Energy Consumption of 6LoWPAN and Zigbee in Home Automation Networks

Falk-Moritz Schaefer, Tobias Groß and Rüdiger Kays
TU Dortmund, Communication Technology Institute
Dortmund, Germany
{falk.schaefer|tobias.groß|ruediger.kays@tu-dortmund.de}

***Abstract*— Current home networks are often split up in high data rate networks for multimedia and internet applications and low data rate networks for home automation. This separation leads to inefficient and cumbersome network configurations. It also hinders the widespread use of home automation networks. The IPv6 adaptation for resource constrained devices can help to realize a common addressing scheme for the complete home network. Often the overhead of IPv6 is considered as a drawback for its utilization in home automation networks with low data rates. In this paper, the overhead and the energy consumption of 6LoWPAN in an exemplary home network is calculated and compared with Zigbee. Assuming efficient configurations of both protocol stacks only small differences can be observed. Additionally, the energy consumption of both 6LoWPAN and Zigbee reference implementations on the same hardware is evaluated. Both protocols have comparable energy consumption. For an energy efficient home automation network an intelligent network configuration with only a small number of routers is most important.**

Keywords*— Zigbee; 6LoWPAN; home automation; wireless network; wireless sensor network; energy efficiency; energy consumption

I. INTRODUCTION

Applications like home automation, ambient assisted living and e-health have been discussed for a long time. They require a reliable and convenient home automation network (HAN). Additionally, new applications envisioned in the smart grid require access to devices within the home. In the domain of home automation, there are many incompatible solutions. On top of that, the home network is basically split up in the domain of high data rate technologies for consumer electronics products on the one hand and low data rate technologies for home automation applications on the other. Usually different and independent networks for both applications domains operate independently of each other within a single home. The question arise how a future proof home network for all kind of applications can be designed that combines both network domains into a heterogeneous home network infrastructure.

The techniques of 6LoWPAN [1] can help to integrate the HAN into existing high rate home networks. The realization of a common addressing scheme between both HAN device and high rate devices can simplify the integration process. It can be argued that in HANs there is no need for a globally unique IP address with all the overhead, and that the IP protocol can never fulfill the requirements of networks with resource constraint devices. However, the development of middleware and applications is simplified. New routing mechanisms can be utilized, which allow for more flexible traffic forwarding and a closer interaction of HAN and the high rate network.

Various comparisons between networking stacks for wireless sensor networks in the home environment are available [2-7]. In this paper, we analyze the energy consumption of an exemplary HAN realized with 6LoWPAN and compare it to the consumption of a Zigbee network of the same topology. Furthermore, the energy consumption of two protocol stack implementations with the same hardware is measured and evaluated.

II. PROTOCOL STACKS

Zigbee and 6LoWPAN both use the physical layer (PHY) and medium access control layer (MAC) of IEEE 802.15.4. In Fig. 1 the MAC layer frame structure is depicted. In beacon-less PANs, only end devices can use the sleep mode. They wake up periodically and request data from the corresponding coordinator. The Zigbee network layer frame format is depicted in Fig. 1, too. When a frame is relayed either Tree Routing or Source Routing is used. With tree routing the frame is routed up the tree, until a parent can address the destination node. Every coordinator that joined the PAN is assigned a certain address range, from which it can assign addresses to associating nodes (distributed address assignment). This way, the address of the parent node can be derived from the node's address. 6LoWPAN adapts the IPv6 header to resource constrained low power wireless personal area networks. Header compression and fragmentation are supported, so that the overhead of an IPv6 packet is decreased (cf. Fig. 1).

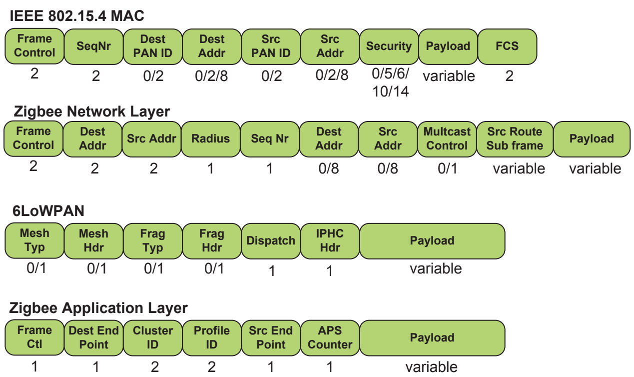

Fig. 1. Example of a figure caption.

As an example application layer protocol, the Zigbee application layer is considered. This includes device and service discovery and some kind of application layer identification, which is done in ZAL with the help of end points, groups, clusters and profiles. A common data packet includes eight byte application layer overhead (Fig. 1). For some applications an acknowledgement (ACK) on application layer with eight bytes is useful.

III. EVALUATION

A. Example Scenario

We consider a medium size two story house environment comparable to [8]. It is populated with different kinds of devices. Some of them are battery powered, and some are mains-operated. We consider a transmission in the 2.4 GHz ISM band with 250 kb/s data rate. To have a comparable situation, the same network topology (cf. Fig. 2) is assumed, i.e. the same parent-child relations are present in both cases. Zigbee and 6LoWPAN both use tree routing mechanisms. For simplification, we consider no packet errors. Additionally, we assume that the PAN coordinator assigns short addresses after association and link layer ACKs are used. For the 6LoWPAN network we consider the storing mode and stateless header compression. The needed distribution of the global prefix is done by RPL [9]. In the end, a 6LoWPAN header is two bytes long for a single hop connection. In our scenario, there are not many nodes in the network. So the required number of consistent transmissions, that avoids a transmission within the trickle based transmission, is not reached. The channel list for active scanning and networks setup includes seven channels [10]. The Scan Duration parameter has the default value of four.

Most envisioned HAN applications require the exchange of frames with lengths between 25 bytes and 100 bytes [11]. We expect the following application characteristics: packets carry 25 bytes of application payload. The meters are read out two times a day. The washing machine and the dish washer send out two status information per day followed by an application layer ACK. The temperature sensors transmit information 48 times a day. The plugs are switched on or off four times a day with application layer ACKs. The fridge is controlled remotely ten times per day. The display devices request information from the aggregating device, e.g. implemented in the internet gateway, four times a day with application layer ACK coming from a request packet and a data update packet. Overall, there are 18 nodes. Four nodes act as router and remain in RX mode when not transmitting. Six nodes only transmit periodically sensor data and sleep the rest of the time. Eight nodes request data every second from the parent device and sleep the rest of the time.

In Zigbee networks the response to the association request already includes the short address. The Zigbee network layer basically does not need any more configuration information for tree routing. The 6LoWPAN device still needs information about the global prefix and configuration information for the DIO transmission. Additionally, 6LoWPAN does not use a distributed address assignment for the short addresses. This is why short addresses must be assigned by the PAN coordinator.

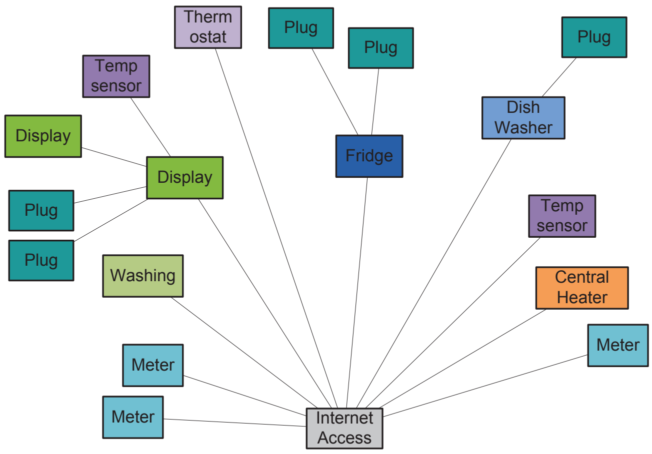

Fig. 2. Scenario network topology

Due to the relatively stable environment in our example, we choose a registration lifetime of one day, so that the messages are repeated once a day. DIO messages are transmitted every 2.3 hours by routing devices. In our example, there are at least four routing devices.

Considering the overhead, 6LoWPAN periodically sends DIOs and reregisters the network address. But 6LoWPAN has smaller network layer headers and therefore requires less bytes for the transmission of payload frames. Since our scenario includes low data rate applications, Zigbee requires only 50 percent of the overhead compared to 6LoWPAN, if only the network layer is considered. However, in both cases approx. 80 percent of the overhead is due to IEEE 802.15.4 MAC, PHY and application layer, which is independent of the network protocol.

We evaluate the energy consumption of the wireless transceiver only. The energy consumed depends on the power consumption in different states and the time the transceiver stays in each state. We used an energy consumption model based on [12]. As a reference we take the AT86RF231 transceiver. We assume that smart meter and temperature sensor wake up periodically to transmit the latest measurement data. The rest of the time, the nodes are in sleep mode. The router must stay in RX mode all the time. The washing machine, the display, the central heater and the plugs wake up every second to request data from the connected node, in order to respond to control requests. The rest of the time, they stay in sleep mode. The energy consumed for the transmission of application layer packets sums up to 0.067 Ws for Zigbee and 0.064 Ws for 6LoWPAN. The regular transmission of DIOs by all routers in case of 6LoWPAN requires 0.01 Ws per day. Zigbee does not include such a periodic transmission. Therefore, there is no power consumption due to network layer management in Zigbee networks in case of a stable environment. The overall network energy consumption of the network is approx. 13 kWs per day for both Zigbee and 6LoWPAN. The main part comes from router devices, which stay in RX mode all of the time.

B. Experimental Comparison

We evaluated the Zigbee and the 6LoWPAN stack in a test bed implementation. We used an ATmega128RFA1 microcontroller, which integrates an 8 bit microcontroller and an IEEE 802.15.4 2.4 GHz RF transceiver. We used the BitCloud system as a reference implementation of the Zigbee Pro Stack. It supports the configuration as end device, router and PAN coordinator. It is important to notice, that the Zigbee Pro implementation includes the transmission of Link Quality Indications (LQI) at regular intervals. We used the Contiki OS as a reference implementation of the 6LoWPAN stack. Only small modifications to our platform were necessary. The configuration parameter however must be chosen carefully in order to build a stable home network. We used a simple UDP protocol for the exchange of application data. ContikiOS was configured to use the nullmac_driver with link layer acknowledgements. The expected transmission count was used as a RPL routing metric. Additionally, we evaluated the Contiki implementation of the ContikiMAC protocol [13] with an improved phase calculation. The basic idea of the ContikiMAC is that all nodes wake up periodically and sense the channel. If they sense certain receive power, they go in RX mode. Otherwise, they sleep again. A device with a sending request memories the intervals of neighbors and predicts the time the destination will wake up again. It starts the transmission a little bit earlier than predicted and repeats the frame until it receives an ACK. For a proper working radio duty cycling (RDC) mechanism it was necessary to disable the automatic packet repetition of the transceiver. There are eight sleep periods per second. Both the transceiver hardware and the CPU core are set to sleep mode.

One general advantage compared to the BitCloud implementation is, that even a Router device can go to sleep. Both protocol stacks use the long addressing scheme of IEEE 802.15.4. BitCloud operates at 8 MHz, whereas Contiki uses 16 MHz This is implementation specific and is not linked to any timing or computing power constraints. We measured the drawn true root mean square current of the microcontroller under different node configurations over time. One measurement takes 2.2 seconds with a sample rate of 436.9 Hz. We averaged over multiple measurements, which are repeated randomly between 5 and 15 seconds. We used a voltage regulator with a low standby current of 50 µs and removed all unnecessary components of the evaluation hardware. All devices sent a data frame every five seconds directly to the coordinator. Additionally, the BitCloud Enddevice requested data every ten seconds from the coordinator. We measured the power consumption shown in Table I. The difference between BitCloud Router and Contiki is due to the different clock speeds of the microcontroller. The most efficient configuration is the BitCloud Enddevice. However, such a configuration is only realistic for devices without user feedback.

TABLE I. POWER CONSUMPTION OF SOFTWARE CONFIGURATIONS

Type	Config	PC [mW]	Config	PC [mW]
BitCloud	Router	55.27	Enddevice	6.63
Contiki	No Sleep	61.97	ContikiMAC	11.4

IV. CONCLUSION

We compared the overhead of the 6LoWPAN and the Zigbee network layer in an exemplary home network scenario and calculated the overall energy consumption of the network. Assuming the most efficient, standard compliant configurations of both 6LoWPAN and Zigbee, 6LoWAN needs a little bit more overhead for network organization, whereas Zigbee needs more overhead for the network header. In the end, considering a home network scenario the energy consumption of both protocols diverge only slightly. Furthermore, we evaluated the energy consumption of two reference implementations on the same hardware. Again only small differences between 6LoWPAN and Zigbee could be observed. Instead, it is obvious that the network topology and the configuration of network nodes have a major impact on the overall home network energy consumption. Our results show that the network layer overhead of 6LoWPAN is no obstacle to its employment in home automation networks. 6LoWPAN can be one tool to realize a uniform addressing scheme for low rate networks as well as high rate networks. It might help to interconnect both home networking domains and to enable a convergent future-proof home network infrastructure.

REFERENCES

[1] IPv6 over Low-Power Wireless Personal Area Networks (6LoWPANs): Overview, Assumptions, Problem Statement, and Goals, IETF RFC 4919, August 2007.

[2] Gomez, C.; Paradells, J.; , "Wireless home automation networks: A survey of architectures and technologies," Communications Magazine, IEEE , vol.48, no.6, pp.92-101, June 2010.

[3] Rathnayaka, A.J.D.; Potdar, V.M.; Kuruppu, S.J.; , "Evaluation of wireless home automation technologies," Digital Ecosystems and Technologies Conference (DEST), 2011 Proceedings of the 5th IEEE International Conference on , vol., no., pp.76-81, May 31 2011-June 3 2011.

[4] Hui, J.W.; Culler, D.E.; "Extending IP to Low-Power, Wireless Personal Area Networks," Internet Computing, IEEE , vol.12, no.4, pp.37-45, July-Aug. 2008.

[5] Zimmermann, A.; Sa Silva, J.; Sobral, J.B.M.; Boavida, F.; , "6GLAD: IPv6 Global to Link-layer ADdress Translation for 6LoWPAN Overhead Reducing," Next Generation Internet Networks, 2008. NGI 2008 , vol., no., pp.209-214, 28-30 April 2008.

[6] Geoff Mulligan. 2007. The 6LoWPAN architecture. In Proceedings of the 4th workshop on Embedded networked sensors (EmNets '07).

[7] Jonathan W. Hui and David E. Culler. 2008. IP is dead, long live IP for wireless sensor networks. In Proceedings of the 6th ACM conference on Embedded network sensor systems (SenSys '08).

[8] Zigbee HomePlug Joint Working Group, Smart Energy Profile Marketing Requirements Document (MRD) Draft 1.0, Mar 2009.

[9] RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks, IETF RFC 6550, March 2012.

[10] Zigbee Home Automation Public Application Profile, Zigbee Alliance, Revision 26, Version 1.1, February 2010.

[11] SG Network System Requirements Specification v4.0.

[12] N. Langhammer, R. Kays, "Evaluation of wireless smart metering technologies in realistic indoor channels," ISWCS, pp.387-391, 6-9 Nov. 2011.

[13] Adam Dunkels, “The ContikiMAC radio duty cycling protocol,“ Technical Report T2011:13, Swedish Institute of Computer Science, 2011.