GPRS are divided into three different classes; A,B and C. Class A devices can simultaneously have circuit- and packet-switched connections ongoing, whereas class B can detect both circuit- and packet-switched connections in the idle mode, but do not support ongoing transmission in both modes simultaneously. The last class C, supports one of the two, circuit- or packet-switched mode. (Seurre, et 2003:49)

One concept of the GPRS operation that differs from original GSM is the use of multis-lot classes. This adds the possibility for a MS to transmit and receive on several TDMA slots, within one TDMA frame. There are some restrictions, such that the MS must also be able to measure adjacent cells. Therefore some delays have been defined such that the MS have enough time between adjacent transmissions and receptions (Seurre, et 2003:125-126).

The first 12 different multislot classes for type 1 MSs are presented in table 1. Type 1 MSs in the range class range from 1-12 are not full duplex devices. Thus, the maximum Rx+Tx is always below 8, such that the MS have enough time to switch between Rx and Tx inside one TDMA frame (Seurre, et 2003:125-126).

Multislot class	Rx	Tx	Sum Rx+Tx	T_{ta}	T_{tb}	T_{ra}	T_{rb}
1	1	1	2	3	2	4	2
2	2	1	3	3	2	3	1
3	2	2	3	3	2	3	1

4	3	1	4	3	1	3	1
5	2	2	4	3	1	3	1
6	3	2	4	3	1	3	1
7	3	3	4	3	1	3	1
8	4	1	5	3	1	2	1
9	3	2	5	3	1	2	1
10	4	2	5	3	1	2	1
11	4	3	5	3	1	2	1
12	4	4	5	2	1	2	1

Table 1. MS multislot classes.

GPRS standard have defined four different coding schemes from CS-1 to CS-4. These schemes are used on data transmitted on the PDTCH (Packet Data Traffic Channel). Different coding schemes are used to support error protection for the data. Applied coding scheme is selected by checking the carrier to interference ratio. Schemes CS-1 to CS-3 are using a ½ rate convolutional code, where known bits are punctured before interleaving, such that every burst consists of 456 bits. CS-4 does not perform any convolutional encoding. In all schemes the data block consisting of 456 bits are interleaved into four bursts, all transmitted inside one TDMA frame. In GPRS mode the signalling of CS used is given by the SF (Stealing Flag), 2 bits per burst and a total of 8 in a radio block (Seurre, et 2003:111-115).

Scheme	Code Rate	Data Rate / Kbps
CS-1	$\frac{1}{2}$	9,05
CS-2	$\frac{2}{3}$	13,4

CS-3	$\frac{3}{4}$	15,6
CS-4	1	21,4

Table 2. GPRS coding schemes.

2.2.4. GPRS Packet Data Mode

A GPRS attach for packet data mode is illustrated in Figure 8. An attach has to be done before the MS can start receiving or sending packets. Another preparation is also the PDP context activation as already described.

When the MS tries to attach to GPRS a request is sent to the SGSN serving the cell where the MS is camping. The core network performs several operations, such as authentication and exchanging of subscriber data needed by the SGSN. SGSN needs information about IMSI and subscriber information, otherwise mapping to NSAPI and TLLI pairs would not be possible. In the core network message signalling is done by the mobile application part (MAP) protocol over Gs and Gr interfaces (Seurre, et 2003:320-325).

After successful attach and PDP context activation the MS can receive paging requests from the network and also send packets to the network. Thus, a MS application may listen for incoming data, have a TCP port open for example, and also send TCP PDUs to other network nodes.

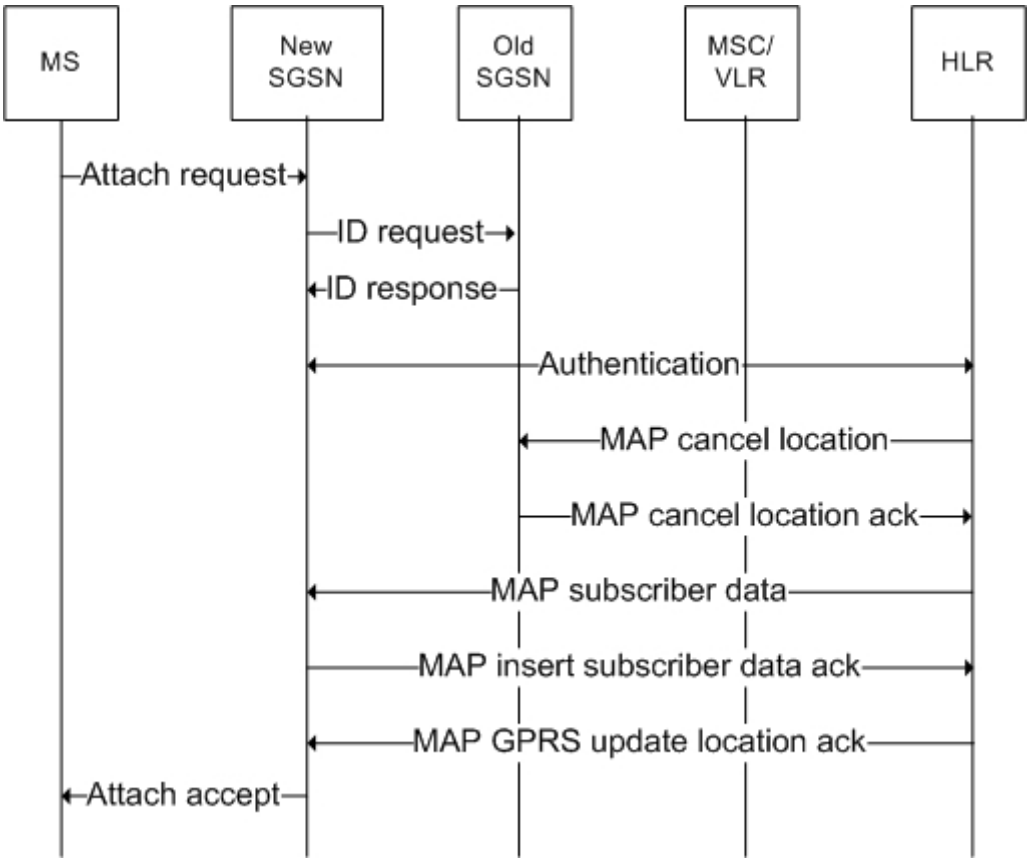


Figure 8. GPRS attach procedure.

2.3. Wireless Sensor Networks

The concept of Wireless Sensor Networks (WSN) is relatively new. The main target is to have sensor nodes that can move and operate without cable connection. There are several different protocol stacks for WSNs, such as Zigbee, 6loWPAN and HART protocol, all using the same PHY/MAC standard IEEE 802.15.4. In this thesis the focus is mostly on 6loWPAN.

A wireless sensor network consists of sensor nodes up to several hundreds of them. In addition to sensors, each node is equipped with radio frequency (RF) chip and microcontroller. A sensor node is often powered by batteries, which implies that the power consumption is very crucial in every sensor network design. This have led to the trend towards System-on-Chip (SoC) solutions, such that power consumption can be reduced.

2.3.1. IEEE 802.15.4

IEEE 802.15.4 standard specifies the physical and medium access control (MAC) sub-layers for low power Wireless Personal Area Networks (WPAN). Nodes that are participating in WSN all have such RF chip that implements IEEE 802.15.4 standard to hardware.

In therms of communication, the network topology in 802.15.4 networks can be star or peer-to-peer as illustrated in Figure 10. A network is built with sensor nodes that may be classified as reduced function device (RFD) or full function device (FFD). In the star topology the leaves on the outer edge are RFDs that can only communicate with their PAN coordinator node (FFD). In practice this means that the FFD checks the address of all incoming packets and discards all others except the packets originating from its PAN coordinator, since it can receive all packets from nodes inside its communication range. Even though RFD and FFD are defined separately all devices in the network employ the same hardware, and the software defines which kind of logical device it satisfies. A RFD device in star topology can not participate in routing, since it only communicates with its own PAN coordinator (IEEE 802.15.4, 2003).

In peer-to-peer networks all nodes may communicate with any node that are in communication range. Thus, any kind of network strategy can be used, since all nodes may participate in routing decisions to forward packets in the network. However, a routing protocol is not defined by IEEE and must be implemented on the higher layers. Even though every node can communicate with any other node, there must be one node that will be the PAN coordinator (IEEE 802.15.4, 2003).

Interfaces between layers are illustrated in Figure 9. The blocks illustrated in grey are the ones specified by IEEE 802.15.4. Upper layers such as network and transport are left open such that different solutions can be used (IEEE 802.15.4, 2003).

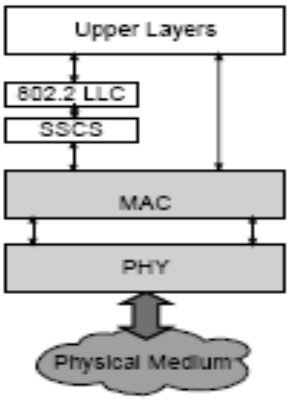


Figure 9. IEEE 802.15.4 protocol stack (IEEE 802.15.4, 2003).

The physical air interface for IEEE 802.15.4 is using the Industry Science and Medical (ISM) band. In the standard 16 channels are specified in the band ranging from 2400 to 2483,5 MHz. There are also two other bands supported, 868 and 915 MHz, although these are mainly not used since they can not be deployed worldwide due to band limitations in regional areas. From higher layers PHY access is made through two SAPs, one for data transmission and another one for control channel that is used to send commands or get information from the PHY layer (IEEE 802.15.4, 2003).

The PHY layer is responsible for transmission and reception of data, for measuring LQI values for received packets, and selecting the channel which will be used and for controlling the radio. In the 2450 MHz band the data rate is 250 kbit/s accomplished with a 16-ary modulation technique, where every 4-bit symbol is transferred into a 32-bit pseudo random (PN) chip code that is modulated with offset orthogonal phase shift keying (O-QPSK) onto the carrier, resulting in a chip rate of 2 Mchip/s, that is given as input to the antenna and transferred over the air interface (IEEE 802.15.4, 2003).

IEEE 802.15.4 MAC layer is responsible for network establishing and providing links between MAC entities in different nodes. It also takes care of carrier sense multiple access-collision avoidance (CSMA-CA) for channel access. Higher layers access MAC layer through SAPs (IEEE 802.15.4 2003). Protocol stacks often implement SAP interfaces as function calls that can be accessed by programmers. In these function calls the commands or information are given as arguments and the function execution returns the result.

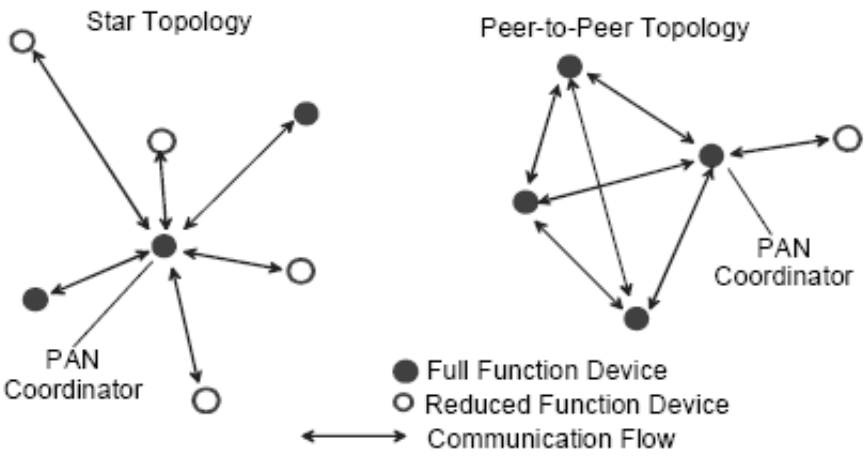


Figure 10. IEEE 802.15.4 network topologies. (IEEE 802.15.4 2003)

2.3.2. 6LoWPAN

6loWPAN is a standard for transmission of IPv6 packets over IEEE 802.15.4 networks. The standard is made by the Internet Engineering Task Force (IETF) group. As already described, the IPv6 packet header is 40 octets. If UDP, MAC, encryption and frame overhead are all added, only 33 octets will be available for the application in the worst case. (Montenegro, Kushalnagar, Hui & Culler, 2007). This clearly justifies the need for a compression of headers when transmitting over IEEE 802.15.4 networks.

An example of the 6LoWPAN header is illustrated in Figure 11 with blue colour. In the best case scenario only 9 bytes are used for network and transport layer headers. The first byte in the 6LoWPAN header contains information about the type of 6LoWPAN packet and the type of next header. HC1 and HC2 defines the compression format of IP and UDP. If full TCP/UDP would be used HC2 would be left out. Thus, every HC1 contains information about the next header as well. The only mandatory part that must be contained in IP part is the hop limit (Montenegro et. 2007).

Compressed UDP contains the same header information as normal UDP. UDP can be compressed to 4 bits for source and destination parts by the use of a predefined port number 61616 to which the 4 bit value is added if compressed ports are used. Otherwise the complete 16 bit port numbers are used. When the length field of UDP is com-

pressed, it is calculated from the IP layer length minus headers between IP and UDP (Montenegro etc. 2007).

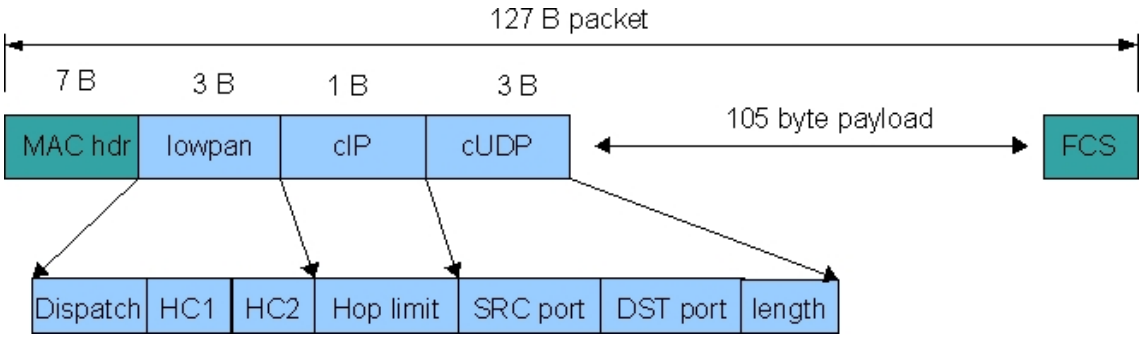


Figure 11. 6loWPAN packet example.

The compression of TCP header is not included in the RFC 4944, since it is not useful in general to use connection oriented protocol in a sensor network due to the overhead of traffic generated to establish and keep the connection alive.

2.4. Secure Communication

In the applications communicating over unsecured networks such as the Internet it is important to have the data flow secured from unauthorized users. However, when dealing with security it is not only the flow of data that needs to be secured but also physical devices.

Data communication security is a complex issue and its complete discussion is out of the focus for this thesis. The first thing to choose regarding the security is that at which layer the security function should be implemented. For example, data can be encrypted at the application layer with some usual encryption scheme such as advanced encryption standard (AES), data encryption standard (DES) or similar. On the other hand, security may also be implemented at transport or network layers. This can be done with socket secure layer (SSL) on top of TCP or IPsec in the case of network security.

In the context of this thesis there are two main security aspects. First, there are users accessing a control application over a web browser. Thus, a secure connection from host

computer to server application is needed. Second, a secure link between the WSN and the end server is considered.

2.4.1. Data Authentication

Data authentication has already many suggested and implemented solutions. However, many security functions are heavy applications that cause a lot of burden for the central processing unit (CPU) (Stallings, 2003). In embedded systems, the CPU resources are scarce and need to be used efficiently. Therefore, common encryption functions are not suitable without a hardware implementation.

From the reasons discussed previously all encryption algorithms are discarded and we consider hash functions instead. A hash function generates a message digest value on a given message such that:

$$h = H(M) \tag{1}$$

where M is the message and H is a hash function that produces a digest h. A sender computes the hash value and sends the message. The receiver can then compute the same hash value and compare the hash values to each other. If the received value and the locally computed value are both the same, the message is considered to be the same (Stallings, 2003).

The message-digest algorithm 5 (MD5) published under the IETF document RFC 1321 is a hash algorithm that takes a message of variable length as input, from the input a fixed length 128-bit message digest is produced. First the message is padded such that the length of the message is congruent to 448 modulo 512. Each 512 bit block is then processed as given in the block diagram for a single MD5 operation illustrated in Figure 12. For every 512 bit block the same hash operation is used, the 128 bit register holds the intermediate, as well as the final value of the checksum value. When all 512 bit blocks have been processed the final value is available at the register (Rivest, 1992).

The MD5 algorithm proceeds as following:

1. First 128 bit register is initialized and represented by the 32-bit strings A,B,C and D.

2. Then the process illustrated in Figure 12 begins. A 512 bit block is taken as input and computed in four rounds with the logical functions F,G,H and I, defined as follows using the logical operations: (Rivest, 1992)

   $$F(X,Y,Z) = XY \vee \neg(Z)X$$
   $$G(X,Y,Z) = XZ \vee Y\,\neg(Z)$$
   $$H(X,Y,Z) = XY \oplus Y \oplus Z$$
   $$I(X,Y,Z) = Y \oplus (X \vee \neg Z)$$

   The values in table T are calculated from the sine function to produce a randomized representation (Stallings, 2003).

3. The same procedure is done until all 512 bit blocks are produced. On completion the 128 bit message digest is available from A,B,C and D registers.

The single operation in one block of the MD5 algorithm is illustrated in Figure 13. The registers are circulated to right at every block operation, g represents one of the logical functions F, G, H or I as described above. Addition is done mod 2^{32}, and the circular left shift is done *s* bits for 32 bit words (Stallings, 2003).

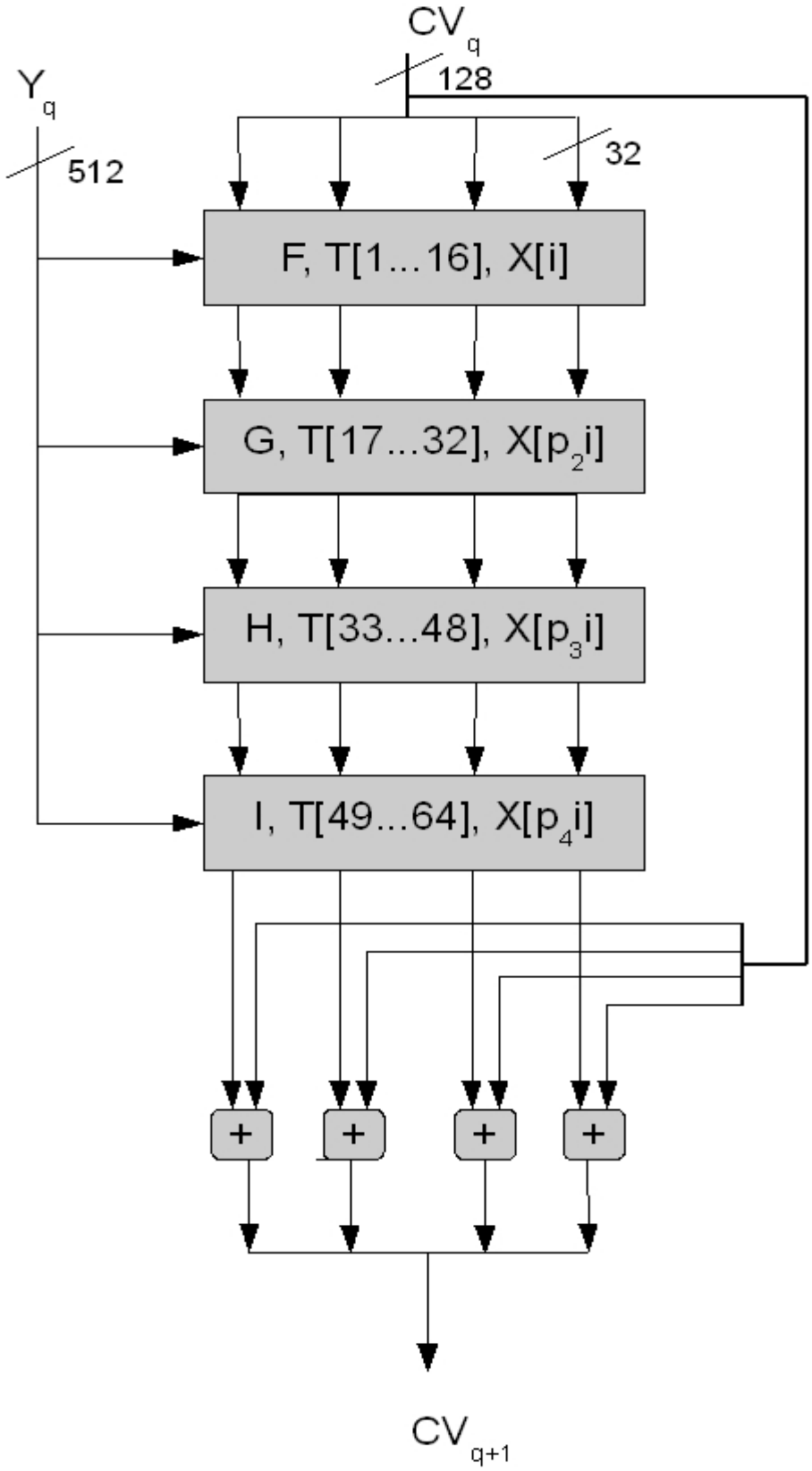


Figure 12. MD5 512 bit word hashing.

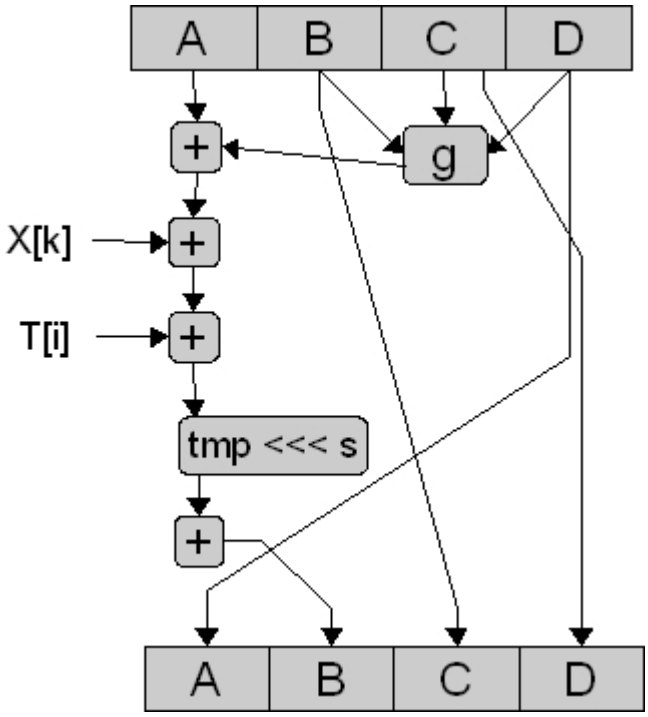


Figure 13. MD5 algorithm single block operation.

Data authentication with a hash function can be performed as illustrated in Figure 14. The sender and receiver share a common key S. Once a message is to be sent, the sender adds S to message M and calculates the hash value for M and S. The hash value is added to M and sent over the channel. The receiver calculates the hash value for M padded with S and compares the result to the received hash value. If the values match the message is originated from an authorized user, since S is secret. Note that this only performs authentication of the message. The message itself is sent as plain text and can be captured and read (Stallings, 2003).

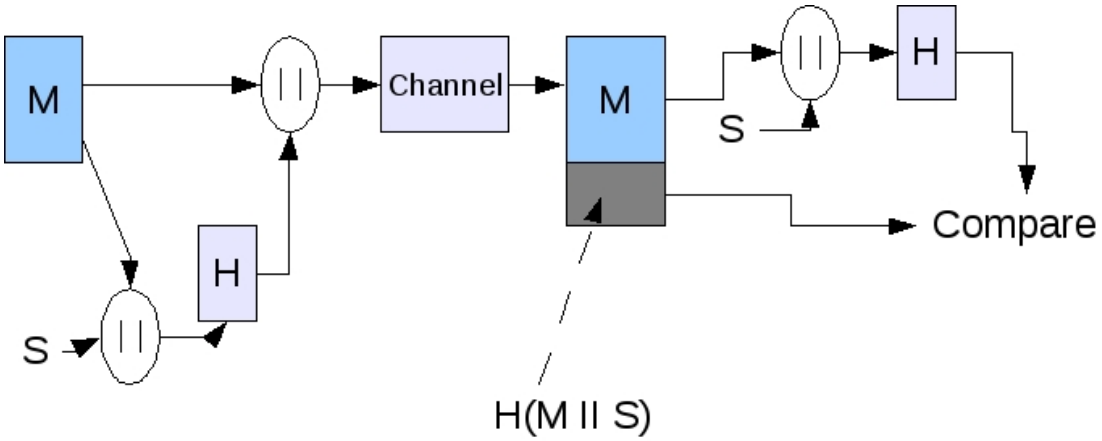


Figure 14. Message authentication with hash function.

3. SYSTEM DESCRIPTION

A system for remote monitoring and control of a distributed energy production have been built in the context of this thesis. First a description of the hardware employed is given followed by the specification and description of the developed system.

3.1. Hardware

Several printed circuit boards (PCB) have been made to fulfil different purposes in this project. The PCB development have been made in Mentor Graphics PADS software, and the PCBs were made in Technobothnia PCB laboratory. A Vaisala WXT520 weather transmitter and Telit GM862 GPRS module were bought for this project. Sensor network devices used in this thesis were already available at Technobothnia laboratory.

3.1.1. Vaisala WXT520 Weather Transmitter

The Vaisala WXT520 weather transmitter is capable of measuring following weather parameters (Vaisala, 2008).

- Wind speed
- Wind direction
- Pressure
- Temperature
- Humidity
- Precipitation

WXT520 weather transmitter needs a DC (Direct Current) operating voltage between 5-32*V,* where the power supply must be able to deliver 60 m*A* current at 12 *V*. The transmitter also have a heating system for rain measurements to keep snow, ice and rain away from the metallic surface. Heating system is fed with its own voltage supply in the same range as main supply. The maximum current drawn is 1.1*A* at 12*V*, therefore from the power equation $P_{max} = U \cdot I_{max} = 12\text{V} \cdot 1.1\text{A} \approx 12\text{W}$. According to previous calcula-

tion, if heating is enabled the power supply must be able to deliver an additional 12 W of power, in addition to the power drawn by normal operation (Vaisala, 2008).

Following power consumptions are given by Vaisala at 12V operating voltage:

- Wind measurement, continuous measurement 2-5 m
- PTU (Pressure, Temperature and Humidity) measurement, 0,8 mA.
- Precipitation, continuous measurement, 0,07 mA.
- ASCII RS-232 standby mode, 0,24 mA.

Combining the above we get a power consumption of:

$$P = U_N \cdot (I_{wind} + I_{PTU} + I_{Prec} + I_{RS}) = 12\text{V} \cdot (5 + 0{,}8 + 0{,}07 + 0{,}24)\,mA \approx 73\text{mW}$$

The fact that heating is consuming a lot of power may be a limiting factor at the system deployment area, As a consequence, the use of heating should be carefully considered from case to case.

The WXT520 can communicate by following protocols RS-232, RS-485/422 and SDI-12 (Vaisala, 2008).

If different connections than the original needs to be used, it is possible to reconfigure the internal wiring by disconnecting the bottom of the transmitter. Thus, the connection between different devices is very flexible. The RS-232 voltage swing is between 0 and +4,5 V. Therefore, high data rates on the serial cable can not be used over long distances. It is recommended that the cable length does not exceed 100 m (Vaisala, 2008).

3.1.2. Sensinode Nano-series Platform

The sensor nodes applied in this thesis work are manufactured by Sensinode Ltd, located in Oulu, Finland (Sensinode, 2009). The sensor nodes in nano series are built with

a RC2301AT device manufactured by Radiocrafts (Radiocrafts, 2009). Radiocrafts RC2301AT module is a SoC solution built around a CC2431 manufactured by Texas Instruments (TI), which includes a 8051 microcontroller core and 802.15.4 transceiver. In addition, Radiocrafts have added an integrated antenna which makes the module ready for use with IEEE 802.15.4 networks.

There are three different sensor products available in Sensinode Nano-series. N100 is a breakout board for the RC2301AT, where microcontroller pins have been taken out. Thus, it is suitable for integration with own manufactured PCBs. N711 is a demonstration board including temperature and light intensity sensors. With two leds and buttons it is an easy way to demonstrate and deploy sensor networks for testing purposes. Last product in Nano Series is the N601 router that includes a USB connector for connection with PCs. N601 can be used as a data logging unit or for network debugging.

For this thesis work the N100 and N711 models have been used. N100 modules are added to manufactured PCBs and N711 are used as routing nodes.

3.1.3. Telit GM862 Module

GM862 module from Telit Communications, is a GSM/GPRS module that can be used for any GSM related tasks. The module has a built in SIM card reader, seen on the right side of the module in Figure 15, which clearly lightens the design of customer applications. Interface to the module is made by a 50-pin Molex connector. The physical size of the module is a square of 43,9mm and 6,9 mm thick (Telit, 2008; Telit, 2007).

GM862 is a mobile station class B device, with support for CS 1-4 in GPRS class 10 (Telit, 2008). Thus, a theoretical maximum data transfer rate for the module would be ~60 kbps upload and ~40 kbps download using CS-4, with three slots download and two slots upload. However, in real applications such data speeds are not realistic to achieve.

The module has a built in TCP/IP stack that is configured through AT commands. One International Telecommunications Union – Telecommunications sector (ITU-T) V.24 serial port with maximum baud rate of 115 kpbs (RS-232 serial port) is applied. It is

possible to use the module as a stand alone application since it has 13 I/O ports for general use and one A/D converter (Telit, 2008).

Programming of the GM862 module is done in Python programming language. When Python interpreter is enabled the normal AT interface on the serial port is disabled and commands to the terminal equipment is made by using internal message bus through Python function calls (Telit, 2008).

Figure 15. Telit GM862 module. (Telit, 2008)

3.2. Developed System Description

3.2.1. Gateway Node

The gateway node consists of a Nano Series N100 sensor and a Telit GM862 that are interconnected through an universal asynchronous receive and transmit (UART). A logical overview of the the gateway node is given in Figure 16. The nodes as seen in the fig-

ure are located at separated boards, such that it is easy to move one of them without the need to design a complete new board. The reason to keep the power board away from the GM862 module is the possible change of power supply source.

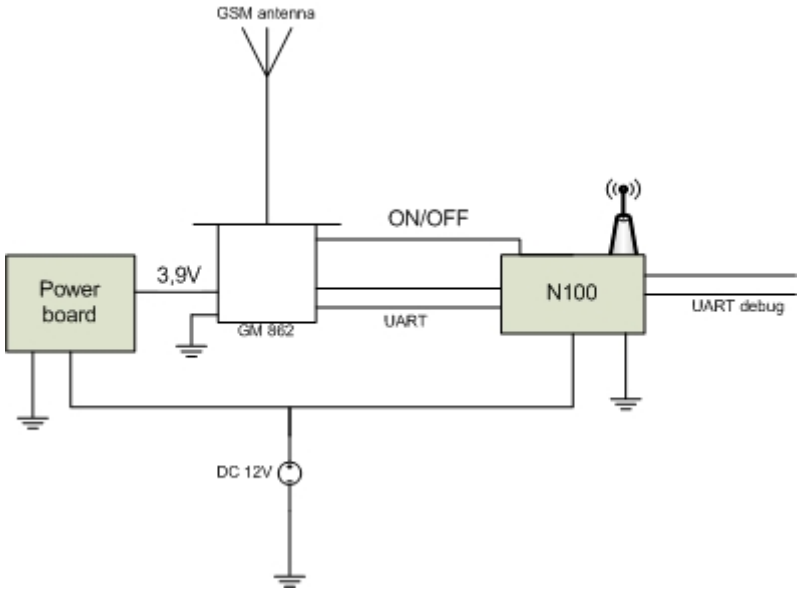


Figure 16. Gateway Node Logical Overview.

One critical design issue with the GM862 module is that it may draw up to 2A power spikes from the supply during transmissions. Therefore, a linear regulator is not suited, when there are large voltage differences between input and output voltage. The linear regulator may have losses up to 70 % (Telit, 2007).

From above assumptions a switching regulator suits the application better, since the losses of a switching regulator are not that high as in a linear regulator. Although, with large switching frequency the traces on the PCB starts to act as antennas, and voltage will be induced between the traces. Thus, a very careful PCB design must be made. A switching regulator L2576T-ADJ was chosen, since it can deliver up to 3 A to the load, and its switching frequency is 52 kHz which relaxes the PCB design issue (National Semiconductor, 2004).

For the adjustable version of LM2576 the feedback pin is set between two resistors as shown in the schematic of the power board, see APPENDIX 1. The resistors are chosen

by using the following formula, where R_1 should be between 1 and 5 kΩ (National Semiconductor, 2004).

$$V_{out} = 1{,}23 \cdot (1 + \frac{R_2}{R_1}) \quad \text{(National Semiconductors 2004)} \tag{2}$$

By choosing R_1 to a resistor available it is easy to calculate R_2 from formula 2, by solving for R_2:.

$$R_2 = (\frac{V_{out}}{1{,}23} - 1) \cdot R_1 \tag{3}$$

Since the GM862 needs 3,8 input voltage that is set as V_{out} and R_1. After some iterations of different R_1 values the pair of resistor values were finally found to be: R_1 = 1,27kΩ and R_2 = 2,77kΩ.

The power board built is made up around the L2576T-ADJ regulator. The inductor used is specially designed for switching regulators, and the capacitors have been chosen with low equivalent series resistance (ESR) to improve the performance of the regulator and to avoid large ripple in the output voltage. In the design it is also important to keep capacitor and diode traces short, otherwise there will be series inductance and resistance in the circuit that induces fast transients in a switching circuit (National Semiconductors, 2004).

The second board is the GM862 board. On this board only the GM862 module is placed and its pins is taken out. The traces of the power supply are made much thicker than other traces to ensure reliable power supply, since spikes may occur on the power line during transmissions. On this board the connector is a surface mounted device (SMD) a which was soldered with solder paste and hot airflow.

Designed PCBs are illustrated in Figure 17, where the power board is placed to the left and GM862 board to the right. On the GM862 board the module itself is placed under the board on the solder side (SMC). The black cable from GM862 board is the GSM an-

tenna cable which is mounted outside of the box. Other connections include power cables from power board, RS232 and on/off cables to gateway N100 node .

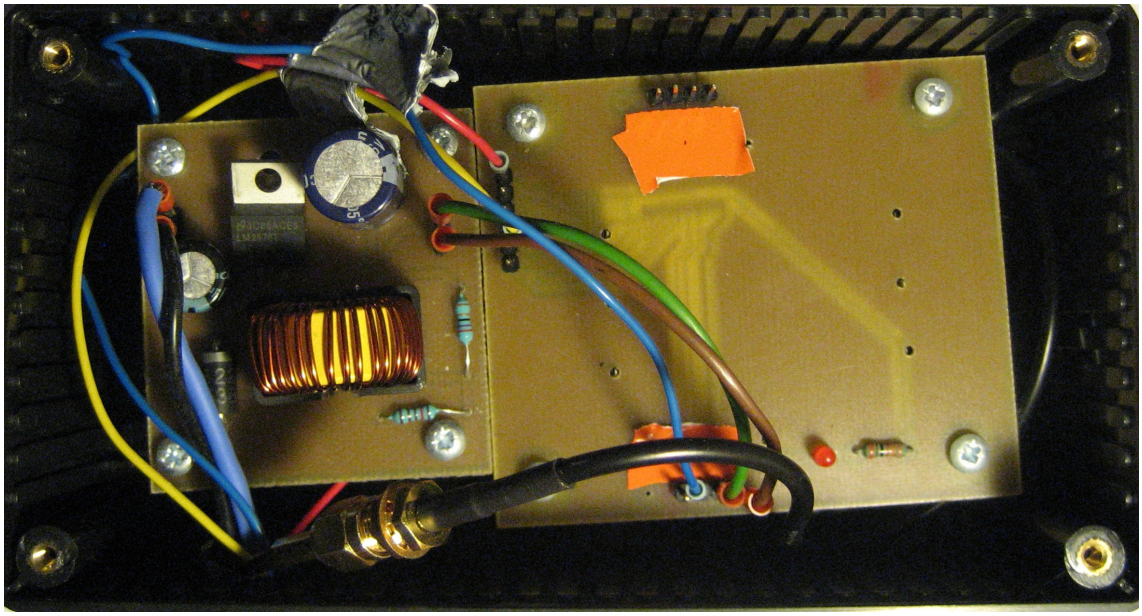


Figure 17. GPRS node and its power board inside protection box.

Last board that belongs to the gateway is the N100 wireless sensor interface, in the right side of Figure 16. As inputs to this board the 12 V power supply feeds a LM2937-3.3 voltage regulator that gives a constant 3,3V output, which is taken to N100 modules voltage pins. Also located on this board is a NPN transistor, controlled by one of the N100 module's general purpose input/output (GPIO) pin, which is used to turn on and off the GM862 module. The intercommunication between GM862 and N100 modules is done with the 2 line UART0. Additionally UART1 pins are made available on the board to enable debugging.

In Figure 18 the N100 gateway board is figured inside its protection box. As presented in the figure, only N100 module, voltage regulator with two filtering capacitors and NPN transistor with according resistors are located at this board. Even though the protection box itself takes away some of the range for WSN communication it is necessary in the deployment to avoid moisture and dust. A more specialized solution for the gateway could be built with everything on the same board, though on cost of the flexibility.

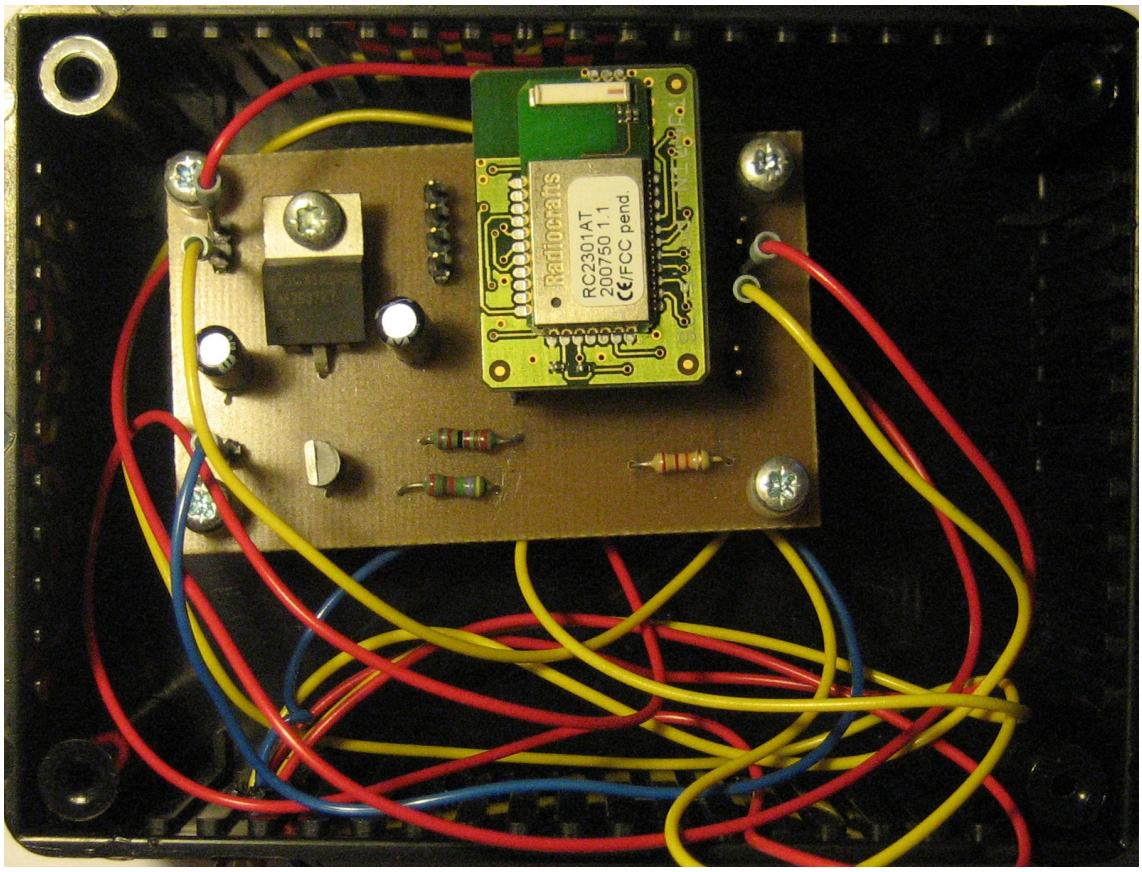


Figure 18. N100 gateway node board.

3.2.2. System Description

To start up the GM862 module its ON/OFF pin must be driven low for one second (pulled to ground), as the system will work on a remote location a switch button is not suitable and therefore the on/off pin is driven by a GPIO of the CC2431 MCU. It is connected to a NPN transistor such that when a 1 second pulse is generated the collector-emitter path will lead from ON/OFF pin of the GM862 to the ground.

A common ground is present for N100 and GM862, since they both share the supply power cable. It is necessary to have a common ground in this case to have reliable UART communication, since no signal ground is available at N100 module and therefore signal voltage must be compared to ground voltage. Above enables a minimum of two cables between N100 and GM862 for UART communication, RX and TX. Al-

though, also clear to send (CTS) and ready to send (RTS) pins are made available on the PCBs for both modules so that hardware flow control may be used.

3.3. Weather Measuring Solution

Weather station system is built up by using a WXT520 weather transmitter and a N100 device. The logical structure of weather system is given in Figure 19, where N100 node is placed on one PCB and only interface cables is added to WXT520.

WXT520 is connected with the N100 module through a RS-232 interface. Since the N100 module uses CMOS levels a 74HC04 hex-inverter is located on the board where UART lines are connected to invert the voltage levels between WXT520 and the N100 module. When the hex-inverter is driven by the 3,3V output of a LM2937-3.3 voltage regulator the threshold for a high bit is 1,5V and for a low bit 0,5V (Philips Semiconductors, 2003). The inverter will not output any higher voltage than its driving voltage V_{cc}. Thus, there is no need for additional protection of the N100 module that can handle a maximum of 3,6V on its GPIO pins. Although, it is necessary to have the same ground voltage for both supplies, otherwise the $V_{p\text{-}p}$ may exceed the maximum tolerance level, since no signal ground is available from the N100 module.

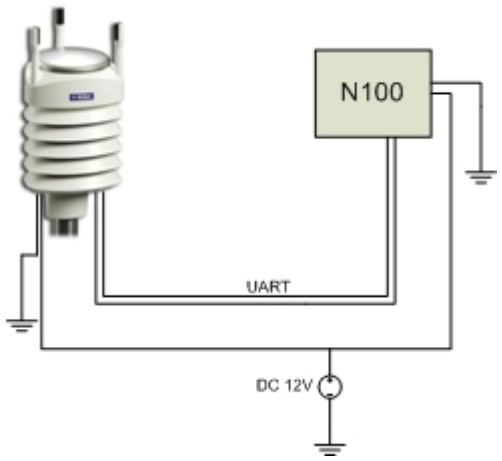


Figure 19. Weather measuring solution.

3.3.1. System Description

WXT measuring system set up is simple, since WXT520 weather transmitter starts up automatically when voltage is fed to its supply pins. Therefore we have to only care about N100 module.

N100 module is powered with a LM2937-3.3 regulator in the same manner as gateway node. In this case the regulator also powers the hex-inverter to enable logical signals inversion (UART).

4. SOFTWARE ARCHITECTURE

4.1. NanoStack

NanoStack is a 6LoWPAN implementation over IEEE 802.15.4 developed by Sensinode and published as open source on http://sourceforge.net. It is implemented and integrated with real time operating system (RTOS) called FreeRTOS. FreeRTOS uses tasks that have different priority. Shifting between the tasks is done when the running task is set to sleep and the scheduler chooses the next task to execute. Tasks may have different priorities in FreeRTOS.

NanoStack protocol stack is running as one task when used. The intercommunication between tasks can be done by queues or semaphores. In NanoStack implementation there is also a type called buffer_t that is used for stack communication.

The socket API used in NanoStack is very similar to the Berkley software distribution (BSD) socket API. Thus, a socket must be requested from the "kernel", bind to a specific port and may then be used for UDP communication.

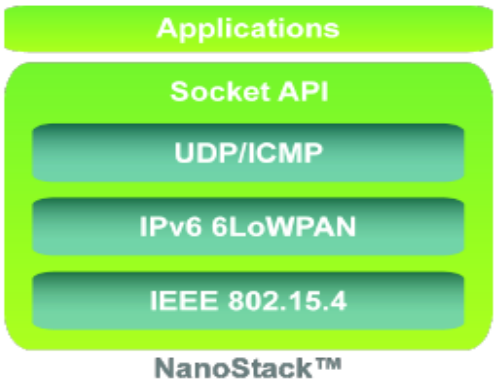


Figure 20. NanoStack protocol stack.

4.2. Contiki

As part of this thesis the Contiki operating system have been considered, since the support for NanoStack is deprecated and they have switched to closed source in the latest NanoStack versions. Contiki is an open source project maintained by Swedish institute of computer science (SICS) (Contiki, 2009). It offers a very memory efficient operating system (OS) for embedded devices. There are support for many different platforms. A port for the CC2430 is under development by Sensinode, unfortunately it was not finished in time to be used in this thesis work.

Some initial testing was done with Contiki and the results look promising. The lightweight implementation of protothreads, thread like implementation for Contiki, gives a very fast solution for programming of different applications. Also the limitation with FreeRTOS, that it must be built with large-auto-stack with SDCC compiler when used with 8051 MCU core, is vanished. That limitation causes some problems with the efficiency and use of libraries.

The Contiki OS brings the feature that different types of nodes may be used with the same application and work together, with only limited need of changes in software. There are also a large numbers of researchers using Contiki compared to NanoStack and therefore it is an excellent choice in research projects for embedded network solutions. There is also a 6loWPAN stack available in Contiki already. The switching from NanoStack to Contiki has been tested to be quite straightforward, because the code implemented in tasks for FreeRTOS can be inserted into a protothread in Contiki, and only FreeRTOS dependency functions must be switched to similar Contiki based solutions.

Also the support from the Contiki community is much larger simce all platforms are using the same code structure for OS dependent functions, which brings further possibilities for support. On the NanoStack side there are very hard to get some support since the users of 8051 cores with FreeRTOS are very rare. As a conclusion, once the Contiki port is ready it may open opportunities to further collaboration between Sensinode and the research community.

4.3. Gateway

As described previously the gateway is a combination of the GPRS module and the WSN sink node. Therefore also the software architecture has two parts. Telits GM862 module is programmed with Python and the sensor node with C. Their common interface is the UART serial line as already described.

One important design requirement for this project has been to enable two way communication over the GPRS link. Thus, scheduling and synchronization between GPRS link and sensor network had to be carefully designed.

GPRS module have its challenges. It must have one socket in listen mode to wait for incoming connections. This would imply a threaded application that would block in listen state for incoming connections and another thread that would handle serial communication. Unfortunately threads are not supported on the Telit module. Thus, a continuous polling of both incoming connections as well as UART communication is needed. This raises a problem of synchronization between GPRS module and sensor node. As a solution it has been chosen to add the start of packet and the end of packet bytes to UART communication, such that GPRS module looks for the end of packets from the received UART buffer. Sensor node polls the UART buffer often enough so that maximum of one packet from GPRS side can be received between two adjacent poll times.

Control messages that are critical for the application are checked with MD5 algorithm as already described in Chapter 2. When such a packet arrives that contains a message with critical data, the message is verified. After success the command is executed, otherwise discarded.

To support new features in the network, by updating the device software, without removing the module from the location a remote script update service is made available. Python source codes are stored on a FTP-server from which the module can download them. A message that a new version of a script is available will be sent to the module, which first verifies the message, such that it is valid and then opens a FTP connection and downloads the script. A source code script has its MD5 checksum appended in the

end of the file, which the module extracts and recalculates the MD5 checksum and compares them against each other, such that no transmission errors have happened. It is very critical that a script is valid and compiles, otherwise the start up procedure will fail. When a message is verified the old file is truncated and the new content written. Then, the module is set to reset, which will make the module to compile the new script and the software is updated.

Since the N100 node that forwards packets to other nodes in the network do not have to know what kind of information it is forwarding, when adding new nodes to the the network with new hardware/software capabilities, GPRS module is updated through remote script update and the sink node may be left unchanged.

The N100 module that works as the interface between the WSN and GPRS module also needs a quite complex design to deal with both WSN related communication as well as GPRS communication. In the design the gateway WSN node was chosen as to be the one that will take care of failures of any kind. This means that it regularly checks that the GPRS module is powered and serial line works. It is further secured with a watchdog timer that continuously has to be cleared. Otherwise the watchdog will reset the node. As a result a reset occurs if the software gets stuck for some reason or other failures such that the regular clearing of the watchdog would not be done in time. This assures that the node is always online when power is available at the supply cable.

4.4. Sensor Nodes

There are several sensor nodes participating in the network. Each of them have their own software running. Detailed description about the sensor node software is given in this section.

4.4.1. WXT Node

The node that handles WXT520 operation has a quite simple and limited functionality. Its main goal is to listen for incoming queries from the network and regularly query

measurements from all WXT520 sensors. For these purpose the node uses the UART hardware to send out ASCII messages with queries, WXT520 weather transmitter, answers over the same bus with the results. At the moment no parsing of the messages is done at this node, but all parsing is done on the campus server. Future features for this node, that are not implemented in the first phase, include local operation of WXT520 weather transmitter, such as heating control and transmission of weather data to other sensor nodes in the network.

Weather measurements are collected at a fixed interval, which can be changed during run time. Initially the interval is 3 minutes. The setting of measuring interval is given in $sleeptime \times 10s$. Thus, if a sleep time of 10 minutes is required sleep time would be set to 60, which would result in $60 \times 10s = 600s = 10min$.

4.5. Collecting Server

The server located at Technobothnia in the University of Vaasa campus collects measurements that has been measured on the remote location. It also works as a gateway for the web-application. When control is requested from a web browser, a message is passed to the server application that forwards them to the remote location GPRS module. This is done since the GPRS module has a dynamic IP address that is reported to the server upon start up so that server knows to which IP address packets should be sent. Critical control messages need also careful consideration. Therefore this solution was chosen, where the server calculates the MD5 checksum before sending packets over the Internet. The complete server software is written in C programming language.

The server runs two threads (see Figure 21) where one thread listens for incoming connections and the other for IPC communication. In the socket thread a call to listen is made which will block until a connection is arriving at the socket. When a connection is established the server will handle the type of data and, if necessary, save it to the database. IPC thread first creates a queue and starts to listen on this queue. Messages will arrive at the queue when the web-interface sends a query that should be sent to remote

location. Upon arrival of a message the IPC thread will send the message to the socket which is handled by the socket thread

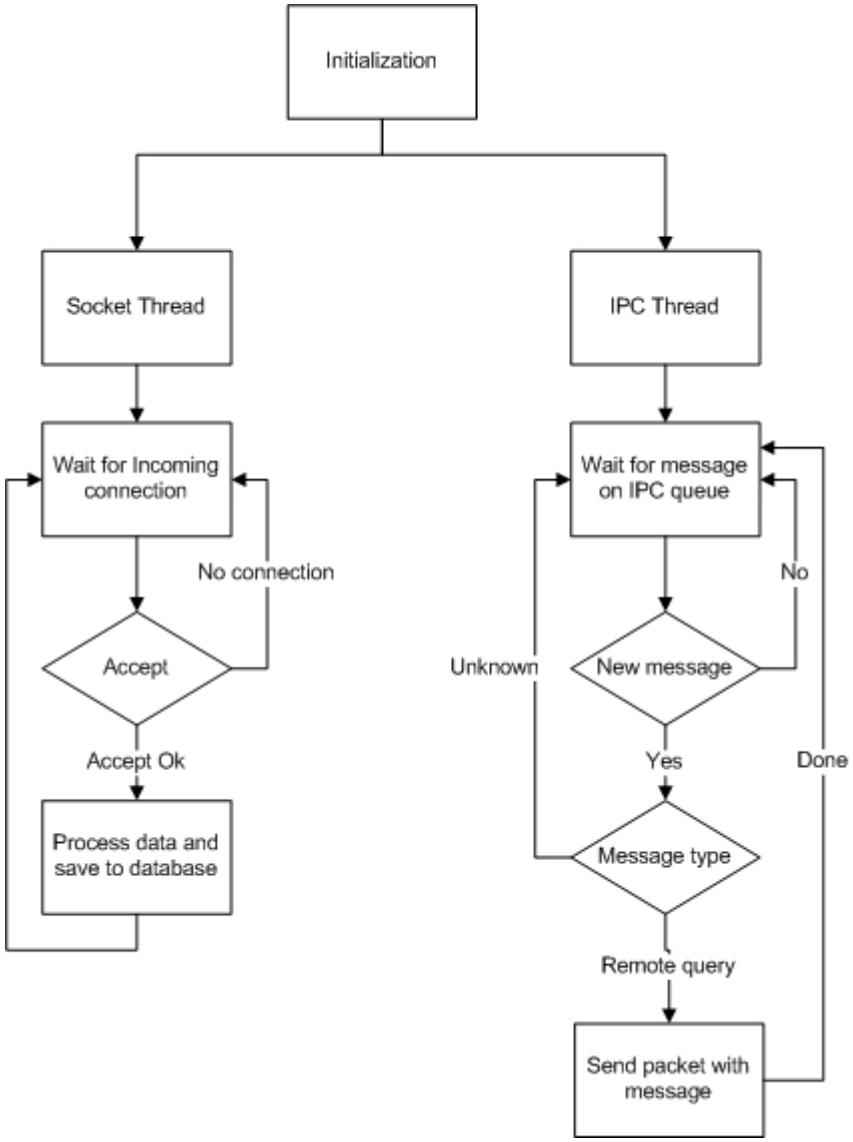


Figure 21. Software data flow.

To calculate the MD5 checksums the open source library provided by L. Peter Deutsch available on http://sourceforge.net has been used. This library is directly implemented from the RFC document example code describing MD5.

4.5.1. Web-Interface

A user interface for the visualization of the measured data and for the control is an important part of remote monitoring and control application. However, since the usability is not on the main focus of this thesis, just a simple web interface has been created. A login prompts for legitimized user password are applied. The user database is stored on the Technobothnia server with the passwords saved as MD5 checksums to avoid password leakage if database would be hacked.

Regular measurements from the remote location weather station are shown on the front page, and additional information can be requested by doing a query from the web interface. Weather data can be saved in Matlab format for further processing in research purpose.

4.5.2. Database

The database used in this thesis is a MySql database located at Technobothnia server in the University of Vaasa campus. The database itself is very simple and does not contain any complex relations. Three tables are created for weather data and one table for user account data. The database structure is illustrated in Figure 22.

The weather data is divided into the three tables as presented in Figure 22 due to the simple fact that there are three specific queries to do from WXT station to retrieve all measurements. In the weather table, temperature, humidity and air pressure are saved as floating point values. In wind table all data related to wind measurements is saved and finally rain related data is stored in the precipitation table.

All data that is stored in the MySql database can then be queried by SQL statements with a given criteria to obtain values wanted. The server only accepts localhost connections, therefore SSL tunnel is needed to retrieve queries from other hosts. In this particular application the web interface is located on the same server and no SSL tunnel is used.

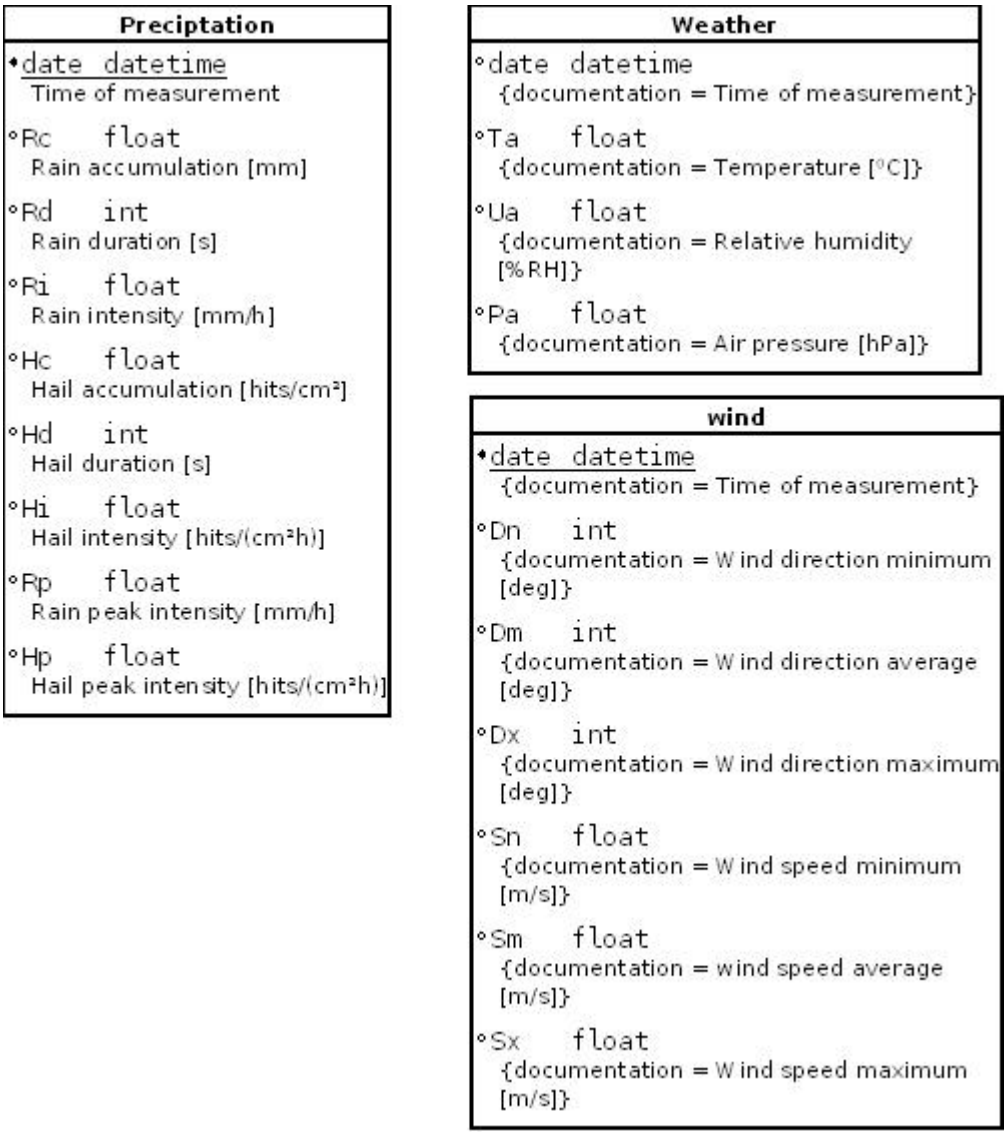


Figure 22. Measurement tables.

5. EXPERIMENTAL SET UP

The system described in the previous chapter have been deployed for experimental use. In the design phase the nodes are not intended to operate during winter time. The maximum operation time is limited for battery power nodes, but nodes with wired power supply have no lifetime limit. It is indeed very challenging to develop a sensor network for continuous outdoor use in Finland, where the temperature may fall to tens of minus centigrades, during wintertime. To be able to support network operation at winter time, further investigation is needed.

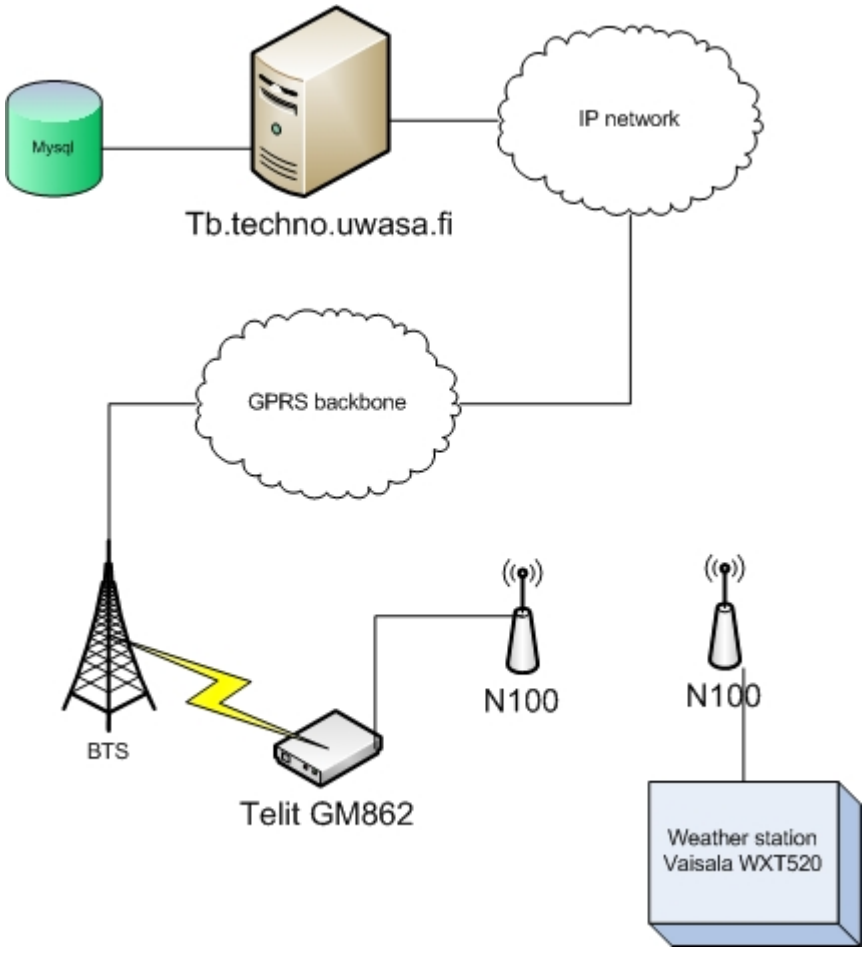


Figure 23. Deployed system overview.

As a first test set up the weather transmitter and gateway node are deployed. The network is planned in such a way that additional sensors can be deployed in later phases without changes to any other nodes. Multi-hop support is included such that we can

cover longer distances by using multi-hop paths consisting of several radio links between source and destination.

The overall view of the system is given in Figure 23. Starting from the weather station a query is made from N100 node, over the cable connection, to get latest measurements from weather station. Data is transmitted over same cable connection to N100 node, where ASCII data is encapsulated into a IEEE 802.15.4 packet. From the N100 node, packet is transmitted to next N100 node, over wireless link, which acts as the WSN gateway. If the direct path is longer than reliable communication range, we may add more N711 nodes who will act as a routers between weather station N100 node and GPRS modem interacting N100 node. N100 gateway node copies the data buffer from the IEEE 802.15.4 packet to its UART buffer and sends the data to Telit GM862 module over the serial cable connection. The Telit module reads the header of the packet to decide what kind of data it is. If the packet contains measured data it encapsulates it into a TCP/IP packet on port 4048 and transmits it to BTS over the Um air interface. GSM core network routes it to GGSN from where it is encapsulated with GGSN IP address and forwarded to a public IP network further routed to uwasa.fi server. The uwasa.fi main server is configured to forward packets on port 4048 to tb.techno.uwasa.fi server. When arrival of a packet to tb.techno.uwasa.fi server, the packet is decoded and handled with respect to the content, with measured data, a connection to MySql database is opened and measurements saved into right table.

5.1. Söderfjärden Research Station

The meteorite crater at Söderfjärden area south of Vaasa works as the first experimental spot. At the location there is a museum about the meteor crash, an energy basement including a diesel generator set, energy storage batteries and a local control of the energy production. Outside of the basement a 0.9 kW wind turbine generator and four solar panels with a total effect of 0.3 kW supports the facility with energy. Additionally a diesel generator (11kW) is available if there is need for extra power.

Inside the museum there is a sub-station from where the supply cable is drawn to sensor nodes, weather station and GPRS module. There is already 12 V supply available from the sub-station, therefore no additional transformers are needed.

Vaisala WXT 520 weather station is mounted at the roof of the building, see Figure 24. A cable including power supply and UART communication cables is taken from the weather station inside the building, where the receiving WSN node is placed.

Figure 24. WXT520 mounted on top of Meteoriihi museum.

GPRS module and receiving WSN node are placed on the opposite side of the small room on the second floor of the building. First tests with Sensinode devices (into protecting boxes), indicated that only short ranges can be used. In inside environment up to 10 meters was reliable. Therefore the distance between the node which is cable-connected to weather station and the node which is cable-connected with the GPRS modem can be only a couple of meters. and WXT node and GPRS receiving node is only a couple of meters, to obtain a reliable radio channel.

5.2. System Operation and Maintenance

The complete system was developed to work independently without continuous maintenance. All operations that are necessary to make can be done remotely through a web-interface. If something unexpected happens and causes a system malfunctioning, someone must go to the site and manually boot the GPRS modem, which will automatically reset the whole system to the initialization state.

5.3. Wireless Control Loop

One of the design issues in this work was to create an architecture for remote control of the biodiesel generator and the electricity system. Obviously there are delays in the GPRS used in the public GSM network. Time delays are also introduced by the polling mechanism of the GPRS module and by the back-to-back transfers inside the WSN. These delays are challenging to handle, since they are time variant. Thus, advanced statistical methods must be applied and one must avoid time-critical applications.

Developing the control part is out of the focus of this thesis. As a preliminary work a mechanism that keeps track on the application to which to send the request. At the moment this mechanism is working, but the server application only puts the request on a common queue from where the web application is reading the answer. Thus, if several users would use the web interface there could be mismatching in the reception queue. This feature does not require any changes to WSN, only to GPRS module and to the server application. The system was developed such that it can be deployed as such without the feature, and the feature can be added later on over the air script update.

5.4. Security

Since the remote location were the system is mounted is working as a museum it has also been a critical issue to ensure that the system is well hidden such that it is not visible to the visitors. At the end it was quite easy to perform basic security. A node con-

nected to weather station is placed on the top of a construction bearer for the building, thus it is high above the ground level.

GPRS modem and receiving node are put in covering boxes and placed on top of one the museums displays. At the site it is quite dark and a black wall hides the boxes well. If one does not now that the nodes are placed there it is very hard to notice. It was considered to be safe enough for this project, since the facility is locked during the time when no presentations are ongoing.

6. RESULTS

6.1. About the System Performance

Operation of the system has only been tested in summer conditions and no expectations about how the system would perform in winter conditions are known. Even though the sensor nodes are covered by protection boxes it is not guaranteed that the protection is well enough during winter time, when cold weather might interfere with the electronics and make them malfunction.

During the test phase it was also noticed that even though the watchdog is enabled for the sensor nodes they do not operate as well as expected. It is clear that a system must be tested further before it can operate independently and without maintenance. It was noticed during the experiments that a small voltage drop on the supply voltage may interfere with the nodes in such a way that they do not run the software any more.

6.2. Wireless Sensor Network Analysis

The sensor network was tested with additional nodes and the nodes that were cable-connected to weather station and to the GPRS modem. First test set up was done with one routing node inside the building and the measuring node outside the building on ground level. This test failed due to communication problems. The wireless sensor nodes were not able to communicate through the 15 cm thick log type wooden wall. The compact wall dissipated all the power in such a way that it was not possible to continue this test.

In the second test both of the nodes were placed inside the building, the routing node and the measuring node. In this set up the communication went well and measurements were transmitted over the long haul link to the collecting server. The measuring node measured temperature and luminosity using N711 sensors. Since the node was placed inside the building during the test, there was not much variations in the measurements. However the main focus was to test the possibility to use routing and multiple measure-

ment nodes. It was not tested how many nodes there can be in the network before the packet drop would be too large to accept.

Figure 25 Shows collected temperature measurements between 13-15 of September 2009. There are a total of 930 measurements. During this two day measurement period the total throughput was 94,5 %. This level of reliability is relatively good for a WSN and is also comparable to WSN measurements made in the lab environment. An interesting result is that the lost packets were always in groups, such that inside a 15 minute interval there could be 3-4 lost packets or a packet loss of ~50%. There is no easy explanation to this behaviour although in cases where several packets where lost in a sequence there might have been a malfunction of the Telit module which will be reset by the N100 module if no response is received from the module within three minutes.

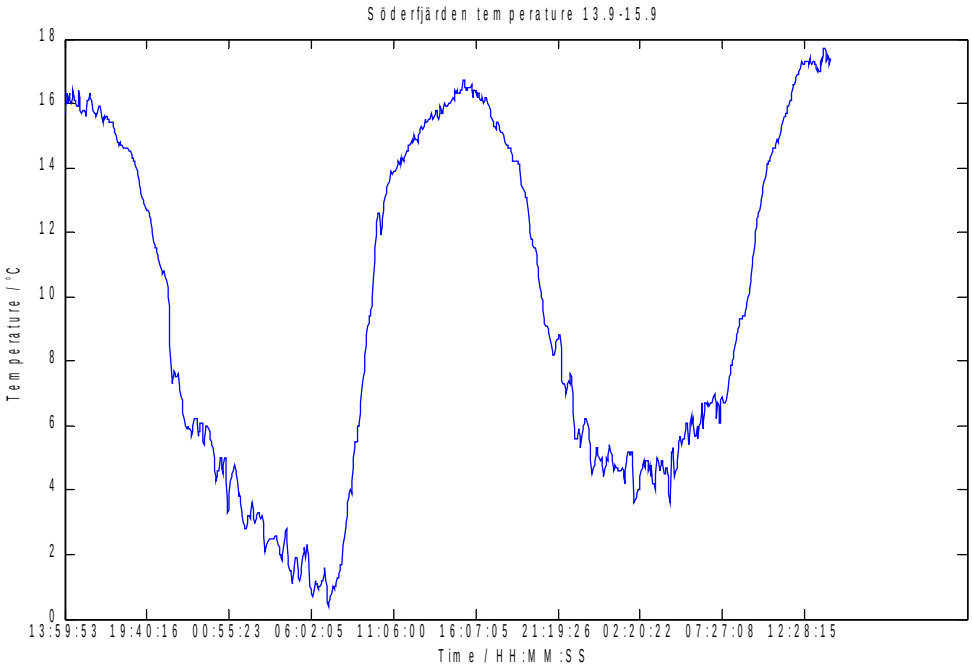


Figure 25. Temperature in Söderfjärden on September 13-15, 2009. Measurements are collected over the develop system.

7. DISCUSSION AND FUTURE WORK

7.1. Future Work

In this thesis the main task was to develop a sensor network system with a GPRS backbone connection to support remote control and data logging. Due to the limited time and the complexity of the system it took a long time to develop the hardware and software for the system. In order to extend the system into its full scale, energy consumption in a large WSN network should be tested.. Also scheduling of sleep periods, which are one crucial way to save energy in a large WSN, should be tested.

There are many things to develop further as a continuation of this thesis. One application that was requested by the Söderfjärden museum association was a local WSN node with LCD display that shows the current weather measurements from the weather station. It should be a rather simple task to implement this one.

Another small implementation that should be added is the monitoring of the battery voltages and current consumption in the museum electricity system. It is necessary to have it if an automated power control scheme is developed. In the design of these nodes one must investigate carefully what sensors to use. A simple solution would be to use a voltage divider and measure the voltage with an A/D converter.

More work is needed to integrate the developed system to the diesel generator's own control system. Not only the interface connection itself is challenging, but also the network performance to support ModBus communication inside the WSN.

In order to extend the WSN into a large scale network with many different measurement nodes, the WSN routing and scheduling should be developed further. When the scale of the WSN expands also the complexity of the network increases. If more and more critical data is transmitted inside the WSN also correction and re-transmission schemes should be developed to ensure reliable data transmission.

One critical task that should be implemented for a constant online system is logging of errors. In the current implementation no error logging is used which clearly complicates the debugging of faults. Every fault where abnormal behaviour occurs should be logged to file if it is not possible to send it directly to receiving server.

7.2. Project Summary

WeatherLAN project works as a pilot phase which results can be used on several other areas of sensor networks. The importance of a gateway increases as the locations become more difficult. For example, in the archipelago it is unlikely that a wired internet connection would be available for direct use.

When starting with the project the aim was to have a remote control application through long haul connection with Söderfjärden museum, where a local WSN should measure the surroundings. Those two things were completed. One thing that was also considered was to implement the remote control of the diesel generator as well, but in the end there where to much work for the first part of the project and therefore it was left out as further work to be done. Network planning was made such that it should be easy to implement new features in the network, by using the remote script update and routing possibilities.

The project has given much new impacts for many areas, such as network programming, PCB making and electronics planning. Earlier mentioned things where completely new to me when I started with the thesis and certainly delayed the completion of the thesis work. As can be seen from the results, the hardware platform was not tested well enough, which led to unexpected behaviours of the system. From these observations it should be clear that hardware testing should be investigated more for these so called always online systems. It is much more easy to develop a system that works for a short time and will be then taken away.

Up to this point WeatherLAN has only been tested as an experimental part of this masters thesis and several limitations have been noticed. To further extend and improve the

system more expertise from different engineering branches are needed, since the complexity of the system may easily expand out of the hands of one person to implement.

System view testing was first kept on a minimum level, which was noticed in the deployment phase. Even though everything was running fine in the laboratory, the different environment during deployment caused troubles not foreseen in the development phase.

WeatherLAN project has showed that a WSN in combination with a long distance link may be good solution for remote monitoring and control. Even though the system does not yet have all the features that were considered from the beginning, it clearly gives an overview that a WSN can be used for applications where remote control is needed.

REFERENCES

3GPP. (1999). *AT command set for User Equipment (UE) Release 99.* [Online]. Available: http://www.3gpp.org

Andersson, Christoffer. (2001). *GPRS and 3G Wireless Applications.* New York: John Wiley & Sons, Inc. 317 p. ISBN 0-471-41405-0

Contiki, 2009. *Contiki Operating System web page*. [Online.] Available: http://www.sics.se/contiki/

Eberspächer, Jörg. Vögel, Hans-Jörg. Bettstetter, Christian. (2001). *GSM Switching, Services and Protocols*. 2. edition. New Jersey etc.: John Wiley & Sons Inc. 296 p. ISBN 0-471-49903-X

ETSI Std. TS 100 916. (1999). *Digital Cellular System (Phase 2+); AT Command Set for GSM Mobile Equipment (ME)(GSM 07.07 version 7.4.0 Release 1998).*

Finnish Meterological Institute (FMI). (2009). *Helsinki test bed project page*. (Referred 28.7.2009).[Online]. Available: htttp://testbed.fmi.fi

IEEE Std. 802.15.4 (2003). *Standard for Part 15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specification for Low-Rate Wireless Personal Area Networks (WPANs). 1*. of September in 2003, New York, USA.679p.

Montenegro, Gabriel, Kushalnagar, Nandakishore, Hui, Jonathan, Culler, David. *Transmission of IPv6 Packets over IEEE 802.15.4 Networks*. IETF RFC 4944. [Online]. Available: http://www.ietf.org/rfc/rfc4944.txt.

National Semiconductor. (2004). *LM2576 series simple switcher 3A Step-Down Voltage Regulator*.

Philips Semiconductors. (2003). *74HC04 Hex Inverter*. [Online]. Available: http://www.datasheetcatalog.com/datasheets_pdf/7/4/H/C/74HC04.shtml

Postel, Jon. (1981). *Internet Protocol.* IETF Std RFC 791. [Online]. Available: http://www.ietf.org/rfc/rfc0791.txt

Postel, Jon. (1981). *Transmission Control Protocol.* IETF Std RFC 793. [Online]. Available: http://www.ietf.org/rfc/rfc0793.txt

Radiocrafts, (2009). *Radiocrafts webpage*. [Online]. Available: http://www.radiocrafts.com/

Rivest, Ronald. (1992). *The MD5 message-digest algorithm*. IETF RFC 1321. [Online]. Available: http://www.ietf.org/rfc/rfc1321.txt

Sensinode, (2007). *NanoStack manual version 1.1.0.*

Sensinode, (2009). *Sensinode Ltd, webpage*. [Online]. Available: http://www.sensinode.com.

Seurre, Emmanuel & Patrick Savelli & Pierre-Jean Pietri. (2003). *GPRS for Mobile Internet.* Norwood etc.: Artech House. 419 p. ISBN 1-58053-600-X.

Stallings, William. (2003). *Cryptogrpahy and Network Security - Principles and Practice*. *3*. edition New Jersey.etc: Pearson Education, Inc. 681p. ISBN 0-13-111502-2

Stallings, William. (2007). *Data and Computer Communications*. 8. edition New Jersey.etc: Pearson Education, Inc. 878 p. ISBN 0-13-243310-9

Telit (2007). *GM862-QUAD/PY Hardware User Guide*.

Telit (2008). *GM862-QUAD/PY Datasheet*.

Texas Instruments, (2008). *A True System-on-Chip Solution*

Vaisala Oyj. (2008). *WXT520 User Guide.*

8. APPENDIX 1

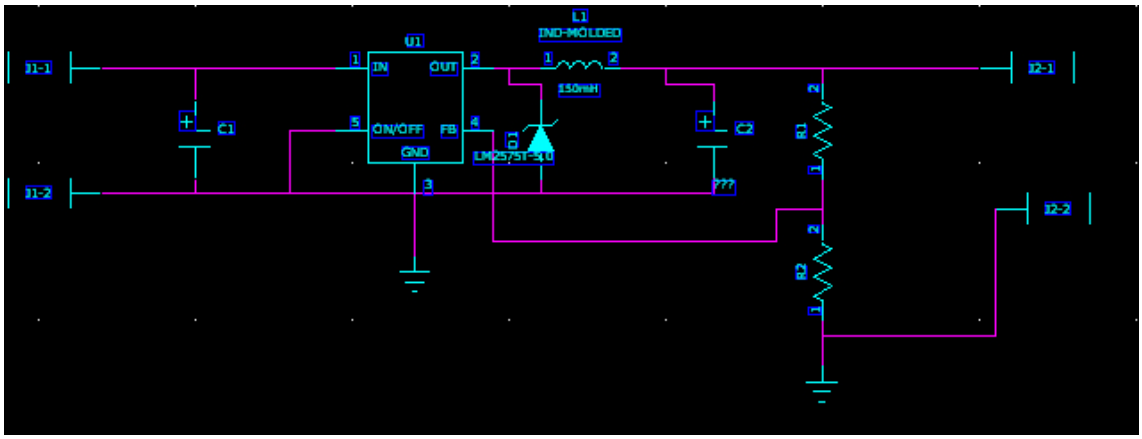


Figure 26. Power board logical layout.