2 State of the Art

2.1 Wireless Sensor Networks

Wireless Sensor Networks are networks of wirelessly communicating nodes which have the ability to measure physical variables. The individual nodes are usually embedded devices consisting of a microcontroller, memory, sensors, a wireless network interface, a programming interface and batteries. The development in the field of Micro Electro-Mechanical Systems (MEMS) has led to miniaturisation of the nodes down to the size of less than 1 cm^3 [Har+07].

The vision that is pursued by the WSN community is to connect the physical world with the virtual world [Est+02]. Information, i.e. measurements from real-world phenomena, is made available in a virtual world in order to benefit from this information. This vision can be applied in many application fields, e.g. Nature Monitoring, Smart Home/Office, Logistics, Agriculture, Traffic Management, and Healthcare.

In order to be economically and/or technically feasible WSN nodes are restricted with respect to the available resources, e.g. energy, processing power, communication bandwidth. The nodes therefore typically need to be custom designed in terms of hardware and software to the desired application. In this thesis the custom design is reduced for the software of the nodes and based on standard protocols which can be configured to meet the user's requirements.

The integration of WSNs and IP networks as an enabler to the integration of data into business processes is a precondition to a wide acceptance of Wireless Sensor Networks. WSNs in itself are useful – a tighter integration with IP networks would improve the applicability to more application areas, e.g. logistics.

While Wireless Sensor Networks (WSNs) are a recent and active research area, IP networks have been around for several decades, are widely accepted and used, and still under further improvement.

The interconnection between WSNs and IP networks has typically been performed by transport level gateways (e.g. TinyOS Serial Forwarder). The data from the WSN is extracted over the serial or USB port and can be fetched by other applications on a Transmission Control Protocol (TCP) port. Recently network layer gatewaying between WSN and the Internet has been made feasible, so that IP packets can be fed into WSNs. The IETF 6LoWPAN working group has made

M. Becker, *Services in Wireless Sensor Networks,* Advanced Studies Mobile Research Center Bremen, DOI 10.1007/978-3-658-05402-1_2, © Springer Fachmedien Wiesbaden 2014

Internet Protocol Version 6 (IPv6) in WSNs possible by RFC 4944 [Mon+07]. The advantages of the availability of an IP adaptation layer for WSNs are among others [SB10]:

- the interoperability with other IP networks (many control systems support IP as an option)
- the presence of well established tools for network management (e.g. ping, telnet, traceroute)
- trusted security solutions (firewalls, access control).

In the following the organisations, communities and standardisation activities which are relevant to this thesis and Wireless Sensor Networks (WSNs) will be presented.

2.1.1 Organisations and Standardisation Bodies

Professional organisations which are working in the field of WSNs are coming from standardisation bodies such as the IEEE, the IETF or from marketing alliances such as the ZigBee Alliance and the Internet Protocol for Smart Objects Alliance or the research community as well as indivual companies.

Institute of Electrical and Electronics Engineers

The IEEE has standardised the protocol 802.15.4 in 2003 [IEE03] and enhanced it in 2006 [IEE06], under the umbrella of the 802.15 working group. The standards describe the Physical and Medium Access Control Layer for Low-Rate Wireless Personal Area Networks (LR-WPANs). This standard is implemented in a variety of low-power chipsets, e.g. the Chipcon CC2420 and the Atmel AT86RF230.

The physical layer of 802.15.4 takes care of the listed functionalities:

- Energy Detection (ED) on the chosen channel
- Clear Channel Assessment (CCA) for implementing Carrier Sense Multiple Access / Collision Avoidance (CSMA/CA)
- Link Quality Indicator (LQI) on received packets
- Powering and unpowering the radio transceiver, channel switching and data transmission/reception

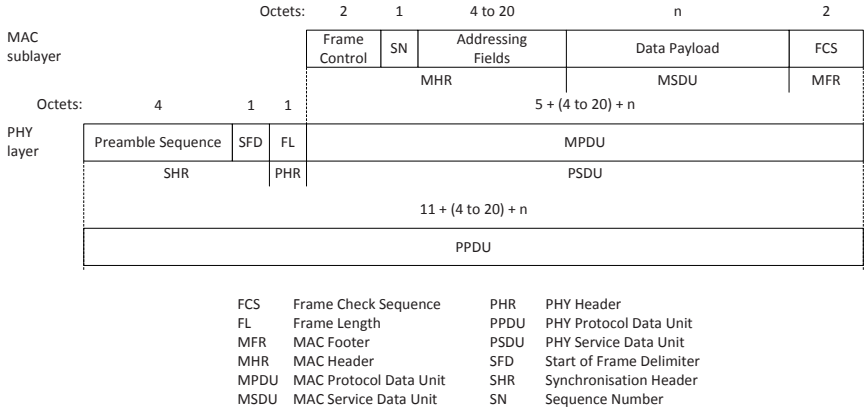


Figure 2.1: IEEE 802.15.4 frame format (From: [IEE03])

The MAC and Physical Layer (PHY) frame format of IEEE 802.15.4 is shown in figure 2.1.

The 802.15.4 standard defines the usage of two frequency bands at 868/915 MHz (continent dependent) and 2400 MHz (internationally). The lower band, although having a better radio propagation, is not easily usable for international logistics as the higher layers would have to detect in which location it is currently operated to meet the regulatory requirements (cf. Annex F of [IEE03] and [IEE06]). Additionally, only few radio chips are available for the lower bands. The 2400 MHz band, however, is internationally available and regulations on this band are more homogeneous, so that it can be seen as a good fit for internationally deployed WSNs in logistics.

The coding and modulation on those bands has originally been specified in the 802.15.4-2003 standard to be Direct Sequence Spread Spectrum (DSSS) with Binary Phase Shift Keying (BPSK) with 20/40 kbit/s datarate for the 868/915 MHz and DSSS with Offset Quadrature Phase-Shift Keying (O-QPSK) with 250 kbit/s on the 2.4GHz band. A constellation diagram of O-QPSK created with a Software Defined Radio (SDR) is shown in figure 2.2, while figure 2.3 shows the in-phase (I) and quadrature (Q) symbols for the start of an IEEE 802.15.4 packet transmission. The I symbols are modulated on the half-sine pulse shape carrier, while the Q symbols are delayed by half the chip duration and modulated on a cosine carrier.

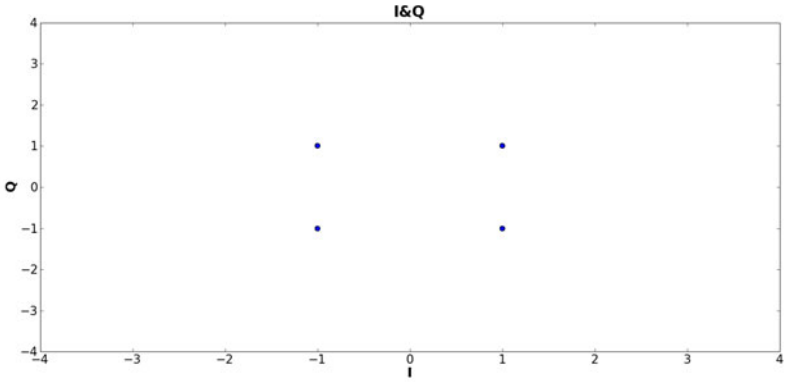


Figure 2.2: Constellation diagram of IEEE 802.15.4

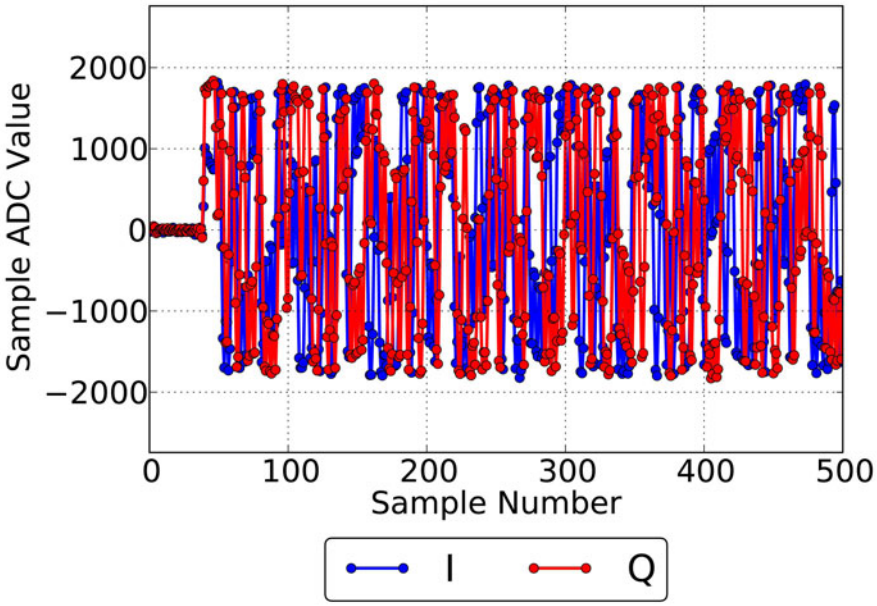


Figure 2.3: Received IEEE 802.15.4 I/Q DAC samples

The 2006 standard added DSSS with O-QPSK and Parallel Sequence Spread Spectrum (PSSS) with BPSK and Amplitude Shift Keying (ASK) modulation, thereby introducing higher data rates for the lower band [IEE06].

The MAC layer which is described by the standards mentioned above is usually not in use in programs based on TinyOS, though implementations exist [Hau09] for specific hardware [KH08]. In TinyOS typically a Carrier Sense Multiple Access (CSMA) with random backoffs and possibly a Low Power Listening (LPL) duty-cycle algorithm [ML08] is used.

Internet Engineering Task Force

The Internet Engineering Task Force (IETF) is standardising networking solutions on top of link-layer protocols in the form of RFCs. Due to the specific nature (i.e. low throughput, lossy links and small packet size) of the 802.15.4 layers, the adaptation of the widely used IP protocol needs to be specified. As solutions involving IEEE 802.15.4 based devices are likely to be deployed in the hundreds or thousands of nodes per installation, the addressing space available for the nodes needs to be of adequate size. The addressing space of Internet Protocol Version 4 (IPv4) [Pos80] is limited to less than 2^{32} and is currently almost depleted [Hus09] due to the inefficient usage of the addressing space. Thus IPv6 [DH95] with its address space of 2^{128} addresses was chosen to be the networking protocol to be adapted to the underlying link layer protocols, as detailed in the following paragraph.

The IETF Working Group 6LoWPAN has standardised the transmission of IPv6 packets in IEEE 802.15.4 Wireless Sensor Networks [Int; TZI]. The working group has created the problem statement document RFC 4919 [KMS07] and the specification of the frame format in the document RFC 4944 [Mon+07] and RFC 6282 [HT11].

The challenges of implementing IPv6 on WSN nodes are to meet the critical embedded wireless requirements of long lifetime under the constraint of limited energy, high reliability and adaptability as well as manageability of many devices with highly constrained resources (i.e. Central Processing Unit (CPU) power, memory size).

The advantages that are prevalent when using IP in WSNs include interoperability with existing IP network links (e.g. Ethernet and 802.11), security support by established mechanisms (e.g. Firewalls and Authentication Schemes), existing name-address translation (i.e. Domain Name Service (DNS)), inter-working with mobility protocols (e.g. MobileIP), end-to-end transport protocols (TCP and User Datagram Protocol (UDP) among others) and many application and manage-

ment protocols and applications (Hypertext Transfer Protocol (HTTP), Hypertext Markup Language (HTML), Simple Object Access Protocol (SOAP), ping, traceroute, etc.) [SB10].

The most important challenges imposed by layer 2 (IEEE 802.15.4) are the small Maximum Transmission Unit (MTU) of maximum 127 bytes while IP requires support for 1280 bytes MTUs, from which the need for fragmentation support and header (including addresses) compression can be inferred. This is handled by the definition of an adaptation layer. RFC 4944 defines the packet format for transmission of IPv6 RFC 2460 [DH98] frames. Additionally the creation of IPv6 link-local addresses as well as statelessly autoconfigured addresses for IEEE 802.15.4 networks are defined.

A basic UDP/IPv6/802.15.4 header has a length of 58 bytes, of which UDP/IPv6 make up 48 bytes. However, those headers contain redundant information across the layers, especially the length information and addresses (IPv6 addresses can be derived from 802.15.4 addresses), which can be removed (reduction by 20.5 bytes). The compression of commonly used values (flow labels, address prefixes, multicast addresses, UDP ports and checksum) can be compressed (reduction by 24.5 bytes). Thus in the best case the UDP / IPv6 header can be compressed to 6 bytes, according to [HC08]. In more detail, when using link-local unicast the UDP/IPv6 header is 6 bytes, when using link-local multicast 7 bytes and when using global unicast the header has a length of 10 bytes [HC08].

Routing 6LoWPAN allows for the forwarding and routing processes to be implemented at the link layer (termed 'Mesh Under') and at the network layer (termed 'Route Over').

Route Over In Route Over forwarding and path computation is performed by layer 3, i.e. the IP layer above the 6LoWPAN adaptation layer. Every node in this scheme is an IP router. The architectural view of this scheme is depicted in figure 2.4. With this option the IP link spans only over one-hop radio transmission range.

Mesh Under In Mesh Under forwarding and path computation is performed by layer 2, so either in the 6LoWPAN adapation layer or in the data link layer below the adaptation layer (both options are possible). Every node in this scheme is an 802.15.4 switch. The architectural view of this scheme is depicted in figure 2.5. With this option the IP link spans over several radio transmission ranges. Mesh Under emulates transitivity across the link for the IP layer, which means that if

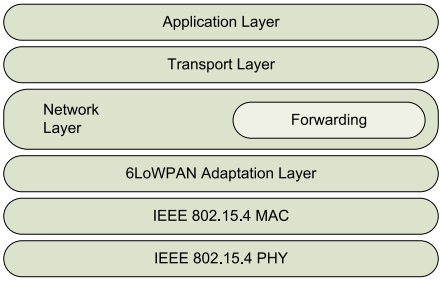


Figure 2.4: 6LoWPAN Routing Options: Route Over

node A and node B as well as node B and node C can communicate, then node A and node C can communicate as well.

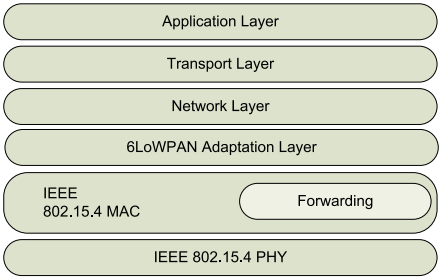


Figure 2.5: 6LoWPAN Routing Options: Mesh Under

6LoWPAN header format The 6LoWPAN frame format starts with a Dispatch byte. The various Dispatch options are shown in fig. 2.6.

If the Dispatch byte begins with '00', the packet is not a 6LoWPAN frame. If the first two bits are '01', a normal IPv6 addressing header is following. When '10' are in the first two bits, a mesh header is following. In the case of '11' in the first bits, a fragmentation header is following.

A normal IPv6 addressing header can have uncompressed or various compressed modes for the addressing and other header information.

The mesh header allows for the 'Mesh Under' mode of 6LoWPAN relying on meshing capabilities of the underlying protocols.

The fragment header is present when the service data unit and the 6LoWPAN header would not fit into one link-layer packet and indicate the fragment number.

If multiple of these headers are present in one frame, the order is mesh, fragment and then normal header.

Implementations Implementations of 6LoWPAN are available from ArchRock [Arc07], Sics (SicsLowpan) [Dur+08], Jacobs University [HS08], Berkeley University (blip) [Jia+09; HC08], SensiNode and others.

The specification of a routing protocol has been left out of the 6LoWPAN working group and is performed in the Routing Over Low power and Lossy networks (ROLL) working group. The working group ROLL started by assembling requirements documents for several application domains, i.e. for Home Automation [BBP10], Industrial Applications [Pis+09] and Urban Environments [Doh+09]. Additional requirements of 6LoWPAN on the routing protocol were stated in [Kim+12].

ROLL has specified a routing protocol for Low power and Lossy networks (LLN), termed IPv6 Routing Protocol for Low power and Lossy Networks (RPL), in the standardisation documents from RFC 6550 up to RFC 6554 [Win+12; Vas+12; Thu12; HV12; Hui+12].

The Trickle algorithm which is modelled and studied in this thesis has been standardised by the ROLL group in RFC 6206 [Lev+11]. The algorithm is used for the distribution of so called Destination Oriented Directed Acyclic Graph Information Object (DIO) messages, which contain routing information that is necessary for the discovery of routes towards the RPL root node, which is in most cases also the 6LoWPAN border router.

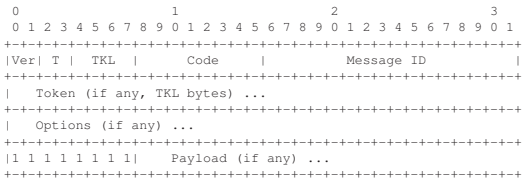
The Constrained RESTful Environments (CoRE) working group of the IETF is working on the standardisation of an application level protocol for constrained nodes and networks (such as WSNs) termed Constrained Application Protocol (CoAP), which is based on the Representational State Transfer (REST) architecture. The current status of the CoAP protocol is specified in [She+13]. The protocol is based on UDP, implements its own retransmission scheme and has a more compact representation than HTTP, while also defining proxying between HTTP and CoAP.

Using link-formats under a well-known Uniform Resource Identifier (URI) (i.e. /.well-known/core) allows for the discovery of other available resources. Resource discovery and service discovery (as described in this thesis) complement each other. E.g. service discovery can be used to discover the nodes while resource discovery is then used to discover the resources on the nodes. The service discovery described in this thesis additionally can perform in a fully decentralised fashion. Whereas the resource discovery follows the client-server model more strictly.

In the following the main data formats and identifiers of CoAP are summarised

based on [She+13]. The extended version of this summary can also be found in [Bec13].

The CoAP Message Format is shown in listing 1, where Ver denotes Version, T denotes Type, and TKL stands for Token Length. Code either includes the CoAP method code in requests or the CoAP response code in a response. The Message ID field gives a unique identifier for a CoAP message to be matched with the response. A token is used for CoAP response matching when the response is separate response with a different Message ID. Options might be included in the CoAP messages, the end of the options is marked by a payload marker of eight '1' bits (0xFF). The payload is appended at the end.



Listing 1: CoAP Message Format (Source: [She+13])

The method types in the CoAP message formats' field T can be the values listed in listing 2.

Type	Name
0	CONfirmable
1	NON-confirmable
2	ACKnowledgement
3	ReSeT

Listing 2: CoAP Method Types (Source: [She+13])

The CoAP method codes for requests are similar to the HTTP verbs and are listed in listing 3. The method codes are included in the field Code of the CoAP message format.

Code	Name
1	GET
2	POST
3	PUT
4	DELETE

Listing 3: CoAP Method Codes (Source: [She+13])

In listing 4 the structure of CoAP codes for responses is shown, while in listing 5 the classes of response codes and in listing 6 the various codes and the equivalent HTTP code and meaning are listed. The response code can be found in the field Code of the CoAP message format.

```
 0
 0 1 2 3 4 5 6 7
+-+-+-+-+-+-+-+-+
|class|  detail |
+-+-+-+-+-+-+-+-+
```

Listing 4: CoAP Response Code Structure (Source: [She+13])

Class	
2.xx	Success
4.xx	Client Error
5.xx	Server Error

Listing 5: CoAP Response Code Classes (Source: [She+13])

http://www.springer.com/978-3-658-05401-4

Services in Wireless Sensor Networks
Modelling and Optimisation for the Efficient Discovery of Services
Becker, M.
2014, XXVII, 179 p. 99 illus., 7 illus. in color., Softcover
ISBN: 978-3-658-05401-4