Proc. of the 17th IEEE Conf. on Emerging Technologies and Factory Automation (ETFA 2012), Cracow, Poland, September 17-21, 2012.

WBuST: A Real-Time Energy-Aware MAC Layer Protocol for Wireless Embedded Systems

Gianluca Franchino and Giorgio Buttazzo
{*g.franchino, g.buttazzo*}*@sssup.it*
Scuola Superiore Sant'Anna Pisa, Italy

Abstract—**The design of wireless embedded systems for real-time applications requires a careful management of timing and energy requirements. This paper describes a wireless communication protocol that can guarantee both message deadlines and system lifetime by properly allocating the network bandwidth to each node. The protocol allows multi-hop wireless communication under different network topologies. The proposed approach is validated through both theoretical and experimental results.**

I. INTRODUCTION

During the last decade, the interest on networked wireless systems has experienced an exponential growth, mainly for their potential use in a wide range of applications, including target tracking in military fields, health monitoring, domotics, intelligent buildings, and so on.

The delay introduced in the network has a significant impact on the system performance, which can be specified according to different Quality of Service (QoS) levels. For example, time-critical data related to alarms must be delivered within stringent deadlines, and control loops data have to be transmitted periodically with a bounded delay variation (jitter).

When considering the design of a communication stack for real-time systems, using a deterministic Medium Access Control (MAC) layer is crucial to guarantee a bounded transmission delay for any packet sent throughout the network. The techniques adopted to handle the channel access can be roughly divided in three categories: contention based, scheduling based, and hybrid approaches. The former makes use of CSMA/CA or ALOHA [1] methods, the second one implements scheduling algorithms to rule the channel access and the latter is a combination of both. Each approach has its own advantages and drawbacks. CSMA/CA is simple, robust, highly scalable and does not need clock synchronization between nodes. The downside is that it suffers access collisions where two or more nodes can access the channel at the same time, causing a delay in the message transmission. Moreover, since carrier sensing does not work for nodes more than one hop away, a handshake mechanism [1] is necessary to mitigate the hidden/exposed terminal problem [2]. As a consequence of both collisions and handshake, the network throughput can be greatly reduced. On the other hand, scheduling based methods do not suffer hidden/exposed node problems, are collision-free and highly predictable in terms of transmission delay. The main shortcoming is that, in many cases, they need some form of clock synchronization between nodes that increases the protocol overhead; the network scalability is more difficult to achieve, and much more infrastructure support is needed with respect to CSMA/CA.

In battery-operated systems, the energy management represents another key issue to be addressed at design time. Radio devices available in the market have different operating modes, each characterized by a different level of power consumption. The most common are: sleep, receiving, and transmitting. In this work, the possibility offered by the sleep mode is exploited to the reduce energy consumption.

A. Contributions and summary

This paper describes the Wireless Budget Sharing Token (WBuST) protocol, which is a MAC layer protocol designed for real-time wireless networks of embedded devices. WBuST can handle real-time and best effort traffic in multi-hop networks, while saving energy to guarantee a desired lifetime. The channel access is handled by a mixed approach that adopts a bandwidth-reservation mechanism to guarantee the desired performance and a contention-based mechanism for transmitting control and management messages.

Network devices are grouped into clusters of adjacent nodes, with a different radio channel assigned to each cluster. In this way, the transmissions within adjacent clusters can take place at the same time without interfering with each other. Clusters can be connected to form various network topologies, with each cluster managed by a coordinator, which is the node with the best link quality to neighbor nodes.

The most relevant contribution of this work is the theoretic analysis of the protocol performance, which provides a powerful method for guaranteeing a given QoS level and a desired lifetime for a given amount of network traffic. This is particularly useful to implement admission control mechanisms to handle overload conditions. Concerning power management issues, besides of minimizing the energy consumption, as done in most of the related works reported in Section II, this work also provides a method for selecting the protocol parameters to guarantee a given network lifetime. The properties of WBuST are also validated by experimental results.

The rest of the paper is organized as follows. Section II analyzes the related works, Section III describes the proposed protocol in detail, Section IV introduces the traffic model, the bandwidth allocation schemes and the analysis of the protocol performance. Section VI reports the experimental results and, finally, Section VII states the conclusions.

II. RELATED WORK

Real-time communication and energy saving issues over wireless networks have received great consideration in the literature during the last years. However, not many authors addressed both problems simultaneously.

Caccamo et al. [3] proposed a cellular network architecture with a MAC protocol based on the Earliest Deadline First (EDF) algorithm [4]. Implicit prioritization is achieved by exploiting the periodic nature of the traffic in sensor networks. The implementation of this scheme requires clock synchronization among nodes contending for a channel. Crenshaw et al. [5] presented an improved version called *Robust Implicit*-EDF (RI-EDF) protocol, which does not require clock synchronization, providing bandwidth reclamation, energy saving techniques and robustness in the presence of certain classes of node failures. Sobral and Becker [6] proposed a Hybrid Contention/TDMA-based MAC protocol for ad-hoc wireless networks organized into clusters. The proposed protocol can

978-1-4673-4737-2/12/$31.00 ©2012 IEEE

guarantee timely bounded communications both inside and outside the clusters, operating without a central coordinator. Prabh and Abdelzaher [7] considered hexagonal meshes networks and proposed a transmission scheduling algorithm that guarantees a real-time communication. Bui et al. [8] introduced a prioritized MAC layer protocol that provides soft real-time communication in multi-hop wireless networks.

Unlike the previous approaches, which are mainly focused on the MAC layer, the SPEED protocol proposed by He et al. [9] is designed for real-time communication in sensor networks and defines the rules for all layers of the communication stack. The message deadlines are guaranteed by a probabilistic guarantee method.

The RT-Link protocol, proposed by Rowe et al. [10], is a time-synchronized link layer protocol that guarantees a predictable lifetime and a bounded end-to-end delay across multiple hops. The authors provided an analytical estimation of the maximum energy consumption, such that it is possible to derive the minimum network lifetime. Koubaa et al. [11] analyzed the power efficiency and the timeliness of the IEEE 802.15.4 standard [12] when the GTSs mechanism is used. Koubaa et al. [13] provided a methodology based on Network Calculus to estimate the end-to-end delay bounds, buffering and bandwidth requirements in IEEE 802.15.4 cluster-tree networks. Toscano and Lo Bello [14] presented an algorithm for superframe scheduling in industrial sensor networks based on the IEEE 802.15.4 standard. This algorithm is able to avoid beacon collisions by scheduling cluster superframes over multiple radio channels. Saifullah et al [15] analyzed the problem of real-time transmission scheduling in WirelessHART networks. Later, the same authors extended their work with an analysis of the end-to-end delay for fixed-priority scheduling [16].

III. THE WBUST PROTOCOL

This section describes the proposed protocol in detail. WBuST is a MAC layer protocol that can operate both in single-hop and in multi-hop networks, serving different kinds of communication flows.

A. Network model

Network nodes are grouped into n clusters, each denoted by C_i. A node can be of three different types:

- Cluster node. It is a node that may exchange information with other nodes within and outside the cluster.
- Coordinator node. It is a node located in the central area of the cluster in charge of synchronizing and scheduling the cluster nodes to access the wireless medium. Depending on the context, C_i is also used to denote the node coordinator of cluster i.
- Router node. It is a node located in the central area of the cluster in charge of managing the communication with other router nodes. It is denoted by R_i.

An example of a network is illustrated in Figure 1. Note that both the coordinator and the router are also cluster nodes. In the rest of the paper, we assume that each cluster contains a set of cluster nodes and a coordinator that also operates as a router. Moreover, nodes are connected either by a mesh or star topology, depending on the application requirements. In a mesh topology, any node can communicate with any other node within its communication range, and all nodes are connected to the coordinator. In a star topology, all nodes can only communicate with the coordinator, meaning that any communication between two nodes must pass thorough the coordinator. For instance, in the network illustrated in Figure 1 the nodes of cluster C_1 are connected using a star topology, whereas the other clusters adopt a mesh topology.

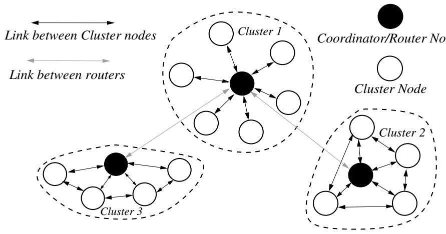


Figure 1. Example of network structure.

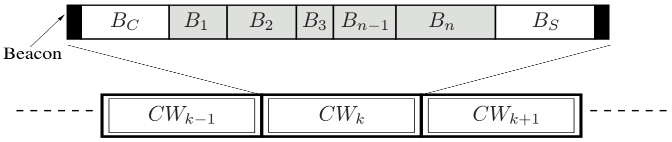


Figure 2. Intra-cluster Communication Structure.

More complex topologies can be implemented at the network level by connecting different clusters through their coordinator/router nodes. For example, the three clusters shown in Figure 1 are connected according to a tree structure. The support offered by WBuST for multi-hop networks is described in Section III-C.

The network formation problem is not taken into account in this work, since this problem can be solved using standard techniques available in the literature, see for instance [17].

B. Intra-cluster communication

The communication among cluster nodes occurs by sharing a periodic Communication Window (*CW*), whose structure is illustrated in Figure 2. Each *CW* is delimited by a coordination packet, namely the *beacon*, periodically sent by the coordinator. The beacon is used to define the *CW* length, synchronize the nodes and communicate the *CW* schedule.

Each *CW* is divided into slots, whose duration is referred to as *time budget*. Some slots have a specific usage. In particular:

- B_C is the contention slot. It immediately follows the beacon and it is used by cluster nodes to send requests to the coordinator for joining the cluster, reserving a slot, or exchanging control information with the coordinator.
- B_i, with $i = 1, ..., n$, is the slot reserved for node i by the coordinator, so node i can transmit its messages accessing the channel without contention. Its dimension depends on traffic parameters.
- B_S is the last slot in the *CW*, used by the all nodes to enter in sleep mode to save energy.

A slot can be used by a node to transmit both real-time and non real-time traffic. The rules for allocating and managing slots are described in Section IV. Note that, since each node transmits during a different slot in the *CW*, transmissions are collision-free, except within the B_C slot, where nodes willing to communicate with the coordinator contend for accessing the channel using the CSMA/CA scheme available in the IEEE 802.15.4 standard [12].

C. Inter-cluster communication

Inter-cluster communication is handled by router nodes. Although a single router per cluster is assumed in the following, protocol rules are still valid in the case of multiple routers per cluster.

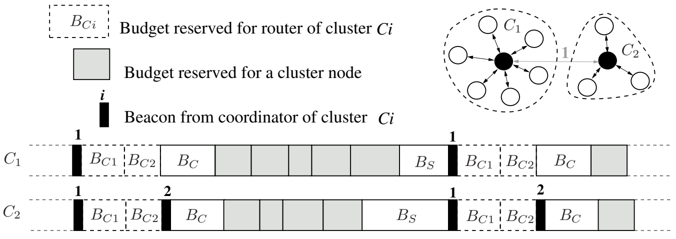


Figure 3. Example of inter-cluster communication.

1) Cluster-chain topology: The inter-cluster communication is introduced through a simple example consisting of a network composed by two clusters, C_1 and C_2, as shown in Figure 3. In each cluster C_i, both coordinating and routing functions are carried out by the same node, denoted by R_i.

To achieve a reliable and efficient inter-cluster communication, the following rules must be observed:

- The link between two clusters must be synchronized by the beacon transmitted by one of the cluster coordinators, defined at design time to act as a master. In the example shown in Figure 3, the inter-cluster synchronization is obtained through the beacon sent by cluster C_1.
- To guarantee a correct inter-cluster communication, both clusters must use the same beacon period, T_b.
- To allow simultaneous communications within different clusters preventing interference, each cluster must use a different radio channel. To use low-cost radio devices, each router is assumed to be equipped with a half-duplex transceiver, which can use one frequency at a time and cannot receive and transmit simultaneously.
- The two routers communicate on the channel allocated to the master coordinator (R_1 in the considered example).
- Each router can transmit real-time and non real-time traffic using a budget B_{C_i} assigned at design time.
- Both router budgets must be allocated in the *CWs* of both clusters.

The communication between clusters proceeds as follows:

1) At the beginning of each *CW* in C_1, R_2 listens to the channel allocated to R_1 to receive the beacon from it.
2) Once the beacon is received, both routers are synchronized and the inter-cluster communication can take place within the slots B_{C1} and B_{C2} reserved to them in each *CW*. For the master coordinator, the budgets for inter-cluster communication are placed at the beginning of its *CW*, right after the beacon, while for the other router they are placed at the end of the window.

At the end of budget B_{C2}, R_2 switches to the channel assigned to its cluster and sends its beacon to synchronize the intra-cluster communication, which starts with the contention budget B_C (see Figure 3). Instead, in cluster C_1, the intra-cluster communication starts immediately after B_{C2}, because the beacon in C_1 has already been sent at the beginning of the *CW*. Note that each *CW* of C_2 includes the transmission of two beacons on two different channels: one from R_1 and one from R_2. Instead, the *CW* of master includes only a beacon transmission. It follows that the bandwidth lost due to the protocol overhead (due to beacon transmissions) is greater in C_2 than in C_1.

The two-cluster topology can easily be extended to n clusters connected as a chain. As before, the link between two adjacent clusters has to be managed by one of the cluster coordinators. For instance, the link between two adjacent clusters could be managed by the coordinator with the smallest index.

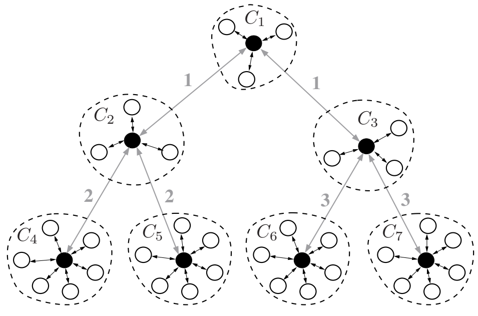


Figure 4. Cluster-tree topology example.

2) Cluster-tree topology: The second topology considered in this work is the cluster-tree structure, an example of which is shown in Figure 4, including seven clusters connected as a binary tree. To guarantee a correct inter-cluster communication, the links between a parent node and its leaves is synchronized by the coordinator of the parent node. For example, the links C_1-C_2 and C_1-C_3 are both coordinated by C_1. In the figure, the number on each link identifies the link coordinator.

The inter-cluster communication scheduling starts from the root of the tree, going downward to the tree leaves. The schedule of the inter-cluster communication is represented in Figure 5, for the left branch, and in Figure 6 for the right branch. At the beginning of each *CW* of cluster C_1, R_1 sends its beacon and both R_2 and R_3 are listening to its channel. Once the beacon is received, C_1, C_2 and C_3 are synchronized and can start transmitting their inter-cluster traffic within slots B_{C1}, B_{C2} and B_{C3}, in the corresponding *CW*. After transmitting its messages in B_{C2}, R_2 switches on its cluster's channel and transmits the beacon to coordinate the inter-cluster communication with C_4 and C_5, as well as its intra-cluster communication. Note that, at the beginning of each *CW* in C_2, R_4 and R_5 are listening to the channel of C_2 to get the corresponding beacon. Once this is received, R_2, R_4, and R_5 can communicate within slots B_{C2}, B_{C4} and B_{C5}. After the inter-cluster slots, R_4 and R_5 transmit their local beacons to coordinate the intra-cluster communication.

The inter-cluster coordination in the right branch of the tree, shown in Figure 6, is performed in a similar way. Finally, it is worth reminding that the intra-cluster communication requires the beacon period to be the same for every cluster.

IV. BUDGET ALLOCATION AND PROTOCOL PROPERTIES

This section analyzes the timing properties of the protocol in order to perform real-time guarantee tests on message deadlines. In particular, worst-case transmission times are derived for a number of bandwidth allocation schemes.

Referring to the *CW* structure shown in Figure 2 and 3, it can be noted that cluster nodes access the channel, one by one, in a circular fashion, and the access time of node i is limited by the slot budget B_i. In other words, the channel access is regulated by a weighted round robin policy, where each budget B_i is proportional to (weighted with) the length of stream S_i. Since each node uses its budget to transmit both real-time and best-effort traffic, it is not difficult to see that the WBuST scheduling policy is equivalent to that of the BuST [18] protocol (designed for wired networks), hence the results obtained for BuST can be exploited for the analysis of WBuST.

BuST is based on a token-passing scheduler where network nodes form a logical ring by exchanging a control packet, the token, in a circular fashion, and only the node holding the token can access the channel. Once a node gets the token, it

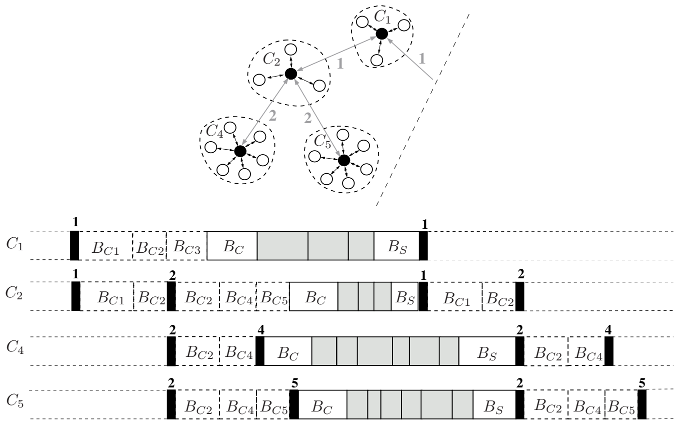


Figure 5. Communication schedule for the left branch of the cluster-tree.

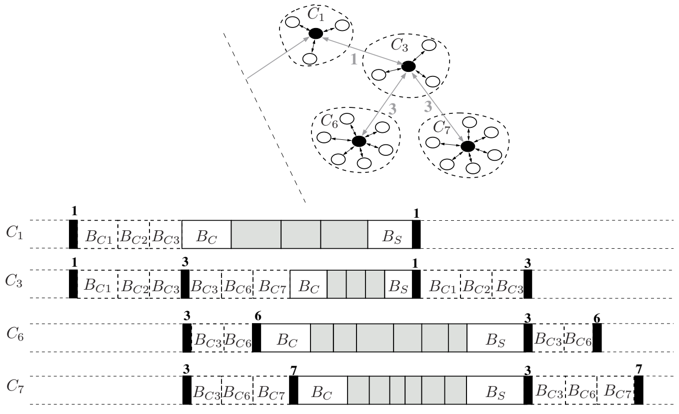


Figure 6. Communication schedule for the right branch of the cluster-tree.

can transmit its traffic, real-time and best-effort, for a time no greater than its time budget. Then, the token is passed to the next node along the logical ring. If a node has not traffic to deliver or it finishes its transmission without consuming the entire budget, the token is passed to the next node. In this way, the unused bandwidth is implicitly reused by the other nodes, so reducing the time interval between two consecutive channel accesses by the same node. In WBuST, a similar behavior is obtained by the transmission of the beacon, which can be seen as a token that synchronizes the nodes and defines the structure of the *CWs*. With respect to BuST, however, there are a couple of differences: first, by using a single beacon per *CW*, the protocol overhead is reduced; second, if a node does not use its budget, the following nodes cannot advance their transmission. Although WBuST does not allow the nodes to advance their channel access, as done in BuST, a bandwidth reclaiming method that emulates the token passing mechanism will be described in Section IV-C. Another difference between the two protocols is that BuST cannot be used in multi-ring networks, while BuST also provides multi-hop communication.

BuST and WBuST is that, the latter, supports multi-hop communication instead the former does not support multi-ring communication.

A. Traffic Model

WBuST manages three types of traffic: real-time sporadic, bandwidth-guaranteed, and best-effort traffic.

The sporadic traffic of a node i is modeled by a sporadic message stream S_i^{RT}, characterized by three parameters:

- the maximum length M_i, measured in time units, of the messages generated by the stream;
- the relative deadline D_i associated to each message of the stream;
- the minimum inter-arrival time T_i (equivalent to the period in case of periodic traffic) between the generation of two consecutive messages in the stream.

The time unit in the system is equal to the time needed to send a packet, hence all stream and protocol parameters are expressed in number of packets. The ratio $U_i^{RT} = M_i/T_i$ denotes the bandwidth of the stream S_i^{RT}. The bandwidth-guaranteed traffic of a node i is described by a message stream S_i^{BG}, defined by a single parameter U_i^{BG}, denoting the bandwidth required by the stream. Finally, the best effort-traffic is generated by non real-time messages without specific timing requirements.

Without loss of generality, we assume that each node i generates a single message stream, S_i^{RT} or S_i^{BG}. For the sake of simplicity, whenever the stream type is not relevant or is implicit, we simply refer to it as S_i. Furthermore, a node can also generate best-effort traffic without considering any particular traffic model. Thus, the traffic generated in each cluster is defined by a set of n streams $\Gamma = \{S_1, S_2, ..., S_n\}$, where each stream can be a real-time or a bandwidth-guaranteed stream. The total channel bandwidth U required by Γ is defined by

$$U = \sum_{i=1}^{n} U_i \tag{1}$$

where $U_i = U_i^{RT}$ or $U_i = U_i^{BG}$.

To cope with the unreliability of wireless channels, each node uses a Forward Error Correction mechanism to encode each message. A node can use different code lengths, depending on the channel status, and the maximum code redundancy is included in the maximum message length M_i or in the required bandwidth U_i^{BG}.

In addition to the time budgets B_C, B_is, and B_S, the following protocol parameters are defined:

- T_b is the beacon period which defines the dimension of each *CW*.
- the Target Beacon Time (T_{BT}) is the greatest value for T_b that guarantees the correct operation of WBuST.
- τ is the protocol overhead, that is, the time in each *CW* that cannot be used by nodes to transmit their messages. It is given by the time needed to transmit the beacon plus other components, such as the the timed needed to switch between radio channels and inter-frame spacings (*IFSs*) required between consecutive packet transmissions to leave a receiving node the time to process a packet before receiving the next one.
- $\alpha = \tau/T_b$ is the bandwidth lost due to the protocol overhead.

To guarantee the correct operation of the protocol, T_{BT} has to be not greater than the minimum relative deadline $D_{min} = \min_i(D_i)$. This is a necessary and sufficient condition to guarantee at least one packet transmission for each node i, between the time t_i^r a new message in S_i is produced for transmission and its absolute deadline $d_i = t_i^r + D_i$. Figure 7 shows the maximum delay between the time t_3^r a new message is ready in stream S_3 and the end of the budget B_3 in the next *CW*. Such a delay is equal to $T_b \le T_{BT}$ and has to be no greater than D_3. Supposing $D_{min} = D_3$ and $T_b = T_{BT}$, the necessary and sufficient condition that guarantees at least one packet transmission becomes: $t_3^r + T_b \le t_3^r + D_3$. Note

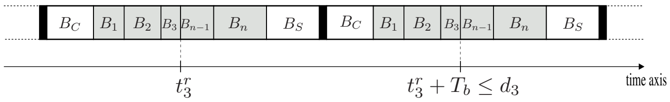


Figure 7. Example showing the constraint on T_{BT}.

that a message in S_i experiences the worst-case transmission delay when it becomes ready just after the end of the budget assigned to node i.

To guarantee a correct operation of the protocol, any selection of communication parameters must satisfy the following constraints.

Definition IV.1 ***(Bandwidth Constraint)*** *For each network cluster, the total channel bandwidth allocated to nodes must not exceed the available bandwidth:*

$$\sum_{i=1}^{n} \frac{B_i}{T_{BT}} \leq 1 - \alpha \tag{2}$$

where T_{BT} is the maximum beacon transmission period, that is, the maximum dimension of the CW.

The *Bandwidth Constraint* is necessary to guarantee a stable operation of the protocol.

Definition IV.2 ***(Deadline Constraint)*** *For a stream S_i, let WC_i be the maximum time interval between the generation of a message and the time at which its transmission is completed, namely the worst-case transmission time. Then, the Deadline Constraint requires that for any i:*

$$WC_i \leq D_i \tag{3}$$

where D_i is the relative deadline of stream S_i.

The *Deadline Constraint* is necessary to guarantee that all messages are sent by their deadline. Note that, while D_i is imposed by the application, WC_i depends on the protocol parameters, such as the budget B_i and the dimension of the contention window (i.e., the beacon period T_b).

In the rest of the paper, only streams with $D_i = T_i$ are taken into account. In the case of streams with $D_i < T_i$, the results can be extended by replacing T_i with D_i. The case of $D_i > T_i$ is not treated and it will be part of future work. Moreover, due to the lack of space, the proofs of Lemmas IV.3 and IV.4 are not reported. Such proofs can be found in [19].

To guarantee a correct transmission of both real-time and bandwidth-guaranteed traffic, time budgets have to be properly dimensioned as shown by the following lemma, which provides the worst-case transmission time WC_i for any message generated by stream S_i.

Lemma IV.3 *Under the WBuST protocol, if $T_i \geq T_{BT}$ and the network traffic is generated by real-time streams, it holds that, for $i = 1, \ldots, n$,*

$$WC_i = \left\lceil \frac{M_i}{B_i} \right\rceil (T_{BT} - B_i) + M_i. \tag{4}$$

If the network traffic includes both real-time and best-effort streams, it holds that, for $i = 1, \ldots, n$,

$$WC_i = \left\lceil \frac{M_i}{B_i} \right\rceil T_{BT}. \tag{5}$$

From the previous lemma and the *Deadline Constraint*, it is clear that the guarantee of message deadlines depends on the budgets reserved to the nodes. Such an issue is discussed in the following section.

Budget Alloc. Scheme	Assignment rule	U^*
PA	$B_i = U_i(T_{BT} - \tau)$	$\frac{1-3\alpha}{2(1-\alpha)}$
NPA	$B_i = \frac{U_i}{U}(T_{BT} - \tau)$	$\left\lfloor \frac{\beta_{min}}{\beta_{min}+1} \right\rfloor (1-\alpha)$
MLA	$B_i = \frac{M_i}{\lfloor \beta_i \rfloor}$	$\left\lfloor \frac{\beta_{min}}{\beta_{min}+1} \right\rfloor (1-\alpha)$

Table I
BUDGET ALLOCATION SCHEMES

B. Budget Allocation Schemes

The *Deadline Constraint* and the *Protocol Constraint* can be satisfied by a proper budget allocation to each node. Several schemes have been proposed in the literature for timed token protocols [20], which can also be used in this context. In particular, this work will focus on the analysis of Proportional Allocation (PA), Normalized Proportional Allocation (NPA) and Modified Local Allocation (MLA) schemes. Such schemes are listed in Table I together with their assignment rule, where $\beta_i = T_i/T_{BT}$.

The performance of each BAS (Buget Allocation Scheme) has been extensively analyzed for timed token protocols [20] and BuST [21], [22]. Mainly, the metric adopted to compare the allocation schemes is the Worst Case Achievable Utilization (WCAU), which is the maximum channel bandwidth U^* such that, for any stream set having total channel utilization $U \leq U^*$, the scheme can guarantee that all message deadlines will be met.

The third column of Table I shows the WCAU of each budget allocation scheme considered in this work. Note that, since the BuST and the WBuST protocols implement the same scheduling policy, the U^* derived for BuST in each scheme is still valid for WBuST, hence the formulas shown in the table are taken from the literature [22]. Note that all schemes but PA have the same WCAU, which depends on $\beta_{min} = \min_i(\beta_i) = \min_i(T_i/T_{BT})$. In particular, given the minimum stream period T_{min}, the lower T_{BT} the greater U^*. It follows that, to guarantee more bandwidth for real-time streams, it is necessary to keep T_{BT} as small as possible. Conversely, when decreasing T_{BT}, the protocol overhead, and consequently α, increases; hence, the value of T_{BT} has to be carefully chosen.

Concerning best-effort traffic, BuST [18], [21] and consequently WBuST can guarantee a minimum bandwidth for non real-time traffic as long as the bandwidth U required by real-time streams is less than $1 - \alpha$, so avoiding the risk of starvation.

Since each *CW* also contains the contention slot (B_C) and sleep slot (B_S), to verify the message schedulability through the WCAU, it is necessary either to add these slots to the overhead τ, or to create two message streams with dummy parameters, namely S_C and S_S, and then assigning B_C and B_S with the same BAS used for the other streams.

All allocation schemes shown in Table I work for intra-cluster communication (single-hop networks), whereas for inter-cluster communication only the NPA scheme can be adopted. The reason is that, to guarantee the inter-cluster synchronization as described in Section III-C, the *CWs* of all clusters must have the same dimension, that is, the beacon period must be the same for all clusters. It means that, given T_{BT}, for a multi-hop network formed by n clusters C_j with $j \in \{1, ..., n\}$ it must be:

$$T_b = \tau + B_C + \sum_i B_i + B_S = T_{BT}. \tag{6}$$

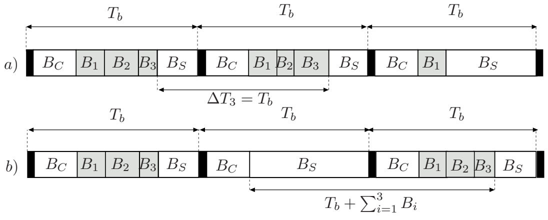


Figure 8. Example of bandwidth reclaiming mechanism.

To guarantee this requirement, it is necessary to select $T_{BT} \leq min_{i,j}(D_i^j)$, where D_i^j is the relative deadline of stream S_i in cluster C_j, and to use an allocation scheme that satisfies Equation 6. The only scheme that satisfied such a requirement is NPA. For the other schemes, if $U < 1$ then $T_b < T_{BT}$.

C. Bandwidth Reclaiming

When a node does not use its reserved slot completely, the left budget can be reclaimed to increase the transmission time of the other nodes. Under WBuST, this situation can occur every time a node scheduled to access the channel has not messages to deliver or has less traffic to transmit than expected.

To overcome this problem, a bandwidth reclaiming mechanism can be implemented as follows. When node i saves some budget, the unused budget is added to that of node $i+1$, and so on, until the unused budget by all nodes is added to the sleep budget. In this way, since the sum of node budgets is constant and equal to $\sum_{i=1}^{n} B_i$, the dimension of each *CW* is constant and equal to T_b.

When the reclaiming mechanism is used, the bandwidth available for each node i is:

$$BW_i = \frac{B_i + B_{i-1}^l}{T_b} \tag{7}$$

where B_{i-1}^l is the amount of the budget saved by node $i-1$ and by all previous nodes. Note that, the bandwidth saved in a *CW* can only be reclaimed in the same *CW*. In the worst-case, all the unused bandwidth is reclaimed in the sleep slot, and hence it is entirely used to save energy.

Figure 8a shows an example in which in the second *CW* the budget left by node 2 is used by node 3. In the third *CW*, nodes 2 and 3 do not transmit, so their budgets are added to B_S. Figure 8b shows another example in which all the nodes have not traffic to deliver in the second *CW*, so that all the budgets are added to B_S. Observe that the beacon period, hence the dimension of each *CW*, is constant and equal to T_b.

To implement the bandwidth reclaiming mechanism, each node can add the transmission length to every packet header, such that the following node can derive the amount of budget unused by the preceding one. In practice, each node i starts transmitting as soon as node $i-1$ finishes its transmission, and continues to transmit until the end of slot B_i. In this case, the transmission time of node i will be $B_i + B_{i-1}^l$, where $B_{i-1}^l > 0$ if node $i-1$ finished before the beginning of B_i in the current *CW*, otherwise $B_{i-1}^l = 0$. In case a node has no traffic to transmit, it sends a short packet with transmission length equal to 0 to the following node.

The following lemma provides the worst-case transmission time for any message from stream S_i, when the reclaiming mechanism is implemented. The result provided below also holds in the case the nodes have both real-time and best-effort traffic to deliver.

Lemma IV.4 *Under the WBuST protocol with the bandwidth reclaiming mechanism, for* $i = (1, \ldots, n)$*, if* $T_i \geq T_{BT} + \sum_{j=1}^{i} B_j$*, then*

$$WC_i = \left\lceil \frac{M_i}{B_i} \right\rceil (T_{BT} - B_i) + M_i + \sum_{j=1}^{i} B_j. \tag{8}$$

From the lemma above it follows that, for any node i, the worst-case interval between two consecutive channel accesses depends on the position of B_i within the *CWs*: the smaller the node index (i), the shorter the transmission delay of node i. This property can be taken into account when selecting the node indexes. In general, the index assignment should be based on the message deadline: the shorter the deadline the smaller the node index.

V. ENERGY SAVING MECHANISM

As already mentioned, a sleep slot is allocated at the end of each *CW* to allow cluster nodes to turn off their radio transceiver. This section describes how to calculate the dimension of this slot to guarantee a desired lifetime for each network cluster.

To calculate the average energy consumption of a node, observe that, in each *CW*, a node i transmits for a time no greater than B_i, it is in receiving mode for a time no greater than $T_b - B_i - B_S$, and in sleep mode for B_S time units. If P^{tx} is the power dissipated by a node in transmission mode, P^{rx} in receiving mode, and P^{sl} in sleep mode, then the average energy wasted by node i after t units of time is

$$E_i(t) = \left[P^{tx} B_i + P^{rx}(T_b - B_i - B_S) + P^{sl} B_S\right] \frac{t}{T_b}. \tag{9}$$

Note that, since the time needed to join a cluster is usually negligible with respect to the time a node operates in the cluster, the energy wasted during the joining phase is not considered in the equation above. Moreover, the coordinator node is assumed to be mains powered, thus, its energy consumption is not a concern.

In the following, we show how to guarantee a minimum lifetime L_j^m for each cluster C_j by properly dimensioning the sleep budget. The cluster lifetime is defined as the time instant at which k nodes of the cluster run out of energy. The value of k depends on the application. For instance, if cluster C_j includes a set of $n = 10$ sensor nodes that sample the same physical quantity, e.g. the temperature, and the application requires that for each sampling period at least $r = 3$ samples (from r different nodes) are needed for an accurate measure of the temperature, then the cluster lifetime L_j can be defined as the time at which $k = n - r + 1 = 8$ nodes exhaust their energy.

Given a desired lifetime L_j^m for each cluster C_j, Equation 9 allows calculating the minimum sleep slot that can guarantee L_j^m. For the sake of simplicity, it is assumed that at the system startup all nodes have the same amount of available energy E^0.

To guarantee that in cluster C_j all nodes will operate at least for L_j^m time units, it is sufficient to impose that for any i, $E_i(L_j^m) \leq E^0$, that is:

$$\left[P^{tx} B_i + P^{rx}(T_b - B_i - B_S) + P^{sl} B_S\right] \frac{L_j^m}{T_b} \leq E^0. \tag{10}$$

Rearranging the terms, it is possible to derive the sleep budget B_S as a function of B_i:

$$\forall i \; B_S(B_i) \geq \frac{\left(P^{tx} - P^{rx}\right) B_i + \left(P^{rx} - \frac{E^0}{L_j^m}\right) T_b}{P^{rx} - P^{sl}}. \tag{11}$$

Considering $(P^{tx} - P^{rx}) \geq 0$, if $k = 1$, since $B_S(B_i)$ is a straight line growing with B_i, given the greatest node budget $B_{max} = max_i(B_i)$, to guarantee the desired lifetime is sufficient to select B_S as follows:

$$B_S \geq \frac{(P^{tx} - P^{rx})\, B_{max} + \left(P^{rx} - \frac{E^0}{L_j^m}\right) T_b}{P^{rx} - P^{sl}}. \tag{12}$$

If $k = 2$, B_S is computed considering the second greatest node budget; if $k = 3$, it is necessary to consider the third one, and so on. Instead, if $(P^{tx} - P^{rx}) < 0$, since $B_S(B_i)$ is a straight line decreasing with B_i, the k-th smallest budget should be considered to compute B_S.

After computing the sleep budget, the next step is to verify the stream set feasibility. Considering B_S as a budget assigned to a dummy stream $S_S(M_S, D_S, T_S)$, where parameters M_S, $D_S = T_S$, and $U_S = M_S/T_S$ depend on the allocation scheme, it is possible to assess the message schedulability by the methods shown in Section IV-B. In particular, to exploit the worst-case achievable utilization tests, for both the PA and the NPA schemes it is necessary to derive U_S such that, if $U + U_S \leq U^*$, then all message deadlines will be met.

From the assignment rules of Table I, it is possible to derive the parameters of stream S_S for the PA and NPA schemes. To derive U_S with the PA scheme it sufficient to impose $B_S = U_S(T_{BT} - \tau)$, that is:

$$U_S = \frac{B_S}{T_{BT} - \tau}. \tag{13}$$

For NPA, $B_S = \frac{U_{Sl}(T_{BT}-\tau)}{U+U_{Sl}}$, thus, it turns out:

$$U_S = \frac{U B_S}{T_{BT} - \tau - B_S}. \tag{14}$$

For the MLA scheme, the Deadline Constraint (Inequality 3) is always met for any stream set with $U \leq 1$ [21]. Hence, it is not necessary to derive U_S, but to guarantee the stream set schedulability it is sufficient to verify that the Bandwidth Constraint (Inequality 2) holds, that is:

$$\sum_{i=1}^{n} \frac{B_i}{T_{BT}} + \frac{B_S}{T_{BT}} \leq 1 - \alpha. \tag{15}$$

Finally, if message deadlines cannot be met for a given value of B_S, it is possible to adopt an elastic approach [23], where the stream utilization U_i is not fixed, but can range in an interval $[U_i^{min}, U_i^{max}]$, varying the slot B_i in the range $[B_i^{min}, B_i^{max}]$, selected such that both the message deadlines and lifetime are met. The development of this is idea is part of future work.

VI. EXPERIMENTAL RESULTS

The effectiveness of WBuST has been tested through experimental evaluations. The testbed is based on 7 FLEX boards [24], equipped with a 16 bits microcontroller and a IEEE 802.15.4 compliant transceiver. The firmware has been written in C under the real-time kernel ERIKA Enterprise [24].

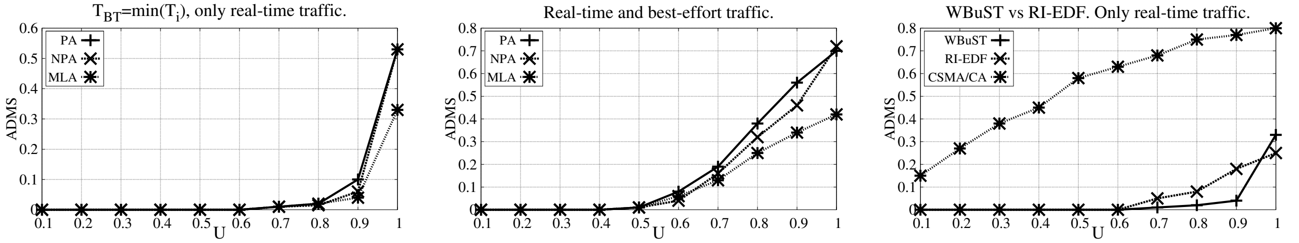
All experiments consider a cluster of 6 nodes plus the coordinator. Two message streams are assigned to each node, to a total amount of 12 real-time streams. The metric used to assess the protocol performance is the Average Deadline Miss Ratio (ADMS), which is the average ratio of messages that do not respect their deadline to the total number of messages generated by all network streams. ADMS is measured by varying the total channel utilization U of real-time traffic, as defined by Equation 1, from 0.1 to 1 with a step of 0.1. A total amount of 20 experiments have been carried out for each value of U. Furthermore, in each test, the deadline miss ratio is computed monitoring the network for 10 minutes. In each experiment, the stream parameters are generated as follows. First, the stream utilizations are randomly generated within a uniform distribution, after that, for each value U_i, a relative deadline D_i is randomly selected in the interval $[300\ ms; 900\ ms]$ with a step of $5\ ms$. Message length M_i is computed multiplying U_i by D_i. Note that, in this evaluation all stream periods are considered equal to deadlines. Finally, node budgets are assigned by the allocation schemes of Table I.

Since for these experiments the network is static, that is, the number of node/streams is fixed during the tests, the control budget B_C is not necessary and so is not allocated in the *CWs*. In the same way, as long as the energy consumption is not taken into account, the sleep budget is not considered as well. The rest of this section presents and analyzes the main experimental results.

Figure 9(a) shows the ADMS of the budget allocation schemes, when nodes transmit and receive only real-time traffic. Note that, in this first test T_{BT} is set equal to $min(D_i)$. The results show that, as long as U is not greater than 0.6, under all schemes every message is delivered within its deadline. With $U \geq 0.7$, the number of messages that do not meet the deadline starts increasing. However, for $U \leq 0.9$ the ADMS is still less than 0.1, i.e. more than the 90% of the messages are respecting the deadline. In particular, the best performing scheme, MLA, presents an average deadline miss ratio not greater than 0.05. The ADMS is quite high for $U = 1$, when the cluster channel is overloaded, because in this first test the available bandwidth is $1 - \alpha = 0.9$. Notice that, since the nodes are transmitting only real-time traffic, the results obtained in this experiment are better than those predicted by the WCAU of each scheme, as shown in Table I.

Figure 9(b) reports the results of the second set of experiments, where $T_{BT} = min(D_i)$, and the nodes also transmit best-effort traffic. In particular, it is assumed that a node has an infinite amount of best-effort traffic to deliver. It means that, a node always uses its entire budget. In other words, for any value of U the channel is fully loaded, that is, the sum of U and the best-effort traffic utilization is always equal to $1 - \alpha$. The results reported in the figure shows that, for all schemes, the ADMS is null as long as $U < 0.5$, after that it increases. The MLA scheme presents the lowest ADMS for all values of U. Using the formulas provided by Table I, with $T_{BT} = min(D_i)$ and $1 - \alpha = 0.9$, it turns out that the WCAU for all schemes is slightly lower than 0.5, hence, the experimental results are consistent with the theory.

WBuST was also compared with the RI-EDF real-time protocol and a CSMA/CA approach as defined by the un-slotted IEEE 802.15.4 standard. In this test, the node budgets are allocated through the MLA scheme, $T_{BT} = min(D_i)$ and the nodes transmit only real-time traffic. The results are shown in Figure 9(c). As reported in the graph, both WBuST and RI-EDF do not present any deadline miss as long as $U \leq 0.6$; for greater U, both protocols show a non-null ADMS. In particular, WBuST performs better than RI-EDF for $0.7 \leq U < 1$. RI-EDF performs better than WBuST only when the channel is overloaded, i.e. $U = 1$. Notice that, although based on EDF, RI-EDF can guarantee a channel bandwidth not greater than 0.6 for real-time streams, due to the protocol overhead which is mainly given by the packet recovery mechanism. Since the CSMA/CA approach is not designed to support real-time



(a) Average deadline miss ratio with only real-time fraffic.

(b) Average deadline miss ratio with real-time and best-effort traffic.

(c) Comparison with RI-EDF and CSMA/CA (IEEE 802.15.4)

Figure 9. Experimental results.

communications, its ADMS is always non-null and increases rapidly with U.

Finally, a set of experiments was carried out to test the ability of WBuST on saving energy by means of the sleep budget mechanism. Due to lack of space, these experiments are not reported. An interested reader can find them in [19].

VII. CONCLUSIONS AND FUTURE WORK

This paper presented WBuST, a MAC layer protocol for time sensitive communication in wireless embedded systems. The proposed protocol support both real-time and best-effort traffic in multi-hop networks, grouping the network devices into clusters managed by a coordinator node. A different radio channel is assigned to each cluster, where the nodes are synchronized by the coordinator through the transmission of a periodic beacon. The channel access is regulated by a budget reservation mechanism that guarantees a predictable transmission time, both for intra-cluster and inter-cluster communications.

The protocol performance is assessed by a theoretical analysis, that provides the tools to verify whether a given amount of real-time traffic, described by a stream set, can be guaranteed by the protocol. Moreover, an energy saving mechanism is provided to reduce the energy consumption and guarantee a desired lifetime. The experimental evaluation showed the ability of WBuST in managing real-time traffic and the consistency between theoretical and experimental results. Moreover, the comparison with the RI-EDF protocol showed that WBuST outperform this last in terms of deadline miss ratio.

The future work concerns the analysis of the maximum end-to-end transmission delay for cluster-tree networks. The goal is to derive a method that guarantees end-to-end message deadlines, by properly selecting the dimension of the time budgets assigned for inter-cluster communication.

REFERENCES

[1] T. S. Rapport, *Wireless Communication, Principles and Practice.* Prentice-Hall, 1996.

[2] F. A. Tobagi and L. Kleinrock, “Packet switching in radio channels: Part II - the hidden terminal problem in carrier sense multiple-access modes and the busy-tone solution,” *IEEE Transactions on Communication*, vol. 23, no. 12, pp. 1417–1433, December 1975.

[3] M. Caccamo, L. Y. Zhang, L. Sha, and G. Buttazzo, “An Implicit Prioritized Access Protocol for Wireless Sensor Networks,” in *Proc. of the IEEE Real-Time Systems Symposium (RTSS 2002)*, Dec. 2002.

[4] C. L. Liu and J. W. Layland, “Scheduling Algorithms for Multiprogramming in Hard Real-time environment,” *Journal of the ACM*, vol. 20, no. 1, pp. 40–61, 1973.

[5] T. L. Crenshaw, A. Tirumala, S. Hoke, and M. Caccamo, “A Robust Implicit Access Protocol for Real-Time Wireless Collaboration,” in *Proc. of the 17th IEEE Euromicro Conf. on Real-Time Systems (ECRTS 2005)*, Jul. 2005.

[6] M. M. Sobral and L. B. Becker, “A wireless hybrid contention/tdma based mac for real-time mobile application.” in *Proc. of the ACM Symp. on Applied Computing (SAC 2008)*, Fortaleza, Brazil, March. 2008.

[7] T. F. A. K. Shashi Prabh, “On Scheduling and Real-Time Capacity of Hexagonal Wireless Sensor Networks,” in *Proc. of the 19th IEEE Euromicro Conference on Real-Time Systems (ECRTS 2007)*, Jul. 2007.

[8] B. D. Bui, R. Pellizzoni, M. Caccamo, C. F. Cheah, and A. Tzakis, “Soft Real-Time Chains for Multi-Hop Wireless Ad-Hoc Networks,” in *Proc. of the 13th IEEE Real Time and Embedded Technology and Applications Symposium (RTAS’07)*, Apr. 2007.

[9] T. He, J. A. Stankovic, C. Lu, and T. F. Abdelzaher, “SPEED: A stateless protocol for real-time communication in sensor networks,” in *Proceedings of the International Conference on Distributed Computing Systems (ICDCS 2003)*, Providence, RI, USA, May 2003.

[10] A. Rowe, R. Mangharam, and R. Rajkumar, “RT-Link: A Time-Synchronized Link Protocol for Energy-Constrained Multi-hop Wireless Networks,” in *Third IEEE Int. Conf. on Sensors, Mesh and Ad Hoc Communications and Networks (IEEE SECON06)*, Dec. 2006.

[11] A. Koubaa, M. Alves, and E. Tovar, “Energy/Delay Trade-off of the GTS Allocation Mechanism in IEEE 802.15.4 for Wireless Sensor Network,” *Inter. Journal of Comm. Systems*, vol. 20, no. 7, pp. 791–808, July 2007.

[12] IEEE 802.15.4 Std-2006, *Part 15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (LR-WPANs)*, 2006.

[13] A. Koubaa, M. Alves, and E. Tovar, “Modeling and worst-case dimensioning of cluster-tree wireless sensor networks,” in *Proc. of the 27th IEEE Real-Time Systems Symposium, (RTSS 2006)*, Dec. 2006.

[14] E. Toscano and L. Lo Bello, “Multiplechannel Superframe Scheduling for IEEE 802.15.4 Industrial Wireless Sensor Networks,” *IEEE Trans. on Industiral Informatics.*, vol. 8, no. 2, May 2012.

[15] A. Saifullah, Y. Xu, C. Lu, and Y. Chen, “Real-time Scheduling for WirelessHART Networks,” in *Proc. of the 31st IEEE Real-Time Systems Symposium (RTSS 2010)*, Dec. 2010.

[16] ——, “End-to-End Delay Analysis for Fixed Priority Scheduling in WirelessHART Networks,” in *IEEE Real-Time and Embedded Technology and Applications Symposium (RTAS 11)*, Apr. 2011.

[17] P. Kumarawadu, D. Dechene, M. Luccini, and A. Sauer, “Algorithms for node clustering in wireless sensor networks: A survey,” in *Proc. of ICIAFS 2008*, Dec 2008.

[18] G. Franchino, G. C. Buttazzo, and T. Facchinetti, “BuST: Budget Sharing Protocol for Hard Real-Time Communication,” in *Proc. of the 12th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA 2007)*, Sept. 2007.

[19] G. Franchino and G. C. Buttazzo, “A MAC layer protocol for Wireless Communication in Embedded Systems.” TeCIP, Scuola Superiore Sant’Anna, Pisa, Italy., http://www.franchino-gianluca.it/SSSA-ReTiS-Franchino-TR01-12.pdf, Tech. Rep., 2012.

[20] S. Zhang, A. Burns, J. Chen, and E. Lee, “Hard Real-Time Communication with the Timed Token Protocol: Current State and Challenging Problems,” *Real-Time Systems*, vol. 27, no. 3, pp. 271–295, Sept. 2004.

[21] G. Franchino, G. C. Buttazzo, and T. Facchinetti, “Properties of BuST and Timed Token Protocols in Managing Hard Real-Time Traffic,” in *Proc. of the 13th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA 2008)*, Sett. 2008.

[22] ——, *Token Passing Techniques for Hard Real-Time Communication*. IN-TECH, 2010.

[23] G. Franchino, G. C. Buttazzo, and M. Marinoni, “An Energy-Aware Algorithm for TDMA MAC Protocols in Real-Time Wireless Networks,” in *Proc. of the 5th IEEE Symposium on Industrial Embedded Systems (SIES2010)*, July 2010.

[24] “RTOS Erika Enterprise,” http://erika.tuxfamily.org.