Real-time Enabled IEEE 802.15.4 Sensor Networks in Industrial Automation

Feng Chen*†, Thomas Talanis†, Reinhard German* and Falko Dressler*

*Dept. of Computer Science, University of Erlangen, Germany
†Siemens AG, Industry Automation Division, Germany

Abstract—**Sensor networks have been investigated in many scenarios and a good number of protocols have been developed. With the standardization of the IEEE 802.15.4 protocol, sensor networks became also an interesting topic in industrial automation. Here, the main focus is on real-time capabilities and reliability. We analyzed the IEEE 802.15.4 standard both in a simulation environment and analytically to figure out to which degree the standard fulfills these specific requirements. Our results can be used for planning and deploying IEEE 802.15.4 based sensor networks with specific performance demands. Furthermore, we clearly identified specific protocol limitations that prevent its applicability for delay bounded real-time applications. We therefore propose some protocol modifications that enable real-time operation based on standard IEEE 802.15.4 compliant sensor hardware.**

I. INTRODUCTION

Wireless technology successfully started its way into many industrial application fields including industrial automation. IEEE 802.15.4 [1], [2] is a standard designed for low-rate Personal Area Networks (PANs). In contrast to WiFi, which is standardized within IEEE 802.11 family, IEEE 802.15.4 based Wireless Sensor Networks (WSNs) are designed for low-rate applications, however they especially stress energy efficiency. Their ease of deployment and the widespread use makes WSNs also attractive for a number of commercial, especially industrial applications. For example, the Siemens Industry Automation Devision is currently evaluating such wireless technologies for use in automation environments. Since the data rates involved in this application domain are low, the use of WiFi technology according to IEEE 802.11 would be a waste of resources. Additionally, the untethered use of sensor nodes suggest IEEE 802.15.4 as a cost-efficient design alternative, since they are specifically designed to support communication over short ranges with low data rate and reduced energy consumption. As for example discussed by Willig [3], the IEEE 802.15.4 standard [1] has become a recognized industry standard and, thus, well accepted by industrial users. It provides specifications for the Physical Layer (PHY) and Medium Access Control (MAC) sublayer. Products that implement this standard are commercially available at an acceptable low cost. One of the main design goals of these standards has been energy efficient operation, whereas hard real-time aspects were not a primary concern. IEEE 802.15.4 is also being used to define more complex network protocols such as WirelessHART [4], which has its primary roots in wired industrial networks.

In this paper, we study the applicability of IEEE 802.15.4 based solutions in industrial automation focusing on its real-time capabilities. In general, our aim is to verify whether the protocol suits all the demands in industrial automation fields. Based on this more general performance evaluation [5], we were able to clearly identify a number of shortcomings in the protocol definition. Therefore, in a second step, we evaluated the protocol with analytical methods focusing on its capabilities for real-time operation. Again, we show that the protocol specification does not fulfill industry demands for low-latency transmission in terms of guaranteed delay bounds. Therefore, we propose modifications of the standard to circumvent these limitations. In all these simulations and analyses, we closely keep to specific requirements relevant to industrial sensor network applications, which we outline in the following subsection.

II. AN INDUSTRIAL CASE STUDY

In our performance analysis, we focus on limitations w.r.t. industrial application. A number of relevant metrics have been pointed out by Willig [3]. According to this study, the main characteristics of industrial traffic become visible in the following properties: the presence of deadlines, i.e. the need to support real-time communication; high reliability requirements regarding the successful transmission of single messages; the predominance of short packets, e.g. single sensor readings. According to the recent efforts by leading industrial automation companies and their standardization efforts,[1] the application scenario to be studied in this paper is typical for factory automation. We specifically focus on a well planned industrial environment, which can be considered a typical case. In factory automation, planning tools are used to ensure proper signal distribution between the deployed nodes [6]. Such tools are usually relying on raytracing methods as studied since a decade in the field of wireless LAN.

As stated before, the main focus is now the real-time behavior of the wireless communication. The selected scenario is similar to the one used in the forthcoming IEEE 802.15.4e standard [7]. The requirements are different from those discussed and studied recently in the sensor networking community, however, they are similarly challenging on a different level, i.e. real-time behavior and reliability. The assumption in factory automation is that a number of sensor nodes are scattered within an area and associated to a central node to form a star network, which is continuously monitoring industrial processes. Thus, the following requirements must be met by the MAC protocol – the numbers in brackets are examples from typical automation projects of the Siemens Industrial Automation Division:

- n nodes in a star topology ($n = 20$)
- very short alarm or sensor messages (1 Byte)
- guaranteed low latency delivery ($d_{GUA} < 10$ ms)

III. ANALYTICAL PROTOCOL EVALUATION

In the following, we denote the length of protocol elements as $l_{element}$ measured in symbols and the corresponding delays as $d_{element}$ in seconds.

The beacon interval (l_{BI}) consists of the following fields: a beacon (l_B), a Short Interframe Space (SIFS) (l_{SIFS}), the Contention Access Period (CAP) (l_{CAP}), a number $n \leq 7$ of Guaranteed Time Slots (GTSs) ($n \times l_{GTS}$), and the inactive portion (l_{SLP}). Each GTS is composed of an integer number m of superframe slots ($m \times l_{SS}$) and should accommodate at least one complete transaction (l_{TR}), including one data transmission (l_D) and a SIFS (l_{SIFS}). Thus, the length of a beacon interval can be calculated as follows:

$$l_{BI} = l_B + l_{SIFS} + l_{CAP} + n \times l_{GTS} + l_{SLP} \qquad (1)$$

For a certain scheme of GTS allocation, the guaranteed latency, which is measured by the maximum latency among all the GTS transmissions under all traffic conditions, can be estimated through analyzing the worst case. The worst case would happen in the network under the following constraints: A message is generated at a device during its own GTS slot. At this time, the device cannot transmit the message immediately and must buffer the message. The buffered message must wait for one beacon interval until the start of the corresponding GTS in the next superframe. Then, it needs one transaction period to get transmitted. Therefore, the guaranteed latency, denoted as l_G, under the worst case is bounded by the sum of one beacon interval and one transaction period, which is formulated as follows:

TABLE I
DURATION PARAMETERS

Symbol	Description	Value
l_B	length of beacon transmission	34 symbols
l_D	length of data transmission	40 symbols
l_{SIFS}	short interframe space	12 symbols
l_{TR}	length of one transaction	52 symbols

$$l_G = l_{BI} + l_{TR} \qquad (2)$$

Beacon Order (BO) is set equal to the Superframe Order (SO) to eliminate the inactive portion. Furthermore, the active portion, which is determined by SO, must be set long enough to accommodate seven GTSs in the Contention-Free Period (CFP) and to maintain a minimum CAP length of 440 symbols, denoted as l_{minCAP}, according to the protocol standard. Based on these rules, some duration values calculated according to the standard and listed in Table I are used to choose the minimum (BO,SO) combination. If both, BO and SO, are set to 0, l_{BI} is equal to 960 symbols. The resulting l_{SS} of 60 symbols is larger than l_{TR}. Therefore, one GTS l_{GTS} is allocated with one superframe slot and equals to 60 symbols. According to Equation 1 and the rule of minimum CAP, the minimum required beacon interval equals to $l_{minBI} = l_B + l_{SIFS} + l_{minCAP} + 7 \times l_{GTS} = 906$ symbols. This is smaller than the actual beacon interval of 960 symbols. The guaranteed latency can be calculated according to Equation 2 using $l_{BI} = 960$ symbols. Operating in the 2.4 GHz band, the available bandwidth of IEEE 802.15.4 is 62.5 ksymbols/s. Thus, the worst case latency $d_G = \frac{l_G}{62.5\,\text{ksymbols/s}} = 16.2\,\text{ms}$. Obviously, this result does not satisfy the requirement of 10 ms, even though only seven devices are considered in the network.

In summary, we identified the following limitations for the GTS mechanism in IEEE 802.15.4 protocol. First, the constraint of *maximum seven GTSs* limits the number

---

[1] The same scenario is being investigated in research devision from Siemens and ABB, just to name a few, with focus on mid-term integration in commercial applications.

of devices involved in the GTS usage. The length of the active period determined by SO also has an impact on the number of available GTSs per beacon interval. Once the capacity of GTS allocations is full, other devices desiring for GTS slots have to wait until some of the previously allocated GTSs have been released. The allocation and deallocation process would consume a considerable time, which would be intolerable for real-time applications. Secondly, the *minimum CAP length* of $440\,\mathrm{symbols}$ defined by the standard further restricts the available length of the CFP for GTS allocation, especially if SO is small. If only few transmission occurs in the CAP, such required minimum CAP will introduce an extra latency to GTS transmissions. Finally, one allocated GTS can only consist of an integer number of superframe slots. The length of one superframe slot calculated by 2^{SO} × *aBaseSlotDuration* grows exponentially with an increasing SO. This leads to an inefficient bandwidth use in those applications, in which a longer length of the active period with larger SO is required to enable more GTS allocations. For example in our studied case, 20 GTS allocations are required, however, a small bandwidth fraction is sufficient for transmitting alarm messages with only one byte payload within a GTS.

IV. IMPROVED LOW-LATENCY PROTOCOL

In the following, we present an IEEE 802.15.4 based protocol version that we improved for low-latency industrial applications. Our protocol version is explicitly designed for the industrial real-time application described in Section II and represents a basis for the standardization of IEEE 802.15.4e [7]. Without loss of generality, we started with the specific requirement of supporting 20 nodes in a star topology and a guaranteed upper latency bound of $10\,\mathrm{ms}$. However, we show that the protocol is able to support any (small) number of devices with a constant latency requirement per additional node. In order to achieve better hardware compatibility, the IEEE 802.15.4 PHY layer is completely preserved. Our improvements on the IEEE 802.15.4 MAC sublayer mainly include two aspects, the modification of the superframe structure and the reduction of the MAC overhead.

1) TDMA-based Superframe Structure: Each superframe consists of an IEEE 802.15.4 compliant beacon, n GTSs and $n+1$ Short Interframe Spaces (SIFSs). The frame structure is shown in Figure 1. We completely removed the contention access period, therefore, it is no longer possible requesting a GTS from the PAN coordinator in a request-reply fashion as defined in the standard. Instead, all GTSs need to be preallocated to each of the n devices. In our application, only uplink transmissions, in which devices transmit alarm messages to the PAN coordinator without acknowledgments, will occur in the GTS. Therefore, we distinguish two interframe space types. A SIFS with length $l_{SIFS} = 12\,\mathrm{symbols}$ (equal to *aTurnaroundTime* symbols in the standard) is used before and after the beacon frame to guarantee that radios of the PAN coordinator and devices can switch between RX and TX state. For the interframe space between neighboring GTSs, an Extra Short Interframe Space (XSIFS) with a shorter length $l_{XSIFS} = 4\,\mathrm{symbols}$ is defined. We assume that this value is long enough for two consecutive transmissions, because the PAN coordinator always stays in a receiving state during this period. In the enhanced protocol version, the interframe space has been separated from the GTS, which differs from the way described previously, and each GTS (l_{GTS}) will be allocated with only the length of one data transmission (l_D).

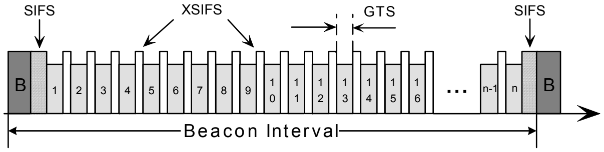

Fig. 1. TDMA-based superframe structure

In the used star topology, the communication is initiated by the PAN coordinator through broadcasting a beacon frame, which carries the information of the deployed superframe structure including the beacon interval and the position of the GTS preallocated to each device. Upon reception of the first beacon, each device knows all the settings of the superframe as well as its own GTS and has the following two options:

Beacon tracking enabled – The device keeps in sync with the PAN coordinator through tracking the beacons transmitted periodically by the PAN coordinator.

Beacon tracking disabled – In order to save as much energy as possible, the device can go to sleep immediately after receiving the first beacon. To transmit the message, the device needs first to resynchronize to the PAN coordinator by tracking the next coming beacon. Upon reception of one beacon, the node can locate its own GTS and send the message within this GTS.

2) Data Frame Format without MAC Header: The IEEE 802.15.4 standards adds a relatively large overhead of $38\,\mathrm{symbols}$ at the MAC and the PHY sublayers. We propose a new data frame format at the MAC layer that only includes a payload of one byte and a Frame Checksum (FCS) field with $2\,\mathrm{octets}$ in length. The

IEEE 802.15.4 MAC header is completely abandoned, resulting in an alarm message with only 9 octets including the PHY header and l_D. Compared to the original length of 20 octets, the overhead in the alarm message has been significantly reduced.

V. EVALUATION

According to Figure 1, Equation 3 has to be used to calculate the new beacon interval. l_{SIFS} and l_{XSIFS} are set to 12 symbols and 4 symbols, respectively. l_B remains at 34 symbols. l_{GTS} is set to $l_D = 18$ symbols.

$$l_{BI} = l_B + 2 \times l_{SIFS} + n \times l_{GTS} + (n-1) \times l_{XSIFS} \quad (3)$$

Based on this, generally d_G can still be calculated according to Equation 2: $d_G = \frac{l_{BI}+l_{TR}}{62.5\,\text{ksymbols/s}}$. According to Equation 3, l_{BI} equals to 494 symbols for 20 devices. The guaranteed latency needs to be evaluated according to the two protocol options:

Beacon tracking enabled – If the device keeps tracking the beacons, no extra latency will be spent on searching for the beacon. The worst case for this option has been discussed in Section III, and the guaranteed latency bound can be calculated. l_{TR} is the sum of l_D and l_{XIFS}, i.e. $l_{TR} = 18\,\text{symbols} + 4\,\text{symbols} = 22\,\text{symbols}$. Thus, the calculated guaranteed latency bound for 20 devices is $l_G = 516\,\text{symbols}$ or $d_G = \frac{l_G}{62.5\,\text{ksymbols/s}} = 8.3\,\text{ms}$, which satisfies our requirements.

Beacon tracking disabled – The worst case for this option is depicted in Figure 2. The device allocated with the last GTS in the superframe generates a new alarm message and wakes up to listen for a beacon. If this device wakes up right at the time when after the first bit of an ongoing beacon transmission, it has to wait an extra beacon interval for the next beacon to arrive and to synchronize with. Upon reception of the beacon, the device has to delay the transmission until the arrival of its own GTS. Thus, in this worst case, the generated message has to wait approximately two beacon intervals before it can be sent to the PAN coordinator. Therefore, the guaranteed latency in this option can be estimated as the transmission time for $l_G = 2 \times l_{BI} + l_{init}$, where l_{init} depicts the initial startup delay after the node wakes up. This delay is a hardware specific constant, e.g. for the Chipcon 2420 chip $d_{init} = 0.97\,\text{ms}$ [8]. Thus, the maximum guaranteed latency for 20 devices and $d_{init} = 0.97\,\text{ms}$ equals to $d_G = \frac{2 \times l_{BI}}{62.5\,\text{ksymbols/s}} + d_{init} = 16.78\,\text{ms}$. Although this value exceeds the required latency of 10 ms, it can be deployed in the applications with weaker requirements on latency and stress more energy-efficiency.

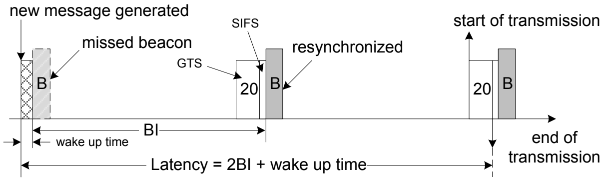

Fig. 2. Worst case for beacon tracking disabled

VI. CONCLUSION

We proposed an improved version of the IEEE 802.15.4 MAC layer that keeps the original PHY layer for best hardware compatibility with existing devices. The improvements include a modified superframe structure supporting only GTS allocations and a new data frame format. This solution allows the network working in either beacon-tracking enabled or disabled mode, which result in different energy consumption levels. The results of the analysis clearly show that the required guaranteed latency bounds can be satisfied for the selected case study when the beacon tracking is enabled in the network. The results also influenced the proposed standard IEEE 802.15.4e [7].

REFERENCES

[1] "Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low Rate Wireless Personal Area Networks (WPANs)," IEEE, IEEE Sandard 802.15.4-2006, 2006.

[2] P. Baronti, P. Pillai, V. W. Chook, S. Chessa, A. Gotta, and Y. F. Hu, "Wireless Sensor Networks: a Survey on the State of the Art and the 802.15.4 and ZigBee Standards," *Elsevier Computer Communications*, vol. 30, no. 7, pp. 1655–1695, May 2007.

[3] A. Willig, "Recent and Emerging Topics in Wireless Industrial Communications: A Selection," *IEEE Transactions on Industrial Informatics*, vol. 4, no. 2, pp. 102–124, May 2008.

[4] A. N. Kim, F. Hekland, S. Petersen, and P. Doyle, "When HART goes wireless: Understanding and implementing the WirelessHART standard." in *13th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA 2008)*. Hamburg, Germany: IEEE, September 2008, pp. 899–907.

[5] F. Chen, N. Wang, R. German, and F. Dressler, "Performance Evaluation of IEEE 802.15.4 LR-WPAN for Industrial Applications," in *5th IEEE/IFIP Conference on Wireless On demand Network Systems and Services (WONS 2008)*. Garmisch-Partenkirchen, Germany: IEEE, January 2008, pp. 89–96.

[6] M. Kunz, "Wireless LAN planning is a science, not an art!" *The Industrial Ethernet Book*, vol. 34, September 2006. [Online]. Available: http://ethernet.industrial-networking.com/articles/articledisplay.asp?id=1353

[7] "Wireless Personal Area Networks: Proposal for Factory Automation," IEEE, IEEE Proposed Standard 802.15.4-15/08/0571r0, August 2008.

[8] M. Kohvakka, M. Kuorilehto, M. Hännikäinen, and T. D. Hämäläinen, "Performance Analysis of IEEE 802.15.4 and ZigBee for Large-Scale Wireless Sensor Network Applications," in *3rd ACM International Workshop on Performance Evaluation of Wireless Ad Hoc, Sensor, and Ubiquitous Networks (ACM PE-WASUN 2006)*, Terromolinos, Spain, 2006, pp. 48–57.