Wake-up Radio based Approach to Low-Power and Low-Latency Communication in the Internet of Things

Rajeev Piyare

Advisor  Dr. Amy L. Murphy, Bruno Kessler Foundation, Italy
Committee  Prof. Stefano Basagni, Northeastern University, USA
           Prof. Olivier Berder, Université de Rennes 1, France
           Prof. Renato Lo Cigno, University of Trento, Italy

Trento, Italy, 2019
Abstract

For the Internet of Things to flourish a long lasting energy supply for remotely deployed large-scale sensor networks is of paramount importance. An uninterrupted power supply is required by these nodes to carry out tasks such as sensing, data processing, and data communication. Of these, radio communication remains the primary battery consuming activity in wireless systems. Advances in MAC protocols have enabled significant lifetime improvements by putting the main transceiver in sleep mode for extended periods. However, the sensor nodes still waste energy due to two main issues. First, the nodes periodically wake-up to sample the channel even when there is no data for it to receive, leading to idle listening cost. On the other side, the sending node must repeatedly transmit packets until the receiver wakes up and acknowledges receipt, leading to energy wastage due to over-transmission. In systems with the low data rate, idle listening and over-transmission can begin to dominate energy costs.

In this thesis, we take a novel hardware approach to eliminate energy overhead in WSNs by addition of a second, extremely low-power wake-up radio component. This approach leverages an always-on wake-up receiver to delegate the task of listening to the channel for a trigger and then waking up a higher power transceiver when required. With this on-demand approach, energy constrained devices are able to drastically reduce power consumption without sacrificing the application requirements in terms of reliability and network latency.

As a first major contribution, we survey a large body of work to identify the benefits and limitations of the current wake-up radio hardware technology. We also present a new taxonomy for categorizing the wake-up radios and the respective protocols, further highlighting the main issues and challenges that must be addressed while designing systems based on wake-up radios. Our survey forms a guideline for assisting application and system designers to make appropriate choices while utilizing this new technology.

Secondly, this thesis proposes a first-ever benchmarking framework to enable accurate and repeatable profiling of wake-up radios. Specifically, we outline a set of specifications to follow when benchmarking wake-up radio-based systems, leading to more consistent and therefore comparable evaluations whether in simulation or testbed for current and future systems.

To quantitatively assess whether wake-up technology can provide energy savings superior to duty cycled MACs, reliable tools are required to accurately model the wake-up radio hardware and its performance in combination with the upper layers of the stack. As our third contribution, we provide an open-source simulator, WaCo for development and evaluation of wake-up radio protocols across all layers of the software stack. Using our tool together with a newly proposed wake-up radio MAC layer, we provide an exhaustive evaluation of the
wake-up radio system for periodic data collection applications. Our evaluations highlight that wake-up technology is indeed effective in extending the network lifetime by shrinking the overall energy consumption.

To close the gap between the simulation and the real world experiments, we adopt a cutting edge wake-up radio hardware and build a Wake-up Lab, a modular dual-radio prototype. Using our Wake-up Lab, we thoroughly evaluate the performance of the wake-up radio solution in a realistic office environment. Our in-depth system-wide evaluation reveals that wake-up radio-based systems can achieve significant improvements over traditional duty cycling MACs by eliminating periodic receive checks and reducing unnecessary main radio transmissions while maintaining end-to-end latency on the order of tens of milliseconds in a multi-hop network.

As a step toward sustainable wireless sensing, this thesis presents a proof of concept system where an extremely low-power switch coupled with a wake-up receiver is continuously powered by a plant microbial fuel cell (PMFC) and a new receiver-initiated MAC-level communication protocol for on-demand data collection. MFC converts the chemical energy into electricity by exploiting the metabolism of bacteria found in the sediment, thus offering a promising power source for autonomous sensing system. However, sources such as PMFCs are severely limited in the quantity of energy they can generate, unable to directly power the sensor nodes. Therefore, we consider radical hardware solutions in combination with the communication stacks to reduce this power gap. Thanks, to the hardware-software co-design proposed above, we were able to reduce the overall power consumption to a point where an extremely low-power PMFC source can sustain the sensor node’s operation with a data sampling rate of over 30 seconds.

Finally, we propose to enhance the LoRa based low-power wide area networks by fusing wake-up receivers and long-range wireless technologies. The current LoRaWAN architecture is mainly designed and optimized for up-links where the remote end devices disseminate data to the gateway using pure ALOHA techniques. As such, this limits the ability of the gateway to control, reconfigure, or query the specific end devices, crucial for many Internet of Things applications. To shift the communication modality from push to pull based, we propose a new network architecture that leverages wake-up receiver and a receiver-initiated On-demand TDMA MAC. The former allows the gateway to trigger the remote device when there is data to be collected else keep the device in sleep mode, while the latter allows retrieving data efficiently from the nodes without congesting the network. Our testbed experiments reveal that the proposed system significantly improves energy efficiency by offering network reliability of 100% with end devices dissipating only a few microwatts of power during periods of inactivity. By moving away from the realm of pure ALOHA communication to wake-up receivers, we were able to exploit the low power modes of the sensor node more effectively.

Through these contributions, this thesis pushes forward the applicability of ultra-low power wake-up radios, by quantitatively measuring the trade-offs, energy efficiency, reliability, and latency. Further, by demonstrating superior performance via proof of concepts, this thesis provides a stepping stone towards the goal of achieving energy-neutral, yet responsive
communication systems using wake-up radio technology.

Keywords: wake-up radio, wake-up receiver, wireless sensor and actuator networks, on-demand communication, MAC protocols, Internet of Things, cyber-physical systems, energy harvesting, LPWAN, COOJA, ContikiOS, LoRa, LoRaWAN
List of Publications

The contributions of this thesis have been published in several peer-reviewed international conferences and journals.

Journals


International Conferences


Publications not part of this thesis

# Contents

Abstract i

List of Publications v

List of figures xiii

List of tables xvii

Introduction 1

I Overviewing WUR 11

1 Ultra Low Power Wake-Up Radios: A Hardware and Networking Survey 13

1.1 Wake-up Radio Defining Characteristics and Requirements . . . . . . . . . . . . 14

1.2 Architecture and Taxonomy of WuRs . . . . . . . . . . . . . . . . . . . . . . . . . 17

1.2.1 Generic Architecture of WuRs . . . . . . . . . . . . . . . . . . . . . . . . . 17

1.2.2 Taxonomy Overview . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 19

1.3 State-of-the-Art Wake-up Radios . . . . . . . . . . . . . . . . . . . . . . . . . . . 23

1.3.1 Active Wake-up Radios . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 23

1.3.2 Passive Wake-up Radios . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 32

1.3.3 Semi-active Wake-up Radios . . . . . . . . . . . . . . . . . . . . . . . . . . 34

1.3.4 Non-RF Based WuRs . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 37

1.3.5 Summary . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 39

1.4 Statistical Analysis . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 43

1.4.1 Modulation Schemes . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 43

1.4.2 Sensitivity vs. Power Consumption . . . . . . . . . . . . . . . . . . . . . . . 44

1.4.3 Data Rate vs. Power Consumption . . . . . . . . . . . . . . . . . . . . . . . 45

1.4.4 Range and Frequency Usage . . . . . . . . . . . . . . . . . . . . . . . . . . . 46

1.4.5 Summary . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 48

1.5 Medium Access Control . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 48

1.5.1 Classification of WuR-based Medium Access . . . . . . . . . . . . . . . . . . . 49

1.5.2 Bi-directional MAC Protocols . . . . . . . . . . . . . . . . . . . . . . . . . . 53

1.5.3 Receiver-Initiated MAC Protocols . . . . . . . . . . . . . . . . . . . . . . . . 56
## Contents

1.5.4 Transmitter-Initiated MAC Protocols ........................................ 57  
1.5.5 Summary ................................................................. 59  
1.6 Routing Protocols utilizing WuRs ............................................... 59  
1.6.1 Routing-Only Protocols .................................................. 60  
1.6.2 Cross-Layer Protocols ................................................... 62  
1.6.3 Summary ................................................................. 64  
1.7 Key Application Areas ........................................................ 65  
1.7.1 Wireless Body Area Network (WBAN) .................................... 65  
1.7.2 Smart City ............................................................... 67  
1.7.3 Smart Metering ........................................................... 68  
1.7.4 Wildlife Monitoring ....................................................... 68  
1.7.5 Security and Surveillance Systems ....................................... 70  
1.7.6 Indoor Localization ....................................................... 71  
1.7.7 Asset Tracking .......................................................... 72  
1.7.8 Wearables ................................................................. 73  
1.7.9 Smart Grid ............................................................... 74  
1.7.10 Summary ................................................................. 74  
1.8 Open Issues, Challenges and Future Research Directions ......... 75  
1.8.1 Hardware Design ......................................................... 75  
1.9 Conclusions ................................................................. 79  

II Assessing WUR: Simulation and Testbed ........................................ 81  

2 WURBench: Toward Benchmarking Wake-up Radio-based Systems .... 83  
2.1 Wake-up Radio Hardware Micro-benchmarking ............................ 84  
2.1.1 What to measure? ....................................................... 85  
2.1.2 Recommended practice ................................................ 85  
2.2 Benchmarking System as a Whole .......................................... 86  
2.2.1 Application settings .................................................... 87  
2.2.2 Metrics ................................................................. 87  
2.2.3 Evaluation mechanisms ................................................. 87  
2.3 Summary ................................................................. 90  

3 WaCo: A Wake-Up Radio COOJA Extension for Simulating Ultra Low Power Radios .................................................. 91  
3.1 Background ................................................................. 92  
3.1.1 WSN Simulation Environment: COOJA ............................ 93  
3.2 WaCo: Enabling Support of WuR in Simulation ....................... 93  
3.2.1 COOJA Platform Extensions ....................................... 94  
3.2.2 Contiki OS extensions ............................................... 95  
3.3 Related Work ............................................................ 96  
3.4 Summary ................................................................. 97
4 Exploiting WaCo: A Data Collection Case Study 99

4.1 W-MAC in a Nutshell .............................................. 100

4.2 WaCo Evaluation Settings ........................................... 102

4.2.1 Network Topology .............................................. 103

4.2.2 Routing Protocols .............................................. 103

4.2.3 MAC Protocols .............................................. 104

4.2.4 Simulation Parameters and Metrics .................................. 104

4.2.5 WuR Hardware Parameters ...................................... 106

4.3 WaCo for Network Performance Analysis ................................. 106

4.3.1 Contiki Collect with Wake-up Radio .................................. 106

4.3.2 RPL with Wake-up Radio ...................................... 111

4.3.3 Addressable vs. Broadcast Mode ................................. 113

4.4 Conclusions ......................................................... 114

5 Wake-up Lab: Implementing a Wake-up Radio Testbed 115

5.1 Hardware Integration: Wake-up radio + Tmote Sky ...................... 116

5.1.1 Wake-up Radio module ...................................... 116

5.1.2 Main Sensor Node: Tmote Sky .................................. 117

5.1.3 Enabling communication between Tmote and Wake-up Radio .......... 118

5.2 Software Design: Wake-up Radio Driver .................................. 119

5.3 Wake-up Lab for Application development ............................... 120

5.4 Summary .......................................................... 121

6 Exploiting Wake-up Lab 123

6.1 Revisiting W-MAC .................................................. 124

6.2 Evaluation Methodology ............................................ 125

6.2.1 Experimental Configurations ..................................... 126

6.2.2 Protocols ..................................................... 127

6.2.3 In-software Power Measurements .................................. 127

6.2.4 Performance Metrics ........................................... 128

6.3 Micro-benchmarks .................................................. 128

6.3.1 Multiple Senders: Unicast Communication .............................. 128

6.4 Testbed Multi-hop Experiments ..................................... 133

6.5 Robustness to External Interference ................................... 135

6.6 Energy Micro-benchmarks .......................................... 137

6.7 Conclusions ........................................................ 138

III Exemplifying WUR: Energy Neutral and LPWAN 141

7 Plug into a Plant: Plant Microbial Fuel Cell and Wake-Up Radio in Synergy Toward Energy Neutral Sensing 143

7.1 Background ......................................................... 145
<table>
<thead>
<tr>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.1.1 Plant Microbial Fuel Cell ........................................ 145</td>
</tr>
<tr>
<td>7.2 System Architecture ..................................................... 146</td>
</tr>
<tr>
<td>7.2.1 Energy Harvester ....................................................... 146</td>
</tr>
<tr>
<td>7.2.2 On-demand Switch ..................................................... 147</td>
</tr>
<tr>
<td>7.2.3 Sensor Node ............................................................ 148</td>
</tr>
<tr>
<td>7.3 Energy Requirements and Delivery ........................................ 148</td>
</tr>
<tr>
<td>7.3.1 Baseline Power Consumption .......................................... 149</td>
</tr>
<tr>
<td>7.3.2 Active State Power Consumption ...................................... 149</td>
</tr>
<tr>
<td>7.3.3 Energy Delivered by the PMFCs ....................................... 150</td>
</tr>
<tr>
<td>7.3.4 Energy Stored by the Capacitor ...................................... 150</td>
</tr>
<tr>
<td>7.4 Energy Efficient Communication for Sustainable Sensing ................. 151</td>
</tr>
<tr>
<td>7.5 Experimental Setup and Evaluation ....................................... 153</td>
</tr>
<tr>
<td>7.5.1 Output Power Characterization of PMFC ................................. 154</td>
</tr>
<tr>
<td>7.5.2 Observations from Controlled in-lab Experiments ....................... 155</td>
</tr>
<tr>
<td>7.5.3 Scalability Analysis using WaCo Simulator ............................ 157</td>
</tr>
<tr>
<td>7.6 Conclusion .................................................................. 158</td>
</tr>
<tr>
<td>8 On-Demand LoRa: Asynchronous TDMA for Energy Efficient and Low Latency Communication in IoT ........................................ 159</td>
</tr>
<tr>
<td>8.0.1 Challenges ................................................................ 160</td>
</tr>
<tr>
<td>8.0.2 Approach .................................................................. 161</td>
</tr>
<tr>
<td>8.1 Background .................................................................. 162</td>
</tr>
<tr>
<td>8.1.1 Low-Power Long-Range Communication: LoRa ......................... 162</td>
</tr>
<tr>
<td>8.1.2 Wake-Up Receivers ....................................................... 163</td>
</tr>
<tr>
<td>8.2 Related Work .................................................................. 164</td>
</tr>
<tr>
<td>8.2.1 Long-Range Communication Schemes ..................................... 164</td>
</tr>
<tr>
<td>8.2.2 Wake-Up Radio-Enabled MAC ............................................ 165</td>
</tr>
<tr>
<td>8.3 Energy Efficient Data Communication Network and Protocol ............ 167</td>
</tr>
<tr>
<td>8.3.1 Network Architecture ................................................... 167</td>
</tr>
<tr>
<td>8.3.2 On-demand TDMA MAC Design ......................................... 169</td>
</tr>
<tr>
<td>8.4 Evaluation Methodology ..................................................... 172</td>
</tr>
<tr>
<td>8.4.1 Testbed Setup ............................................................. 172</td>
</tr>
<tr>
<td>8.4.2 Radio Settings ............................................................ 173</td>
</tr>
<tr>
<td>8.4.3 Ambient Noise Floor ..................................................... 174</td>
</tr>
<tr>
<td>8.4.4 Evaluation Metrics ....................................................... 175</td>
</tr>
<tr>
<td>8.5 Network Performance Analysis .............................................. 175</td>
</tr>
<tr>
<td>8.5.1 Packet Delivery Ratio .................................................... 176</td>
</tr>
<tr>
<td>8.5.2 Latency .................................................................. 176</td>
</tr>
<tr>
<td>8.5.3 Energy Efficiency ........................................................ 178</td>
</tr>
<tr>
<td>8.6 Conclusions .................................................................. 180</td>
</tr>
<tr>
<td>9 Conclusions and Outlook ....................................................... 181</td>
</tr>
</tbody>
</table>
List of Figures

1 (a) Overall Wake-up Radio architecture. The blue region indicates the traditional node integrated with the Wake-up Radio. (b) Remote triggering using wake-up radio scheme. ........................................... 3

1.1 Expanded view of the generic wake-up receiver architecture with energy harvesting capabilities. ................................................................. 17
1.2 Typical wake-up signal packet structure. ........................................... 19
1.3 Taxonomy of wake-up radios showing the hardware design space. ........... 20
1.4 Defining characteristics of RF-based WuRs with various building blocks. Wake-up radios meeting each characteristic appear in Tables 1.2 to 1.7 while Tables 1.8 and 1.9 provide the full summary of each surveyed prototype. ................. 23
1.5 Discrete components based WuRx architecture [21]. ........................... 24
1.6 Generic Block diagram of Heterodyne WuRx [43]. ............................. 25
1.7 Wake-up receiver employing an ultra-low power MCU for address decoding and interrupt generation. .................................................. 29
1.8 Node address comparison using "matched filter" correlation detector. ...... 30
1.9 RFID-based Wake-up receiver prototype [83]. .................................... 34
1.10 Wake-up receiver prototype utilizing Austria Micro Systems AS393x WuRx [15]. 36
1.11 Sensitivity of low power RF based wake-up receivers vs. Power consumption w.r.t different signal modulation techniques. ......................... 44
1.12 Data Rate of low power RF based wake-up receivers vs. their Power consumption w.r.t different signal modulation techniques. ..................... 46
1.13 Frequency selection vs. Power consumption. .................................... 47
1.14 Communication Range of RF based wake-up receivers vs. their Power consumption w.r.t different signal modulation techniques. ................. 48
1.15 Taxonomy of wake-up radio based MAC protocols. ............................ 49
1.16 Various wake-up radio communication schemes. ............................... 52
1.17 Taxonomy of wake-up radio based routing protocols. ........................... 60

2.1 Benchmarking different MAC protocols w.r.t wake-up radio system in simulation using WaCo. ................................................................. 88

3.1 The wake-up radio module extensions of WaCo to COOJA and ContikiOS. ....... 94
3.2 COOJA node power tracker and timeline plugin extensions. .................... 96
# List of Figures

4.1 W-MAC unicast operation. Communication is initiated using the WuR while data is exchanged via main radio. .............................................. 101

4.2 Grid topology of 100 nodes with sink node 1 in the center. .................. 103

4.3 MAC protocols compared against W-MAC in WaCo. .......................... 104

4.4 Contiki Collect evaluation data collected with WaCo. ............................ 108

4.5 Per-node main radio duty cycle of ContikiMAC and W-MAC when routing using Contiki Collect. .................................................. 109

4.6 Scalability analysis of WuR based network. ........................................ 110

4.7 Effect of noise floor on network performance. ..................................... 111

4.8 RPL evaluation collected with WaCo. ................................................ 112

4.9 Comparison of addressing and broadcast mode of the wake-up radio. ....... 113

5.1 Schematic diagram of the adopted always-on wake-up radio module. The ULP microcontroller is used for managing the wake-up radio stack. .......... 117

5.2 Software architecture of dual-radio network stack. The Tmote Sky runs the ContikiOS that controls the CC2420 and the wake-up radio module via the SPI. The two MCU cores execute the two separate radio stacks. ............... 118

5.3 Tmote Sky integrated with the wake-up radio module. .......................... 119

5.4 SoleLab testbed at FBK. ................................................................. 121

6.1 W-MAC: Transmitter-initiated wake-up radio MAC. ............................. 125

6.2 Network topology used for the unicast experiment. ............................... 129

6.3 Comparing main radio duty cycle of ContikiMAC vs. W-MAC using the testbed. .......................... 129

6.4 Baseline power distribution comparison for the main radio using ContikiMAC and W-MAC in the testbed. ........................................ 130

6.5 Single-hop network performance of W-MAC in simulation and the testbed over varying number of senders. ...................................... 131

6.6 Network topology used for the broadcast experiment. ........................... 132

6.7 Broadcast primitive performance of W-MAC in the testbed and simulation. 132

6.8 Testbed network topology for conducting multi-hop data collection. ......... 134

6.9 W-MAC’s performance for data collection over 3-hop network with varying Inter-packet intervals in the testbed. ............................... 134

6.10 Impact of destructive interference on the wake-up address decoding. ....... 135

6.11 Interference test set-up using the testbed. ........................................ 136

6.12 Wake-up radio performance under heavy interference. ........................ 136

6.13 Power microbenchmarking for the Wake-up Lab prototype. Figures show the nodes instantaneous current draw for four respective link primitives: wake-up beacon transmission and reception; data transmission and reception. .... 138

7.1 Laboratory setup of the plant microbial fuel cell powered wireless sensing system.145

7.2 Soil based plant microbial fuel cell. ................................................... 146
7.3 Overall architecture of the wake-up receiver triggered on-demand sensing unit. When a trigger is received, the switch closes to connect the sensing and transmitting node to the PMFC-based energy harvested power supply. Black lines indicate the power supply while blue indicates trigger signaling. PMFC is the only power source for the whole system. 147

7.4 Power consumption measurement of the remote sensor node with radio-trigger capability. In (a) the baseline consumption including the wake-up receiver in idle is approximately 1.8 $\mu W$, while in (b) the active power of the whole remote node when triggered (i.e. from trigger to transmission) is 3.6 $mJ$. Note the two axes have different ranges and units. 150

7.5 Long-term monitoring of PMFC and its output power in an indoor environment. 151

7.6 WRI-MAC in unicast mode using address-based wake-up beacons. A single, separate wake-up beacon (W) is sent to initiate communication with each respective transmitter. 152

7.7 WRI-MAC in broadcast mode: a single broadcast wake-up beacon (W) is sent to acquire data from all the network nodes. 154

7.8 Instantaneous voltage profile of the PMFC (lower trace) and the buffer capacitor (upper trace). At the beginning the remote node is triggered and the PMFC is left to recover towards an open circuit voltage. 155

7.9 Voltage profile of the PMFC (red) and the buffer capacitor (blue) during stress test. The activation of the node is also highlighted by a dashed black line every time it was triggered. 156

8.1 Class A end devices open the two receive windows (RX1 and RX2) after a transmission to receive an ACK or any other downlink traffic from the gateway. 160

8.2 Standard LoRa network topology. 163

8.3 Proposed heterogeneous IoT topology for low-power wide area sensor and actuator networks. 168

8.4 Receiver-initiated on-demand TDMA MAC overview. 170

8.5 Time synchronization between two end devices over the wake-up receiver. 171

8.6 Heterogeneous communication mote used for testbed evaluation. 173

8.7 Network topology and indoor office testbed deployment. 174

8.8 Measurement of noise floor in the office testbed environment. 175

8.9 Reliability and latency evaluation using different LoRa radio setting and protocols. 176

8.10 Energy evaluation of on-demand TDMA. 178
# List of Tables

1.1 Acronyms for wake-up radio technology. ........................................... 14
1.2 Wake-up radio categorization based on technology. .............................. 24
1.3 Wake-up radio categorization based on frequency usage. ....................... 26
1.4 Wake-up radio featuring address decoding. ....................................... 28
1.5 Various wake-up signal modulation techniques. ................................... 31
1.6 Technology utilized for passive wake-up radios. .................................. 32
1.7 Semi-active wake-up radio designs. .................................................. 34
1.8 RF based wake-up radio prototypes. ............................................... 41
1.9 Non-RF based wake-up radio prototypes. ......................................... 42
1.10 Wake-up radio based MAC protocol designs. ..................................... 54
1.11 Wake-up radio based routing protocols. ......................................... 62
1.12 Wake-up radio based application requirements. ................................... 66
1.13 Summary of issues, challenges, and opportunities for wake-up radio based hardware and software designs. .................................................. 76

4.1 Simulation Parameters ............................................................. 105
4.2 Power consumption of the CC2420 and WuR measured in-lab. Idle reflects listening to the channel, but not actively receiving. ................. 106
4.3 Summary of network-wide performance gain of W-MAC over ContikiMAC. W-MAC offers higher packet delivery ratio, lower energy consumption, and lower end-to-end latency. W-MAC performance meets or exceeds the widely used ContikiMAC link layer. .......................................................... 112

5.1 Wake-up radio driver APIs. ......................................................... 120

6.1 Configurations used throughout our indoor testbed experiments. ............. 127
6.2 Measured power consumption using WaCo and hardware. ....................... 137

7.1 Laboratory power and current measurement of the different components in various states. Idle reflects listening to the channel, but not actively receiving. . 152
7.2 Scalability analysis using different MAC operation modes. The table shows the power consumption and latency w.r.t network size. ....................... 158

8.1 Power measurement of the sensor mote in various states at 3.3 V. .............. 173
8.2 Three different LoRa radio settings (SET) (low, medium, and high data rate) used in our testbed experiments. ............................................................... 174
8.3 Scalability analysis using different on-demand TDMA operation modes w.r.t. to latency, power, and network size. .................................................. 179
**Introduction**

The Internet of Things (IoT) offers a new communication frontier considering networks between smart physical objects or "Things", which are embedded with sensors, actuators, and/or processing capabilities [1]. IoT opens the way for novel applications for various fields such as smart cities, building automation, domotics, logistics, smart grid, e-Health, and agriculture [2].

A founding pillar of the IoT concept is the availability of low-cost devices with low-power wireless communication capabilities, often deployed as part of a larger Wireless Sensor Network (WSN) [3], to provide both sensing and actuation capabilities. These devices are usually powered by batteries with restricted size and capacity [4], and thus have limited lifetime requiring careful power management. With the increase in the number of IoT devices, replacing or recharging batteries frequently will not only be costly but infeasible as well. Therefore, prolonging the lifetime of these devices, or even better achieving perpetual operation, becomes fundamental for the realization of the IoT vision.

Traditionally, these problems have been addressed by the introduction of low-power radios and of duty-cycling Medium Access Control (MAC) protocols [5]. Notwithstanding, one of the most power-hungry operations remains the wireless communication task. In most applications, its consumption far exceeds that of sensing, actuation, and processing, and becomes the main bottleneck in extending device lifetime.

**From Duty-cycling MACs to Wake-up Radios.** The main reason duty-cycling MACs alone cannot sufficiently extend the lifetime of a node is that the consumption of low-power wireless radios is almost the same when listening for transmissions as while transmitting. For example, the widely used CC2420 radio module consumes 21.8 mA in listening mode and 19.5 mA in data transmission mode [6]. If such a radio would be always-on (listening for other transmissions or transmitting) it would deplete reasonable sized batteries in less than a week.

During duty-cycling, the nodes are periodically put into sleep mode and are woken up only to transmit or to receive. Unfortunately, the so called duty-cycling ratio (the ratio of time the radio is on to time off) cannot alone extend the device lifetime, due to the following issues:

(i) *idle listening*: occurs when the node monitors the communication medium for ongoing transmissions, but there is no data to be received by the node. Since nodes must listen periodically to limit data latency, there is a listening power consumption that cannot be avoided, even in low data traffic scenarios.
(ii) **overhearing**: occurs when a node receives packets from its neighbors that are not intended for that node, leading to energy waste, especially when the network density is high and the data traffic is heavy.

Due to the sleep intervals, duty-cycling protocols also introduce non-negligible **data latency** since no information can be sent or received until the nodes wake-up, adversely impacting the responsiveness of the system.

Finally, duty-cycling MAC protocols must either maintain time synchrony to make sure transmitters send when receivers are awake, which induces a time synchronization overhead, or in the case of asynchronous operation the MAC protocol must employ continuous (or multiple) transmissions to ensure reception. The longer the sleep interval of the receiver, the longer the continuous transmission must be, dictating a lower-bound on achievable duty-cycles.

These design compromises have led the sensor network community to design and implement various MAC protocols resulting in a "MAC Alphabet Soup" for sensor networks [7] each targeting different scenarios and offering different compromises throughout the design space of energy consumption, latency, throughput, and fairness. Nevertheless, duty cycling protocols may not be suitable for delay sensitive and event-driven applications, and prolonging device lifetime requires extreme compromises in other dimensions of the design space, limiting the applicability of the technique.

The introduction of **wake-up radios** (WuR) aims to provide a novel hardware solution with listening power consumption orders of magnitude lower than that of low-power radios, promising results towards eliminating the aforementioned problems of idle listening, overhearing, continuous transmissions, and data latency.

In a wake-up radio architecture, as shown in Figure.1(a), an **ultra-low power**, secondary radio module with a receiver consuming a few micro watts of power is placed along side the primary, low-power radio. As the power dissipation of the wake-up receiver is very low w.r.t traditional low-power radio, one may keep the wake-up receiver always-on, resulting in a system that is responsive and energy-efficient. This always-on modality of the WuR is illustrated in Figure.1(b). In this setting, the main radio is kept in a deep sleep, or off mode, until it is needed. Instead when a node has a data packet to send, it sends a special packet/preamble known as a **wake-up signal** (WuS) using its **wake-up transmitter** (WuTx) to trigger the receiving node. On the receiver side, the always-on **wake-up receiver** (WuRx) detects this WuS, and generates an interrupt to the main node's micro-controller to switch it from sleep to an active mode. Subsequently, the micro-controller turns on the main radio transceiver to exchange data packets with the other node in a conventional manner.

**Wake-up Radio: Benefits and Design Trade-offs.** As identified previously, idle listening is a significant contributor to the overall energy consumption of duty cycling nodes. With the introduction of a WuRx with orders of magnitude lower consumption, the WuR approach minimizes this unnecessary energy wastage, as the main radio and the node will be activated only when there is an actual transmission.
In addition, some WuRs add circuitry for an addressing mechanism that can be used to solve the issue of overhearing by decoding an address embedded in the packet, waking up only a specific node rather than the entire neighborhood. Further, since the WuRx can be kept always-on, the node can operate in a purely asynchronous manner, activating the main radio on-demand, without requiring continuous transmissions. Finally, since the time taken to trigger the main node is on the order of milliseconds in contrast to protocols such as ContikiMAC [8] where the typical sleep interval is 125 ms, the latency problem faced by duty-cycling MACs is also reduced. Therefore, the wake-up radio based systems can sleep whenever possible, wake-up fast, and operate efficiently.

While the concept of the WuR seems simple and the benefits look promising, the hardware implementation and its usage as part of the larger system present several challenges and design trade-offs.

At the hardware design level, achieving continuous listening with very low power consumption places limits on signal processing and on the components that can be used in the WuRx. For instance, to measure wake-up signal strength (RSSI), receiver circuitry requires active components such as band-pass filters and amplifiers. However, addition of active components may negatively affect the overall listening consumption of the wake-up receiver. Strict bounds on power consumption also limits the choice of modulation schemes and receiver complexity, which, as a consequence, limits i) receiver sensitivity, ii) the achievable communication range, and iii) the receivers ability to overcome interference. As the main radio is triggered by the WuRx, this range limitation of the WuRx inherently limits the communication, regardless of the main radio’s capabilities. Various hardware options had been proposed in the literature exploring a wide range of options, including some that are not radio frequency (RF) based e.g., optical or acoustic.

As far as the MAC protocol is concerned, pure asynchronous operation enabled by the always-on WuRx largely simplifies protocol design. However, the development of new WuR specific MAC protocols are required, taking into account the dual radio setup of the WuR architecture.
Introduction

Thesis Outline and Contributions

The software-only approaches like radio duty cycling substantially reduce the power consumption of WSNs. However, nodes still dissipate power higher than tens of microwatts due to periodic receive checks while facing issues such as data latency and overhearing. This is a far cry from the point to which these nodes can be sustained autonomously and possibly achieve energy-neutral operation. This fact motives us to reduce energy wastage by adopting a radical hardware solution such as the wake-up radio and revising the network stack design. In this thesis, we focus on the always-on wake-up radio approach to circumvent the issues faced by the traditional approaches. Therefore, this thesis sets out to:

_demonstrate that the wake-up radio technology has the potential to revolutionize the battery-powered IoT devices over software-only techniques in alleviating the energy costs, leading the way for energy neutral systems._

We show this by studying, characterizing, providing tools, and designing novel protocols for the wake-up radio systems. The novel hardware and software techniques proposed in this thesis reduce power consumption to a point where nodes can be sustained for up to several years either powered by the batteries or energy harvesting sources. Further, the system-wide evaluations both carried out in simulations and testbeds concretely demonstrate that the wake-up radio is a key component toward achieving low-energy Internet of Things.

We next highlight the key contributions of our work, grouped in three parts: _i)_ overviewing, _ii)_ assessing, and _iii)_ exemplifying wake-up radios and appear in this thesis in the order described below.

**Part I: Overviewing WUR**

**Chapter 1: Ultra Low Power Wake-Up Radios: A Hardware and Networking Survey**

Over the last decade, a plethora of low-power communication protocols have been surveyed that focus mainly on the MAC/routing layer or on the general energy saving schemes for low-power networking. However, none of these surveys focus on the unique properties of the wake-up radio technology, its categorization, and the impact it has on the different layers of the networking stack. To fill this gap, in Chapter 1, we survey a large body of work (in total 180 research articles) to identify the research progress and to provide potential areas of improvement in wake-up radio hardware and networking software. Specifically, we

- survey seventy five radio-frequency (RF) and ten non-RF based wake-up radio prototypes implemented and tested since 2002.
- present a new taxonomy for categorizing the wake-up radios based on power source, addressing capability, channel usage, and communication medium.
- identify the key characteristics of the wake-up technology such as power consumption, sensitivity, and data rates and its impact on the networking layer.
- provide a statistical analysis as a guideline to choose wake-up prototypes for various applications regardless of their specific technology.
- survey and provide a new taxonomy to classify various wake-up radio based MAC and
routing protocols.

- highlight the main application areas that can benefit from this technology, mapping the suitable platforms and protocols studied above to these applications.
- identify the major issues and challenges that need to be addressed both on the hardware design side as well as the design of the upper layers of the networking stack to achieve efficient wake-up radio systems.

Our survey in Chapter 1 provides a holistic view of the wake-up radio technology and its characteristics forming a guideline that will assist application and system designers to make appropriate choices while utilizing this technology.

Part II: Assessing WUR: Simulation and Testbed

Chapter 2: WURBench: Toward Benchmarking Wake-up Radio-based Systems

Following the rigorous survey, this thesis next defines a set of specifications to follow when evaluating the performance of wake-up radio based systems, making experiments repeatable and results directly comparable whether in simulation or test-bed—in other words, a benchmark. Standard methodologies for benchmarking are crucial for quantitatively evaluating the performance of the wake-up radio technology. Unfortunately, currently, no accepted standard for such quantitative measurement exists. Further, there is no consensus in the WSN community on what objective evaluation procedures and metrics should be used to understand the performance of the whole system exploiting this technology. One of the primary challenges is that lack of standardization in the wake-up radio domain has prevented researchers from comparing results and leveraging previous work that could otherwise avoid duplication and speed up the validation process.

To offer progress in this direction, we present, in Chapter 2, WURBench, a set of specifications to follow when benchmarking the wake-up radio hardware and the system as a whole. Through WURBench, we

- provide a well-defined structure, outlining “what to measure” and “recommended practices” for wake-up radio hardware micro-benchmarking.
- offer reliable indicators in terms of key performance metrics, parameters, and tools for researchers to test and fairly compare new solutions against existing ones or baselines when implementations are not publicly available.
- facilitate a repeatable test environment for wake-up radio based systems.

Chapter 3: A Wake-Up Radio COOJA Extension for Simulating Ultra Low Power Radios

As pointed out earlier, one of the goals of this thesis is to provide a tool for assessing wake-up radio based systems. Wake-up radio promises to eliminate the need for periodic channel checks, a process that wastes a lot of energy in duty cycle MACs, especially when the periodicity of data collection is sparse. To assess whether this technology can provide energy savings superior to duty cycled MACs without compromising network performance, reliable tools are required to accurately model the wake-up radio hardware and its performance in combination with the upper layers of the stack. Unfortunately, the existing tools are limited to a few that
Introduction

either do not model the wake-up radio with sufficient accuracy or do not consider the entire stack, ignoring the WSN operating systems. Further, most of these tools are closed-source and implemented in different programming languages requiring complete re-implementation to make it compatible for the final deployment.

To overcome these limitations, in Chapter 3, we present WaCo, an open-source software framework that provides development and evaluation of wake-up radio protocols across all layers of the software stack. WaCo is an extension of COOJA, a standard simulator widely utilized in the WSN community. By developing WaCo, we

- provide a new physical layer interface for the wake-up radio module.
- enable support for multiple radio channels that can be used simultaneously and independently, a feature not available previously in COOJA.
- extend the power tracker module in COOJA to provide an accurate estimation of the energy consumption.
- provide a visual representation of the network behavior on a unified timeline that captures the events generated by multiple radio transceivers. For instance, the wake-up radio and the main data radio.
- enable the developers and researchers to emulate the actual, deployment-ready code of ContikiOS.
- allow researchers to plug in their own wake-up radio module parameters into WaCo simulator.

By developing WaCo, we both advance the state-of-art in wake-up radio simulation and benchmarking tools and demonstrate its practical use in Chapter 4.

Chapter 4: Exploiting WaCo - A Data Collection Case Study

To demonstrate concretely the potential of the wake-up radio technology using our developed tool, WaCo, in Chapter 4, we evaluate a data collection system with two standard data collection protocols. This is achieved through the design and implementation of a new low-complexity MAC layer, W-MAC that allows exploiting the dual-radio interface. On top of this MAC, we integrate the standard network stack for evaluating the complete wake-up radio system and compare its performance to the relevant state-of-the-art protocols. Specifically, we

- propose a low-complexity MAC protocol which exploits the wake-up radio and the main data radio.
- evaluate the performance of networks integrated with wake-up radio over two standard data collection protocols: Contiki Collect and RPL.
- test the scalability of the wake-up radio-based system using different network sizes and data rates.
- show that W-MAC outperforms other state-of-the-art duty cycling techniques in terms of energy consumption, reliability, and throughput.
- demonstrate that wake-up radio can reduce the end-to-end data latency by the factor of 200, while still meeting the needs of several representative WSN applications.
• demonstrate that WaCo provides an effective environment for analysis of protocols proposed for the wake-up radio technology.

Our findings using the WaCo simulator suggest that wake-up radio based systems using simplistic MAC delivers packets with 7.8 times higher reliability, 200 times faster while consuming 20 times less than ContikiMAC. By evaluating different network stacks, we push the envelope of energy efficiency achievable using low-power wake-up radios, successfully demonstrating that WaCo is an effective tool for such analysis.

Chapter 5: Wake-up Lab: Implementing a Wake-up Radio Testbed
The benefits of incorporating wake-up radios with MAC protocols in WSN is enormous as we investigate in Chapter 4, but the effect of the same on real-life deployment is still unknown and unaddressed. Further, simulators have been criticized in the WSN community for not being able to correctly model the dynamics and the complexity of the wireless environment and inability to correctly model the low-level hardware details, often causing inconsistency between the simulated scenario and the real deployments. To complement our simulator WaCo, in Chapter 5, we composed a Wake-up Lab, a modular dual-radio prototype that can be easily integrated into a testbed for characterizing and evaluating the wake-up radio technology under realistic distributed environment. Specifically, we

• integrate a cutting edge wake-up radio module provided by our collaborators at ETH Zürich to a Tmote Sky node forming a dual-radio prototype.
• enable selective activation of the nodes that only need to take part in data collection.
• enable bi-directional communication over wake-up radio, a key characteristic for achieving multi-hop communication.
• detail the software and the hardware integration of the dual-radio prototype that forms the Wake-up Lab.

Chapter 6: Exploiting Wake-up Lab
In Chapter 6, we study the performance of the wake-up radio in a realistic yet controlled environment using testbed composed of Wake-up Lab prototypes. In particular, we demonstrate the effectiveness of the low-complexity W-MAC and provide microbenchmarks for the unicast and broadcast networking using wake-up radios. We also demonstrate the full stack multi-hop capability of the wake-up radio in the testbed for periodic data collection. For the first time, we evaluate the effect of external interference and its impact on the performance of the wake-up radio.

This work advances the state-of-the-art in wake-up radio performance evaluation by demonstrating its practical use in various data collection case studies. Specifically, we

• demonstrate and exemplify that the wake-up radio technology is promising for WSNs through a small-scale indoor desktop testbed.
• demonstrate that using a low-complexity W-MAC, wake-up radio based systems can achieve significant improvements over traditional duty cycling MACs.
• provide microbenchmarks for the unicast and broadcast networking using wake-up
Introduction

• demonstrate the effectiveness of the wake-up radio technology in enabling multi-hop periodic data collection.
• validate our WaCo simulator by comparing simulation results along the testbed.
• evaluate the effect of external interference on the performance of the wake-up radio and the upper layers under controlled laboratory experiments.
• through in-lab measurements provide energy microbenchmarks for the wake-up radio system.

Experiments show that wake-up radio supports on-demand communication over a 3-hop network with an end-to-end latency as low as 78 ms for a 16-bit wake-up packet while dissipating 1.83 µW during periods of inactivity.

Part III: Exemplifying WUR: Energy Neutral and LPWAN

Chapter 7: Plug into a Plant: Plant Microbial Fuel Cell and Wake-Up Radio in Synergy Toward Energy Neutral Sensing

Most IoT devices are battery powered and deployed in remote locations having the battery life expectancy of years rather than hours. With the sheer amount of IoT devices that are expected to be deployed reliance on batteries poses a serious scalability threat due to their costly labor intensive replacement. Alternate solutions such as powering these devices with the energy harvesting sources are essential to maintaining a long term fully autonomous operation. However, the power dissipation of these devices even in the sleep mode is still much higher than that can be harvested from these sources. As a step toward sustainable wireless sensing, in Chapter 7, we present a proof of concept system that uses a novel Plant-Microbial Fuel Cells (PMFC) as a power source. To match the very low power production capabilities of the PMFC, we adopt radical hardware and software solutions to reduce the power consumption. We propose a novel hardware architecture where an extremely low-power switch coupled with a wake-up receiver is continuously powered by a PMFC and a new receiver-initiated MAC-level communication protocol for on-demand data collection. This innovative approach is different from the prior works in that the main node power is completely disconnected during periods of inactivity, allowing the sensing system to be powered with a low-energy source like PMFC. Specifically, we

• design a system that combines multiple, novel hardware and communication technologies to obtain a fully energy autonomous sensing system.
• exploit a novel microbial fuel cell as a power source for sustainable sensing.
• propose a low-complexity receiver-Initiated MAC, WRI-MAC, for on-demand data collection in energy-constrained environments.
• conduct in-lab evaluation of the whole system to validate the feasibility of our proposal for real-world applications.

With the combination of the novel hardware and software techniques, this chapter offers the first steps toward large-scale wireless sensor networks in applications where the sensors
are surrounded by living plants that can provide a green and perpetual power supply. Our proposed design is energy-efficient and flexible allowing it to be powered with a wide variety of energy sources such as photovoltaic cells, thermal, and kinetic for various wireless sensing applications.

**Chapter 8: On-Demand LoRa: Asynchronous TDMA for Energy Efficient and Low Latency Communication in IoT**

Energy efficiency is crucial in the design of battery-powered end devices, such as smart sensors for the Internet of Things applications. Wireless communication between these distributed smart devices consumes significant energy, and even more when data need to reach several kilometers in distance. Low-power and long-range communication technologies such as LoRaWAN are becoming popular in IoT applications. However, LoRaWAN has drawbacks in terms of (i) data latency, (ii) limited control over the end devices by the gateway, and (iii) high rate of packet collisions in a dense network. To overcome these drawbacks, in Chapter 8, we present an energy-efficient network architecture and a high-efficiency on-demand time-division multiple access (TDMA) communication protocol for IoT improving both the energy efficiency and the latency of standard LoRa networks. We combine the capabilities of short-range wake-up radios to achieve ultra-low power states and asynchronous communication together with the long-range connectivity of LoRa. Specifically, we

- propose a new network architecture leveraging short- and long-range technologies for enabling low-latency and energy efficient data collection over a two-hop network.
- design and implement a new receiver-initiated on-demand TDMA MAC for managing channel access and packet collisions. The proposed MAC offers two modalities for node triggering and allows slot allocation for combating packet collisions.
- introduce an analytical model to quantify the data collection latency for on-demand TDMA in broadcast and unicast mode.
- evaluate and validate the proposed network architecture and MAC using an indoor testbed composed of 11 sensor nodes.

The proposed approach still works with the standard LoRa protocol but improves performance with an on-demand TDMA. Thanks to the proposed network and protocol, we achieved a packet delivery ratio of 100% by eliminating the possibility of packet collisions. The network also achieves a round-trip latency on the order of milliseconds with sensing devices dissipating less than 46 mJ when active and 1.83 µW during periods of inactivity lasting up to three years on a 1200-mAh lithium polymer battery.

**Chapter 9: Conclusions and Outlook**

Finally, we conclude and explore possible venues for future research in Chapter 9. We argue that the contributions of this thesis pave the way for future energy-neutral WSNs using the wake-up radio technology. Furthermore, it also emphasizes an important contribution of this thesis in providing the open-source tools to assess and evaluate the attainable trade-offs of utilizing this technology, prior to the actual in-field deployment.
Overviewing WUR Part I
Ultra Low Power Wake-Up Radios: A Hardware and Networking Survey

In battery-powered wireless sensor networks, communication costs dominate system power consumption, motivating research effort on new techniques capable of reducing the footprint of the radio consumption, paving the way for the Internet of Things. The primary challenge is to reduce power consumption when receivers are idle, the so-called idle-listening cost. To overcome this, a plethora of software networking stacks have been proposed including various MAC and routing protocols for managing the channel access and controlling the radio duty cycle. Although these software techniques allow huge savings, the resulting energy dissipation is still much higher than tens of microwatts. This is a far cry from the point to which these devices can be sustained autonomously and possibly achieve energy-neutral operation. Consequently, there is a need for radical hardware solutions in combination with the communication stacks to reduce this power gap.

The ultra-low-power wake-up radio technology represents the ultimate frontier in low power radio communication and promises performance without sacrificing the application requirements. This chapter focuses on this novel hardware approach to reducing the wasteful energy consumption of low-power networks. To offer research progress made in this area, we present a comprehensive literature review (total of 180 research articles) in wake-up radio (WuR) hardware and relevant networking software. Through this literature review, we aim to identify the key characteristics of the wake-up technology (i.e., power consumption, sensitivity), its categorization, and the impact of these characteristics on the different layers of the networking stack.

Contributions. In this chapter, we provide a holistic view of the wake-up radio technology and its characteristics forming a guideline that will assist application and system designers to make appropriate choices while utilizing this technology. Specifically, we:

- survey 75 radio-frequency (RF) (Table 1.8) and 10 non-RF based wake-up radio prototypes (Table 1.9) implemented and tested since 2002.
- present a new taxonomy for categorizing the wake-up radios based on power source,
addressing capability, channel usage, and communication medium.

- identify the key characteristics of the wake-up technology such as power consumption, sensitivity, and data rates and its impact on the networking layer.
- provide a statistical analysis as a guideline to choose wake-up prototypes for various applications regardless of their specific technology.
- provide a new taxonomy to classify various wake-up radio based MAC and routing protocols.
- highlight the main application areas that can benefit from this technology, mapping the suitable platforms and protocols studied above to these applications.
- identify the major issues and challenges that need to be addressed both on the hardware design side as well as the design of the upper layers of the networking stack to achieve energy efficient wake-up radio systems.

Structure of this Chapter. The remainder of this chapter is organized as follows: Section 1.1 depicts the main characteristics of a wake-up radio. Section 1.2 discusses the design space and architecture of wake-up radios followed by some of the main implementation requirements when designing wake-up radio based systems. Sections 1.3 and 1.4 discuss the state-of-the-art wake-up radio hardware designs and comparative analysis between each characteristic, respectively. The integration of different medium access control and routing protocols that are based on wake-up radios are presented in Sections 1.5 and 1.6. In Section 1.7 we briefly discuss some of the application scenarios that can benefit from wake-up radios. Finally, in Section 1.9 we conclude this chapter with open research issues and directions.

1.1 Wake-up Radio Defining Characteristics and Requirements

Before we begin, we summarize in Table 1.1 the key terminology we use throughout our survey to identify components of the wake-up technology.

<table>
<thead>
<tr>
<th>Acronym</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>WuR</td>
<td>wake-up radio, the secondary low-power module</td>
</tr>
<tr>
<td>WuRx</td>
<td>wake-up receiver</td>
</tr>
<tr>
<td>WuTx</td>
<td>wake-up transmitter</td>
</tr>
<tr>
<td>WuS</td>
<td>wake up signal, the message sent by the WuTx</td>
</tr>
</tbody>
</table>

The technology and design considerations for the WuR play a key role in determining the efficiency of low power sensor networks. For the WuR to operate effectively as part of the larger system in a multi-user environment, it should consider the following design points:

(i) **Power consumption.** The most important feature of the WuR is its low power consumption in active mode. In fact, as its use requires the addition of new hardware on top of the main node, the device itself must consume no more than tens of micro-watts. Specifically WuR's active power should be below that of the main radio's sleep power [9] to
provide a positive balance between power saved and used. This is the main specification driving WuR design.

(ii) **Time to wake-up.** The node attached to the WuR must wake-up with minimum latency upon reception of WuS to avoid latency incurred from multi-hops toward the sink and to increase the overall responsiveness of a purely asynchronous network. A range of protocols and applications can benefit from WuR based systems provided that the latency is low. For example, applications in health-care have strict latency requirements and cannot support introducing long delays due to the wake up procedure.

(iii) **False wake-ups and interference.** If all nodes in a sensor network rely on the same wake-up strategy, when the WuTx tries to wake-up a node, it will trigger all the nodes in the neighborhood causing significant energy waste. This causes unnecessary activation of many nodes that should be avoided. There are two possible sources of false wake-ups: 1) nodes waking up when receiving a WuS intended for another node, and 2) interference from nearby devices operating at the same frequency.

To tackle the first, the WuR can employ a node addressing and decoding capability to trigger only the intended node. This allows the WuRx to avoid generating an interrupt if the WuS was not intended for it, however it introduces complexity and often consumption at the WuRx.

Second, interference and background noise that can result in erroneous wake-ups must be filtered. A WuRx must have enough local processing capability to differentiate a WuS from ambient interference, without using the main node's processor. Due to the low power budget, only basic modulation techniques can be used requiring a simple receiver structure [10]. Modulation schemes such as on-off keying (OOK), pulse width modulation (PWM) or amplitude shift keying (ASK) can be used to reduce the possibility of devices interfering with each other. A preamble can be used to differentiate noise from a valid WuS, thus avoiding false wake-ups.

In addition, the WuS must not be missed by the targeted node, as retransmissions are costly in terms of power consumption and latency. To ensure this, a feedback loop such as WuS acknowledgment (WuS-ACK) can be employed by the WuRxs indicating the successful reception of the WuS.

(iv) **Sensitivity and range.** In WuR design, receiver sensitivity is an important parameter as it provides the lowest power level at which the receiver can detect a WuS. Generally, high sensitivity requires more power hungry electronics at the receiver side, thus high power demand. In contrast, low sensitivity for the same communication range will require high radiated power at the transmitter side. Because of this, sensitivity requirements often leads to over-design to ensure reliable communication in adverse conditions.

When the WuR is used to trigger a higher power radio, ideally it should have the same range. Unfortunately this is not reasonable with the power constraints, therefore, most WuR designs target tens of meters of communication range to support many application
scenarios [11]. Very short communication ranges make WuR impractical as high node densities would be required to cover a short distance in a multi-hop fashion increasing node and energy costs. Another side effect of a short communication range is the increase in the hop count messages must traverse to reach the sink, increasing the overall data latency. The wake-up range that can be achieved with most current WuR designs is typically around 30m a value that can be improved by using techniques such as antenna diversity [12] and directional antennas [13].

(v) **Data rate.** The overall power expenditure of a node is not only a function of physical layer properties such as carrier frequency, radio architecture, and the choice of the antenna, but is also a function of the amount of time the radio spends to deliver the data packet over the air. This time depends on the data rate supported by the WuTx and the protocol overhead to establish and maintain the communication link.

Data rate is, therefore, one of the key factors defining the power consumption of WuRs. For example, a WuR with 100 kbps will consume almost half the power of a 50 kbps WuR for the same payload size. For a WuTx with low data rate, the bit duration and the power required to send the WuS will be significantly higher. Due to the longer bit duration, the modulation will keep the transmitter active for a longer time. On the WuRx side, the time and the energy required to generate the wake-up interrupt will also be significantly higher as the receiver and the demodulation circuitry will be active until the transmission ends.

A higher data rate can be seen as a way to improve energy efficiency and to achieve faster wake-up. While a high data rate reduces wake-up latency, a longer bit duration increases the communication range and the reliability of the WuS. At a lower data rate the energy per bit exhibited by the transmitter is higher, which can be accumulated by the WuRx while receiving the WuS. A high data rate is not strictly required by the WuR, especially if it is only used as a triggering device as only a few bytes of data are required.

(vi) **Cost and size.** To integrate the WuR into existing sensor nodes, it should be cost effective. To make the WuR feasible [14], the cost of this additional hardware should be in the range of 5-10% of the cost of the complete sensor node. This is, nevertheless, a loose requirement, as some applications can support higher costs if gains are sufficient. Further, standard off-the-shelf components can be used to speed the development and to reduce the overall cost as compared to designing a single chip solution.

(vii) **Frequency regulation.** Finally, WuR designs should adhere to frequency regulations in industrial, scientific and medical (ISM) bands. It must also comply with communication standards such as the maximum allowed effective radiated power (ERP) used to transmit WuS.
1.2 Architecture and Taxonomy of WuRs

We begin this section by presenting a generic architecture for WuRs and the building blocks that makeup the complete hardware solution. We discuss the functionality of different hardware components and how these devices can be powered and interfaced with traditional sensor nodes. We then move on to present a taxonomy of WuRs, illustrated in Fig. 1.3, showing multiple dimensions that distinguish the designs from one another.

1.2.1 Generic Architecture of WuRs

While WuRs can be constructed in many different ways, each exposing different performance and peculiarities, there are some common building blocks utilized by all designs. Two distinguished implementation approaches have been identified, i.e., prototypes constructed using off-the-shelf discrete components and implementations that exploit CMOS technology for constructing integrated circuits. Power consumption is one of the driving factors behind the use of WuRs due to the energy saving that it can provide. Typically, CMOS implementations achieve improved performance because they better integrate all the components directly on silicon, i.e., more dense integrated circuits result in smaller IC footprints for the same function, hence consuming less power. On the other hand, when using discrete components there are more constraints on each single component selected to build the circuit resulting in worse average performance than CMOS-based designs.

Figure 1.1 – Expanded view of the generic wake-up receiver architecture with energy harvesting capabilities.
Fig. 1.1 illustrates the current architecture and the different functional blocks that form a complete WuRx. This architecture is divided into two sections: the RF front-end and the back-end.

The WuS is first received by the RF front-end via the antenna and then passes through the matching network that filters and boosts the incoming WuS. After input matching, an envelope detector performs signal detection and conversion to baseband signal making the circuit simpler and energy efficient. Then, the signal passes through the amplifiers, often the low noise amplifier (LNA) for increasing the sensitivity of the receiver by amplifying weak signals while meeting noise requirements. The LNA dominates in terms of power consumption. Therefore, while designing ultra-low-power WuRxs it is essential to eliminate some, if not all, of these power-hungry RF components, to reduce power consumption. The voltage multiplier rectifies the RF energy and converts this input signal into a direct current (DC) signal. Usually, the voltage multiplier is constructed by cascading capacitors and zero-bias Schottky diodes. The more energy in the RF signal, the greater the voltage change at the output of the rectifier, which is sensed using a comparator. When there is enough energy to trigger the comparator, the back-end is able to issue an interrupt to the main micro-controller. This back-end can also consist of an ultra-low power micro-controller or correlator circuit that decodes and filters the node address and generates an interrupt.

From the energy point-of-view, one of the hurdles is to supply sufficient energy to operate these devices in a self-sufficient manner without replacing batteries frequently. One of the approaches to achieve this is through Wireless Energy Harvesting (WEH). As illustrated in Fig. 1.1 the subsystem can include one or more energy harvesters that convert the ambient energy into electrical energy. The Generic Energy Harvester module that can power the complete node (including the WuRx, the main transceiver, the main MCU and the sensors) exploiting different energy sources such as magnetic, solar, wind, and mechanical vibrations. Also a separate and standalone RF Energy Harvester, dedicated only for the WuRx, can be employed making the subsystem fully passive i.e., the energy can be scavenged from the incoming WuS itself. The RF-EH unit consists of an antenna and a power management unit (PMU). The PMU basically controls the power supplied to other blocks of the WuRx. In some applications it is possible to directly power the WuRx using the harvested energy from the WuS without energy storage, however, this may not be a viable solution. An alternative would be to include a storage component such as rechargeable batteries or super-capacitors acting as an energy buffer for the subsystem. The main purpose of this storage component will be to accumulate and preserve the harvested energy for later use, thus supporting variations in the RF power level emitted by the WuTx. The wake-up range is relatively short due to free space path loss, low sensitivity, and efficiency of power harvesting at the WuRx. As a result, the WuS is usually transmitted at high power.

The wake-up transmitter, which is usually not detailed in the literature, also plays an important role from the system point of view. Most of the works mentioned in this survey use the standard node’s transmitter as a WuTx such as CC2420 or CC1101 [15, 16, 17, 18, 19, 20, 21].
1.2. Architecture and Taxonomy of WuRs

Finally, we briefly address the content of the WuS, whose packet structure must meet compliance requirements and standards to be used by different technologies. Recent attempts [22] have been made to standardize this for WuRs in medical applications.

A typical WuS packet is illustrated in Fig. 1.2:

(i) **Frame Header.** The frame header consists of the wake-up preamble and start frame delimiter (SFD), a standard byte pattern agreed between the transmitter and the receiver. The preamble contains a set of bits that allow the transmitter and receiver to synchronize their bit intervals and the SFD indicates to the receiver the actual start of the frame and when to start decoding the contents of the packet. The size of the SFD is typically fixed at 1B.

(ii) **Address.** The optional address field contains the destination node ID for identifying the intended receiver. While most designs in our literature survey use node IDs up to 2 bytes [18, 23], the size of this field can be varied depending on the capabilities of the WuRx as discussed below. One of the dimensions of our taxonomy, described next, considers the benefits and costs of addressing inside the packet.

(iii) **Payload / Command.** This field contains the actual application data, command or extra instructions specified by the user or application.

(iv) **Error detection.** Finally, to check data integrity, a frame check sequence (FCS) using a cyclic redundancy code (CRC) is applied. While simple, the CRC provides a high degree of error detection at high speed.

1.2.2 Taxonomy Overview

For the purposes of this survey, we identify four major dimensions for classifying a WuR: power source, addressing capability, channel usage and communication medium. Fig. 1.3 shows multiple options for each of these dimensions and maps, when possible, the WuRs from Tables 1.8 and 1.9. We address each major dimension, beginning with power, as it has the most significant impact on system efficiency.

(i) **Power: Passive.** While the WuR requires power to receive a signal, it does not require continuous power. Instead, it can harvest energy, e.g., from the ambient environment or...
from the incoming wake-up signal itself (Fig. 1.1). The latter case places a burden on
the transmitter side as the WuTx must modulate and transmit the WuS long enough,
typically a few seconds, for the WuRx to detect the signal and accumulate enough
energy to power the trigger circuitry. The longer the WuTx is active, the more power
is consumed. Moreover, this process requires additional hardware at the WuRx side,
thus increasing circuit complexity. The process of accumulating energy also delays
the wake-up of the main node, affecting network performance by increasing latency
and reducing data throughput. Although passive WuRs are energy efficient and offer
extended lifetimes, they often have a shorter operating range than active WuRs, typically
only a few meters.

(ii) **Power: Active.** To address the constraints of passive WuR, the majority of research
efforts focus on fully-active WuRs that receive a continuous, external power supply
either using batteries or a renewable energy harvester hosted on the main node. The
objective of this design is to increase sensitivity, providing longer operational ranges with very low power consumption. 65% of the prototypes that we present in this survey are active WuRs.

(iii) **Power: Semi-active.** In semi-active WuRs, a minority of the components of the receiver, e.g., correlator, comparator and decoder, require continuous power from an external source while the RF front-end remains passive.

Next we consider the recipient of the WuS, specifically whether it can be broadcast-only, with the intent to reach all nodes in range, or can contain an address as shown in Fig. 1.2, intended for a node with a specific ID.

(i) **Addressing: ID-Based.** Optionally, the WuS can contain a bit sequence, typically 8 to 16 bits, for selective node addressing. This increases the size of the packet, but reduces false wake-up and thus overall system energy consumption. After reception of the WuS, the WuRx checks if the signal is intended for it. If so, it triggers and wakes up the main node for data reception. This scheme is referred to as ID-based wake-up and is mostly used to construct unicast-based systems.

It should be noted that energy is consumed to decode a wake-up packet and this is typically performed by an external, low-power micro-controller. Further, the length of the address encoding affects performance. While a long address code is more robust against false wake-ups, it requires a long transmit time, hence more power is consumed. Studies [24] consider the trade-off between the length of the wake-up signal and the energy savings, revealing that the energy used to send the selective wake-up signal only pays off if many nodes are not falsely woken up. In other words, the energy required to transmit the wake-up signal is higher than the energy lost during false-wake up. For low density networks where little data is exchanged, the extra cost of ID-based addressing may not be worthwhile.

(ii) **Addressing: Broadcast.** When the entire neighborhood of nodes receives the wake-up signal, the scheme is referred to as broadcast based wake-up. Broadcast based wake-up can reduce the data latency w.r.t. ID-based systems since the receiving node need not decode a wake-up packet to analyze the recipient ID, but can instead immediately trigger its main radio transceiver after receiving the preamble. However, this is potentially expensive in terms of total system power consumption as all neighboring nodes are woken up.

Next, we turn to how the WuR transceiver utilizes the channel for WuS transmission. Note that the choice of channel or frequency depends on the application and the device to which the WuR is attached.

(i) **Channel: In-Band.** In in-band communication, the main node’s transceiver and the
WuR use the same frequency band, i.e., either 2.4GHz or sub-GHz and can share the same antenna. This technique is cheaper as there is no need for a separate antenna.

(ii) **Channel: Out-of-Band.** In out-of-band systems, the main node and the WuRx are equipped with separate transceivers, each operating at different frequencies. For instance, the WuR prototype presented in [18] operates at 868 MHz while the main data radio operates at 2.4 GHz band. Using frequency or code division techniques such as frequency-hopping spread spectrum, this separate channel can further consist of multiple channels to be able to wake-up specific nodes. The benefits of using separate channels for WuS transmission and data include decreased interference from neighboring nodes operating in the same frequency band and increased signal capacity. However, equipping the WuR with separate channel capability may increase the cost and complexity of the system design.

Finally, we look at the different communication mediums that can be utilized for WuS transmission. Fig. 1.3 does not explicitly show this as a vast majority of the systems we survey fall into a single category, namely RF-Based. Instead, we explicitly indicate the few systems that are not RF-based, and refer the reader to Table 1.9 for details.

(i) **Medium: RF-Based.** If radio signals such as extremely low frequency (~3 kHz) to extremely high frequency (up to several GHz) are used for signaling, the scheme is referred to as RF based wake-up. RF based WuRs have been very widely used and will be discussed in more detail in the next section.

(ii) **Medium: Acoustic.** Acoustic based wake-up such as ultrasonic and audio signals have also been considered. This medium does not require any special infrastructure and the audio signals can be easily generated by speakers or smart phones. Authors in [25, 26, 27, 28] have proposed WuR designs based on sound wave for WuS transmission.

(iii) **Medium: Optical.** Optical as a communication medium for WuRs has also be utilized for indoor sensor networks [29, 30]. For example, authors in [29] have used Free Space Optics (FSO) for sending WuS.

As a system designer, this taxonomy serves as a guide to the available WuR technologies that could meet the constraints of the system. Knowing if continuous power can be provided in a given environment can direct one along the branch with the appropriate power source. Knowing the approximate node density and the expected data rate can serve as indicators for whether unicast, ID-based addressing or broadcast communication is most appropriate. Finally, the amount of expected data to be transferred can lead one to a solution where the WuS is on a same or different channel.
1.3. State-of-the-Art Wake-up Radios

Following this taxonomy for system designers, we now shift focus to the hardware composition of the various prototypes described in the literature. This section offers a comparison of 75 RF-based WuR prototypes, summarized in Table 1.8. To offer a clear picture of the current research landscape, we organize this section first along the power source dimension outlined in the previous section: active, passive, and semi-active systems.

Inside our description of active radios, we offer a categorization, overviewed in Fig. 1.4, that defines the key hardware characteristics. We focus on four: core fabrication technology, frequency usage, address decoding, and modulation techniques.

Following this in-depth discussion of active RF-based WuR, our more concise discussions of passive and semi-active focus on the technology only.

Within each subsection we offer a table categorizing the radios of Table 1.8 according to the options for each feature, highlighting (in bold and yellow) the prototypes that are described in detail in the text. Not all prototypes appear in each, separate table, as not all information is known about each prototype, preventing us from adding it to the tables.

We end the section with a brief summary of non-RF WuRs and a discussion.

1.3.1 Active Wake-up Radios

In this section, we present active WuRs that require an external current source to receive a packet. In most cases, they are used in an always-on manner, but we defer this usage discussion to later. As previously mentioned, we divide our discussion of active WuRs into four categories: the technology used to realize the prototype, operating frequencies that have been...
Table 1.2 – Wake-up radio categorization based on technology.

<table>
<thead>
<tr>
<th>Technology</th>
<th>Reference No.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Discrete</td>
<td>[11],[31],[32],[33],[20],[34],[35],[36],[37],[38],[17],[21],[39]</td>
</tr>
<tr>
<td>CMOS</td>
<td>[40],[41],[42],[43],[44],[45],[46],[47],[48],[49],[50],[51],[52],[53],[54],[55],[56],[57],[58],[59],[60],[61],[62],[63],[64],[65],[66],[67],[68]</td>
</tr>
<tr>
<td>BiCMOS</td>
<td>[69],[70]</td>
</tr>
<tr>
<td>RFID</td>
<td>[71]</td>
</tr>
</tbody>
</table>

utilized in different bands, address decoding techniques, and wake-up signal modulation.

### 1.3.1.1 Technology

The overall power consumption of the WuR depends on its design technology as well as its implementation. Mainly, the chip fabrication technology such as CMOS and BiCMOS for digital circuits and the use of off-the-shelf discrete components for analog circuitry. Although off-the-shelf components allow quick implementation, CMOS based WuRs are more energy efficient and have smaller form factors.

Use of off-the-shelf discrete components and IC packages has allowed designers to simplify and foster rapid prototyping of WuRs with low power consumption, low cost, ease of changes, and reliability.

Petrioli et al. [20] presented a WuRx using fully discrete components that support four different channels in a 2.4 GHz band, thus enabling node addressing. The receiver front end consists of the antenna, low noise amplifier and three power slitters followed by the filter bank. According to the tests, the sensitivity of the WuRx is -83 dBm, while its power consumption is 1620 $\mu$W. The line-of-sight communication range is 120 m, the highest range attained using low complexity receiver design. However, this design also has higher power demand compared to other WuRxs in this category and does not provide the details for the transmission power required to achieve this range.

In recent years power consumption of CMOS devices has greatly reduced allowing researchers to design ultra-low power circuits. There are 29 WuR prototypes based on CMOS technology.

![Figure 1.5 – Discrete components based WuRx architecture [21].](image-url)
In chronological order, the idea of developing and using ultra-low power radios as WuRs was first conceived by the PicoRadio project [40], which proposed a CMOS based node architecture that could be used both as a data radio and as a WuR using a carrier frequency of 1.9 GHz with data rate up to 100 kbps. The PicoRadio has a 10 m range and consumes around 380 $\mu$W from a supply voltage of 1 V. However, not much detail was provided on the hardware side.

Many of the proposed CMOS based prototypes have adopted a heterodyne approach. Heterodyne is a method to convert an incoming high frequency RF signal into one at a lower frequency by mixing two or more signals, where high gain and selectivity could be obtained with relative ease (Fig. 1.6).

Pletcher et al. [41] proposed a 1.9 GHz WuRx chip consuming 65 $\mu$W from a 0.5 V supply in an active mode (receiving and decoding the WuS). The receiver data rate and the sensitivity are 40 kbps and -50 dBm, respectively using OOK for WuS modulation. The design was further improved in [43] by using an "uncertain-IF" architecture to reduce the power consumption to 52 $\mu$W with enhanced data rate and sensitivity of 100 kbps and -72 dBm, respectively. The WuRx consists of BAW resonator for network impedance matching, a front-end-IF (Intermediate Frequency) amplifier for RF signal conditioning and amplification followed by an envelope detector for extracting the shape of the signal and converting it to direct current (DC) for triggering the node's MCU.

A simulation based super-regenerative heterodyne WuRx using duty cycling scheme is proposed by Yu et al. [42]. The super-regenerative WuRx consists of an isolation amplifier as an interface between the antenna and oscillator providing network matching followed by an envelope detector. To reduce power consumption, the oscillator is duty cycled at 10%. With duty cycling, the WuRx dissipates an average power of 56 $\mu$W in listening mode for 100 kbps OOK modulated signal using 2.4 GHz carrier frequency. However, this power consumption increases drastically to 525.6 $\mu$W at 1.8 V supply if no duty cycling is applied. Similarly, the WuRx prototype presented by Yoon et al. [51] also employs duty cycling. The proposed WuRx features two modes of operation; monitoring mode (MO) for receiving the preamble and identification mode (ID) for node address decoding. The WuRx is only duty cycled in the MO mode while in the ID mode the duty cycling is terminated and the data is received at higher data rate. In MO mode this node consumes as low as 8.4 $\mu$W from a 1.8 V power supply offering...
Chapter 1. Ultra Low Power Wake-Up Radios: 
A Hardware and Networking Survey

<table>
<thead>
<tr>
<th>Band</th>
<th>Reference No.</th>
</tr>
</thead>
<tbody>
<tr>
<td>sub-GHz</td>
<td>[66],[17],[11],[48],[31],[55],[32],[33],[34],[36],[21] [51],[62],[45],[59],[44],[38],[61]</td>
</tr>
<tr>
<td>2.4 GHz</td>
<td>[69],[42],[46],[71],[49],[54],[56],[20],[70],[58],[37],[63],[64],[65],[61]</td>
</tr>
<tr>
<td>5 GHz</td>
<td>[60]</td>
</tr>
<tr>
<td>60 GHz</td>
<td>[50],[53]</td>
</tr>
<tr>
<td>Others</td>
<td>[39],[52],[67],[68],[40],[41],[43]</td>
</tr>
</tbody>
</table>

Table 1.3 – Wake-up radio categorization based on frequency usage.

A data rate of 1 kbps. As a consequence of high bit rate of 200 kbps employed for address decoding, the power surges to 1100 µW for the receiver sensitivity of -73 dBm.

Another 2.4 GHz based heterodyne WuRx was proposed by Drago et al. [46]. The WuS is modulated using pulse-position-modulated (PPM) impulse radio modulation scheme. The main building blocks of this WuRx front end are an antenna, a matching network with an on-chip inductor, and a local-oscillator (LO) generator for down-converting the frequency. This IF signal is then amplified using multiple frequency IF-amplifier and then down-converted to baseband by a full-wave rectifier. To achieve low power consumption, the receiver front end as well as the LO generator are duty-cycled at pulse level, thereby reducing the power consumption to 415 µW. The full WuRx prototype achieves a sensitivity of -82 dBm at a data rate of 500 kb/s with energy efficiency of 830 pJ/bit.

There are also designs reported in the literature with power consumption above 1000 µW [56, 68, 45, 70] compared to the ones discussed earlier. The WuRx proposed by Bdiri et al. [68] has attained the longest communication range of 82 m using heterodyne approach at transmission power of 10 dBm with receiver sensitivity of -60 dBm. However, at the same time this particular WuRx has the highest power demand of 5247.5 µW when receiving and decoding the WuS. Other heterodyne based WuRx prototypes achieving power consumption between 22 µW and 100 µW have also been reported in [58, 59, 62, 65].

Radio-Frequency Identification (RFID) technologies have been used as WuR for accomplishing asynchronous multi-modal wake-up where an off-the-shelf RFID tag and an RFID reader has been utilized as a WuRx and WuTx, respectively. Fig. 1.9 illustrates a simple architecture for utilizing RFID technology for WuR systems.

An off-the-shelf active RFID tag based WuRx is simulated in [71]. RFIDImpulse uses an RFID reader as a WuTx to trigger an RFID tag that is attached to a remote sensor node at an operational distance of up to 30 m while consuming 80 µW of power. However, this receiver does not utilize addressing to selectively wake up a sensor node.

1.3.1.2 Operating Frequency

Another layer of complexity is added when considering the transmission frequency of the WuR. Further, if the WuR and the main data transceiver are using different frequencies, each requires a separate antenna for signal detection and separate matching networks. Moreover, the choice of the operating frequency for WuRx is critical as it determines the size of the antenna and the
1.3. State-of-the-Art Wake-up Radios

The sub-GHz WuRx presented by Spenza et al. [35] consumes 1.276 $\mu$W in listening mode. The receiver uses OOK modulation and is made of four main building blocks: a matching network, an envelope detector followed by a comparator and a preamble detector. At the receiver end, the output from the preamble detector is used to interrupt an on-board 8-bit PIC12LF1552 MCU that performs address matching and triggers the main sensor node when a valid wake-up address is received. This sub-GHz WuRx provides high sensitivity and data rate of -55 dBm and 100 kbps respectively, while achieving the maximum wake-up range of 45 m. This design is further improved by Magno et al. [21], which achieves power consumption in listening mode of 0.152 $\mu$W at 32 dBm sensitivity and 1.196 $\mu$W for the -55 dBm version. This particular WuRx has achieved an interesting communication range of up to 50 m and offers data rate of 10 kbps.

Multi-band WuRs have also been exploited to increase the flexibility and to allow interoperability between different frequencies used in WSNs. Roberts et al. [38] propose an ultra-low power WuRx for indoor/outdoor asset tracking systems that consumes only 5 $\mu$W. Authors have developed a tag module that contains a transmitter and two WuRxss integrated in one module. The 434 MHz WuRx is intended for indoor localization, and the 868 MHz WuRx and transmitter are used for the data exchange with the gateways for outdoor localization. The WuRx continuously scans the channel for any predefined wake-up sequences. As soon as the received sequences matches to the reference sequence, a digital control signal is generated immediately to trigger the sensor node. In addition, the proposed WuRx also provides a received signal strength indicator (RSSI) value of the received WuS with 3 bits quantization. A similar prototype for asset tracking applications has also been reported in [61]. The Fraunhofer WakeUp-Receiver [61], which is based on 130-nm CMOS technology, operates in the 868 MHz and 2.4 GHz frequency bands and features -80 dBm sensitivity with 16-bit selective wake-up ID. At a data rate of 1 kbps this prototype consumes 7.5 $\mu$W of power with response time of 30.3 ms. However, no detailed operational communication range tests or complete WuR system design is provided.

To achieve relatively high date rates, a WuRx operating in millimeter-wave band (60 GHz) for short-range applications is proposed in [50]. This duty cycled WuR consists of a 4-path phase array transmitter and a 4-path receiver. By applying OOK modulation for switching the biasing of power amplifiers a 1 Gbps data rate is attained. The WuRx side is built of an injection-locking ring oscillator (ILRO), a frequency mixer and a low pass filter. The performance of this receiver is evaluated in simulations and has achieved a power consumption of 230 $\mu$W with sensitivity of -62 dBm ranging up to 0.2 m. Instead, Wada et al. [53] presented a first successful WuRx prototype operating at 60 GHz. To achieve low power consumption, a power reduction circuit has been implemented that turns off the injection locking oscillator when there is no WuS detected. The fabricated WuRx has a high sensitivity of -68 dBm for a 350 kbps OOK WuS while consuming only 9 $\mu$W from a 1.5 V supply. Another WuRx that operates at 5.8 GHz has been reported in [60] but has lower sensitivity of -44 dBm. Note that for the latter two designs, the operational range of the system as a whole.
Chapter 1. Ultra Low Power Wake-Up Radios: 
A Hardware and Networking Survey

Table 1.4 – Wake-up radio featuring address decoding.

<table>
<thead>
<tr>
<th>Technique</th>
<th>Reference No.</th>
</tr>
</thead>
<tbody>
<tr>
<td>MCU</td>
<td>[39],[11],[33],[35],[36],[21],[54]</td>
</tr>
<tr>
<td>Correlator</td>
<td>[48],[55],[59],[69],[49],[63],[57],[70]</td>
</tr>
<tr>
<td>AS3930</td>
<td>[17],[31],[32],[34],[68],[37]</td>
</tr>
<tr>
<td>Others</td>
<td>[51],[44],[62],[20]</td>
</tr>
</tbody>
</table>

authors have not published any operational distance.

There are also few WuR designs for WBAN applications that use very low frequency for communication. One of the advantages of operating at lower frequency bands is that it enables lower signal attenuation and interference than the traditional operating bands such as 2.4 GHz. Cho et al. [52] proposed the WuRx prototype targeting WBAN applications while operating at 45 MHz. The proposed WuRx uses ILRO instead of RF amplifier to reduce power consumption. The WuS is modulated using Frequency Shift Keying (FSK) and is demodulated by a low power Phase Locked Loop (PLL) demodulator. This prototype features a receiver sensitivity of -62.7 dBm with data rate of 200 kbps while consuming as low as 37.5 µW from a 0.7 V supply in an active mode.

Recently, Juha et al. [39] proposed a 28 MHz always on WuR design based on super-regenerative principle for human body communications. To achieve low energy consumption and high sensitivity, the WuR uses loose synchronization and employs self-quenching while operating at 1.25 kbps. With real-life experiments the proposed designed consumes 40 µW and achieved receiver sensitivity of -97 dBm.

1.3.1.3 Address Decoding

Next, adding node address decoding capability to the WuRx requires additional components at the RF back-end. Usually, a low power micro-controller (MCU) or correlator is employed for decoding. However, this comes with some trade-offs, highlighted in this section.

Some WuR designs use a secondary, dedicated low-power micro-controller to decode the address code. An example is shown in Fig. 1.7, illustrating the integration of low power MCUs with WuR prototypes. As will be discussed later, this extra hardware contributes to energy overhead when used for address decoding.

Using a separate MCU for address decoding and interference filtering is reported in [11]. In this prototype, authors have integrated a PIC12F683 MCU to detect and decode a WuS after signal rectification and amplification, and notify the more powerful AT-mega128L processor of the main node through an interrupt. Due to intervention of this extra PIC12F683 MCU, the overall power consumption of the WuRx increases from 171 µW in listening mode to 819 µW at 3 V when used for address decoding. The proposed prototype was only able to communicate up to 2 m with receiver sensitivity of -51 dBm at data rate of 0.86 kbps using OOK modulation. Another prototype with similar communication range is presented by Bdiri et al. [36], but
1.3. State-of-the-Art Wake-up Radios

has lower power consumption of 0.69 µW operating in 868 MHz band. Authors have also compared two different WuS decoding techniques, one with MCU and the other using AS3932 (a detailed discussion of the AS323X series will follow at the end of this subsection). The results indicate that using AS3932 for address decoding leads to an additional power consumption of 3.9 µW than the MCU.

Other designs that exploit MCU for address decoding while achieving power consumption below 15 µW can be found in [54, 33]. However, these designs do not provide any detail on operational distance that can be achieved with these WuRx.

Instead of using MCUs for address decoding, an energy efficient way is to use correlator circuit for address matching. In the correlator circuit, the node address is stored in the reference signal buffer and the input bits from the WuS are correlated against the reference signal. When a new bit is available, all the samples are shifted one position in the correlator and are compared to the pre-stored one. If the stored and the incoming bits are a match, the wake-up interrupt pin is asserted. Fig. 1.8 depicts a simple "matched filter" based parallel correlator concept used to decode address in a WuS.

Mark et al. [69] simulated one of the first correlator based approaches for decoding node address in a WuRx system and features sensitivity of -50 dBm. The wake-up circuit is composed of a 2.4 GHz matching network, an envelope detector, and low noise amplifier. The output signal from the amplifier is then fed into the correlator circuit to compare the signal to a predefined sequence. However, no values have been reported for power consumption, data rate or WuRx communication range.

Hambeck et al. [48] presented a complete prototype of WuRx employing a 64-bit mixed signal correlator for address matching. At 868 MHz, the design features a receiver sensitivity of -71 dBm and an outstanding measured free-space radio link distance of up to 304 m at transmission power of 6.4 dBm. At this conditions, the WuRx dissipates only 2.4 µW at supply voltage of 1 V.

Milosiu et al. [55] presented a 31-bit correlator based WuRx with scalable data rate and -83 dBm sensitivity. The prototype is fabricated in a 130-nm CMOS technology and requires 4.75 µW from a 2.5 V supply at a data rate of 128 bps. Compared to the other WuRx prototypes
found so far in the literature, the proposed receiver has obtained the longest line-of-sight communication range of 1200 m for a transmit power of 10 mW. Recently, authors have also proposed a 2.4 GHz version of the OOK WuRx that obtains a power consumption of 7.25 µW with reaction time of 30 ms. However, no details on the receiver range is provided. Other low power designs have also been reported in [49] obtaining power consumption below 3 µW.

There are many proposals in the literature where authors have also resorted to a commercially available WuRx chip for address decoding into their prototypes [15, 32, 34, 36, 17, 68]. The AS393X series from Austria Microsystems [72] is a 3D low-power low-frequency Amplitude Shift Keying (ASK) WuRx capable of generating a wake-up interrupt upon detection of signal at a carrier frequency between 15-150 kHz. The AS393X also allows duty cycling the WuRx in order to save energy and includes an integrated correlator to implement a 16 bit or 32 bit wake-up address decoding scheme. This WuRx has maximum sensitivity of -69 dBm with current consumption varying from 1.7 µA up to 12 µA at 3 V power supply. With these characteristics, the AS393X has average performance compared to other experimental WuR prototypes found in the literature.

Sutton et al. [17] presented the first practical application of WuRx that can be used both for initiating the communication and as a full data radio. The OOK WuR transceiver is designed using the off-the-shelf components and leverages AS3930 ASK receiver for address decoding. The CC110L transceiver is used as a WuTx and shares the same antenna with the WuRx module. The OOK receiver is able to receive a 16-bit data packet at a maximum data rate of 8.192 kbps, and features an ultra-low power consumption of 8.1 µW measured at 3 V. The OOK receiver sensitivity is approximately -52 dBm and achieves a 30 m line-of-sight communication range in an outdoor field.

In [34], Oller et al. proposed WuRx incorporating AS3933 for IEEE802.11-enabled wireless access points. This prototype features a WuRx sensitivity of -52 dBm and the total power consumed by the design is 10.8 µW in sleep mode and 24 µW in an active mode with address decoding. Similar wake-up range of up to 40 m has been observed making these prototypes suitable for implementation that require long range communication with minimum power consumption without relying on MCU for address decoding.

Figure 1.8 – Node address comparison using "matched filter" correlation detector.
1.3. State-of-the-Art Wake-up Radios

Table 1.5 – Various wake-up signal modulation techniques.

<table>
<thead>
<tr>
<th>Modulation</th>
<th>Reference No.</th>
</tr>
</thead>
<tbody>
<tr>
<td>OOK</td>
<td>[67],[66],[40],[41],[43],[42],[56],[58],[64],[50],[53],[47],[51],[17], [31],[32],[34],[68],[37],[55],[69],[49],[63],[57],[70],[20], [39],[11],[35],[36],[21],[54],[62],[65],[48]</td>
</tr>
<tr>
<td>ASK</td>
<td>[71],[60],[44],[33]</td>
</tr>
<tr>
<td>FSK</td>
<td>[52],[45],[59],[65]</td>
</tr>
<tr>
<td>Others</td>
<td>[46]</td>
</tr>
</tbody>
</table>

Microsemi based ZL70103 [73] is another off-the-shelf transceiver chip that incorporates a WuRx designed for implantable medical devices. The out-of-band WuRx operates at 2.45 GHz with an average current consumption of 290 nA while sniffing the channel once a second. It allows to initiate the communication between the implanted device and the base station transceiver using specially coded WuS from the 2.45 GHz base station. So far, none of the prototypes presented in this survey use ZL70103, however it is an interesting option for BAN applications.

Other address decoding techniques using Bloom filters [44], shift registers [62], flip-flops, and filter banks [20] have also been exploited. Takiguchi et al. [44] have simulated a Bloom filter based wakeup mechanism for WuRx. A node identifier-matching mechanism uses Bloom filter implemented with a simple circuit that only uses an AND circuit. For a bit rate of 40 kbps, the listening power consumption of the receiver is 12.4 $\mu W$ and in an active state the circuit consumes 368.1 $\mu W$ from a 1.8 V supply.

1.3.1.4 WuS Modulation Technique

Circuit complexity and reproducibility are the key factors that allow designers to tune and simplify WuRs enabling faster prototyping. Nevertheless, this is dependent on the modulation technique used for WuS transmission, the architecture of RF front- and back-end, and the choice of frequency. To meet the requirement of ultra-low power consumption, various modulation schemes such as on/off keying (OOK), Amplitude shift keying (ASK), or Frequency shift keying (FSK) have been exploited for the wake up signals.

As seen from Table 1.5, most of the WuR designs have modulated RF signal using OOK before reception by the wake-up receiver. In OOK modulation scheme the signal information is delivered using ‘1’s or ‘0’s. The source node transmits a large amplitude carrier when it wants to send a ‘1’ and nothing is send for ‘0’, i.e., the transmitter is turned off. Thus, allowing systems to save on transmit power when (not) sending ‘0’s. On the receiver side this signal is sensed by the rising edge of the digital signal from low to high indicating that a valid signal has been received via the antenna. This has enabled OOK hardware implementations to be relatively straightforward due to their low implementation cost for battery-operated applications. Usually, few discrete components are enough to construct OOK signal detection circuitry as outlined in [18, 35]. The super-generative [53, 52], tuned RF [40, 21, 34], or uncertain-IF architectures [56, 70] have been popular solutions to demodulate an OOK signal.
In [21], the WuRx consumed 1.2 \( \mu \text{W} \) and achieved a sensitivity of -55 dBm at a data rate of 10 kbps to demodulate a 868 MHz OOK signal.

ASK is another popular modulation technique used by WuR hardware designers. Similar to OOK, the information in ASK is also transmitted using ‘1’s or ‘0’s. However, instead of keeping the transmitter off when indicating bit ‘0’, it transmits small amplitude carrier in its simplest form.

For FSK demodulation, WuRxs are based on frequency discrimination architecture. In [45], the WuRx consumes 2700 \( \mu \text{W} \) to demodulate a 0.915 MHz FSK signal. The overall receiver sensitivity is -89 dBm at a data rate of 45 kbps.

Most of the designs surveyed in this chapter are compatible with only one modulation technique. Therefore, to make a WuRx compatible with other types of signals, Taris et al. [65] proposed a first dual modulation based WuRx. This proof of concept features an LC oscillator coupled with an envelope detector implemented in a 65 nm CMOS technology. The circuit consumes 120 \( \mu \text{W} \), and properly demodulates OOK and FSK modulated signals at 2.4 GHz with data rate up to 500 kbps.

Although, ASK offers better noise immunity compared to OOK at a lower cost than FSK, it has higher power consumption demand than OOK based WuRxs (refer to Table 1.8 and Fig. 1.12).

### 1.3.2 Passive Wake-up Radios

This section discusses prototypes that harvest and power the wake-up circuitry entirely from the RF signal. In this way, passive WuRxs have the advantage of not consuming any energy from the node battery making the design energy neutral.

The first proof-of-concept passive WuRx design operating at a frequency of 433 MHz was presented by Gu and Stankovic in 2005 [74]. The WuRx is powered using radio signals and is able to trigger a wake-up interrupt once enough energy has been harvested and stored on the capacitor. The proposed WuRx uses a charge pump approach consisting of capacitors and zero-bias Schottky diodes acting as a voltage multiplier and a radio trigger circuit. This WuRx also features the addressing capability by transmitting the WuS at different frequencies to activate the targeted node, reaching an operating range of around 3 m. The power consumption of the WuRx in idle mode (i.e. while harvesting energy from the WuS) is 145 \( \mu \text{W} \), and the design was only evaluated through SPICE circuit simulations.

Another battery-less WuRx operating at 900 MHz band was proposed in [76]. This passive

<table>
<thead>
<tr>
<th>Technology</th>
<th>Reference No.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Discrete</td>
<td>[74]</td>
</tr>
<tr>
<td>CMOS</td>
<td>[75],[76],[77],[78],[79],[80]</td>
</tr>
<tr>
<td>RFID</td>
<td>[81],[82],[83]</td>
</tr>
</tbody>
</table>
CMOS chip consists of an RF front end and a digital baseband with non volatile memory. The radio block includes a voltage multiplier for rectifying and amplifying the RF energy, a voltage limiter, demodulator and modulator circuits, and a ring oscillator. Authors have designed the voltage multiplier by cascading 4-stage voltage doublers using Schottky diodes and capacitors. Using ASK modulation technique, the prototype achieved a sensitivity of -17 dBm with power consumption of 2.64 µW. However, no details regarding the communication range and data rate are provided.

Kamalinejad et al. [78] presented a passive 868 MHz WuRx front end that also harvests energy from the RF signal. The building blocks consist of an antenna, matching network, voltage multiplier and data slicer (comparator and the reference generator). An RF-to-DC converter is used to produce the envelope of the OOK WuS and converts the RF signal to a DC voltage that is used to power the data slicer circuitry. A fraction of this DC output is then compared with the generated reference to produce the wake-up interrupt signal. Using simulations, the proposed design exhibits a sensitivity of -33 dBm and 100 kbps data rate without any node addressing capability. In turn, Zgaren et al. [79] took the idea of Kamalinejad et al. [78] and have proposed a passive WuRx prototype for implantable devices operating in 902-925 MHz band. This prototype has a power dissipation of 0.2 µW for a data rate of 100 kbps at -53 dBm sensitivity. However, the latter design is only evaluated using simulations. Other passive WuRx that are based on CMOS technology can be found in [75, 77, 80]

Ba et al. [83] proposed a passive RFID device called WISP-Mote by combining a Wireless Identification and Sensing Platform (WISP) to a Tmote Sky sensor node. WISP is powered wirelessly by an off-the-shelf UHF RFID reader to generate an external interrupt to a Tmote Sky, achieving communication range of up to 5 m. Upon successful activation, WISP transmits the sensor data using the main node's 2.4 GHz CC2420 transceiver. WISP supports both broadcast and ID-based wake-ups.

Passive RFID based systems usually have a communication range only up to few meters, thus making it difficult to implement a multi-hop sensor network. Therefore, to realize a multi-hop wake-up using RFID technology, Chen et al. [81] proposed an enhanced version of WISP-Mote with energy harvesting capabilities called Multi-hop-Range EnhAnCing energy Harvester-Mote (MH-REACH-Mote). MH-REACH-Mote is equipped with both a WuTx and a passive WuRx. The WuRx side is same as WISP-Mote while UHF RFID reader has been used as the WuTx providing an option for an addressable wake-up with high transmission power. This prototype achieved the maximum wake-up range of 9.4 m when the WuS was transmitted for 10s. Donno et al. [82] also proposed a passive WuRx prototype using commercial 868 MHz UHF RFID tag and RFID energy harvester for achieving long distances. Authors implemented a wake-up strategy called Enhanced Write Wake Up (E-WWU) that supports both broadcast communication and node addressing achieving a range of 22 m with transmission power of 30 dBm. The WuRx side consumes 54 µW for receiving and decoding the WuS.

From the above designs, it is evident that CMOS technology is more popular for implementing passive WuRx due to its low power consumption. RFID has also been utilized since it already
Table 1.7 – Semi-active wake-up radio designs.

<table>
<thead>
<tr>
<th>Technology</th>
<th>Reference No.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Discrete</td>
<td>[15],[84],[85],[86],[18],[87],[88]</td>
</tr>
<tr>
<td>CMOS</td>
<td>[89],[90],[91],[92],[93],[94],[95],[96],[97],[98],[99],[10]</td>
</tr>
<tr>
<td>RFID</td>
<td>[100]</td>
</tr>
</tbody>
</table>

provides energy harvesting capabilities thus reducing the implementation time.

1.3.3 Semi-active Wake-up Radios

To operate in the nano-Watt power range, the majority of the proposed design approaches are semi-active, in which only a few components of the receiver front-end are battery powered while the rest of the components are fully passive. Using passive circuitry allows reducing the power consumption, but at the price of short communication range and reduced receiver sensitivities. For the radio front-end, the most common approach is to implement an envelope detector using passive components such as Schottky diodes, MOSFETs or ICs followed by the active components such as correlators and comparators to generate an interrupt to the main MCU. Next, we present WuR prototypes that utilize such architecture.

Malinowski et al. [100] reported the first "quasi-passive wake-up" system utilizing RFID technology called CargoNet. CargoNet employs a 300 MHz RFID tag to trigger an ultra-low power MSP430 based sensor node. The WuS detector circuit consists of an LC tank with an autotransformer for amplifying the signal received at the antenna followed by an envelope detector and micro-power amplifier for voltage gain. After the main sensor node is activated, data is communicated using a 2.4 GHz CC2500 transceiver. The proposed WuRx design consumes 2.8 $\mu$W in listening mode. The average power consumption of CargoNet is 23.7 $\mu$W when the node is active and receiving the data packet via the main transceiver. At maximum sensitivity of -65 dBm, the WuRx is able to detect an OOK modulated WuS up to a distance of 8 m.

Ansari et al. [18] presented a radio triggered wake-up circuit attached to a TelosB node and exploited its main MSP430 MCU for address decoding. The WuTx uses an additional out-of-band 868 MHz CC1000 transceiver for generating WuS using Pulse Interval Encoding (PIE) scheme and a frequency amplifier for communication range extension. The main buildings blocks include an impedance matching network, a voltage multiplier and a digital comparator interfaced to the main MCU. The matching network is constructed using discrete components.
such as capacitors and inductors while the 5-stage voltage multiplier uses RF Schottky diodes. The MCU tracks the low-to-high transitions and the time intervals between the PIE signal to successfully decode the data. In case the wake-up packet is not addressed to the node, it switches back to the sleep mode. Otherwise, the node triggers its main CC2420 transceiver for data exchange. The WuRx in listening mode consumes only 2.628 $\mu$W and the micro-controller consumes 1020 $\mu$W when it switches from sleep to active mode for address decoding. Empirical measurements using simulation shows that the proposed WuRx has an operating range of 10 m for the 500 $\mu$W transmission power.

Le-Huy et al. [10] also simulated a semi-active WuRx that uses correlator as a decoder. This work has become one of the reference designs for several newer proposals, since authors have outlined the complete steps from signal detection to address comparison. The proposed architecture consists of a shared antenna between the WuRx and the main transceiver, impedance matching network and zero-bias Schottky diode based envelope detector. It is followed by an address decoder circuit that has three subsystems: the amplifier stage, the PWM demodulator and the correlator circuit consisting of shift register and a logic comparator. The power consumption of the proposed architecture is 19 $\mu$W at a data rate of 50kbps with receiver sensitivity of -53 dBm. Using Pulse Width Modulation scheme, the receiver exhibits a maximum range of 5 m for 2.4 GHz band.

Ammar et al. [84] also proposed a semi-active 868 MHz WuRx that uses Flip Flops for address decoding and dissipates only 13.41 $\mu$W. However, this design is only evaluated using simulations. Other simulated designs based on semi-active WuRxs can be found in [99, 91, 90, 86].

Gamm et al. [15] proposed the first in-band sub-Carrier modulation WuRx system based on AS3932 (Fig. 1.10). In the wake-up mode the WuS is directed to the AS3932 WuRx for envelope and address decoding after impedance matching and demodulation of OOK signal. First, AS3932 extracts the 125 KHz signal from the 868 MHz WuS and then the original data is decoded for address comparison. Once the address is matched, the main node is triggered. Afterwards, an antenna switch is utilized to bypass the WuRx and the data exchange takes place using the main CC1101 transceiver. The main radio is also utilized as a WuTx to generate the WuS, thus the first complete WuR transceiver. The WuRx circuitry is supplied with 3 V battery and has an active power consumption of 7.8 $\mu$W while the total node consumption is 44 mW. For an output power of 11 dBm at the WuTx, the maximum wake-up distance was 45 m at a data rate of 250 kbps and sensitivity level of -52 dBm. The design by Gamm et al. [15] has become the starting point for other AS393X based WuR systems such as the ones presented in [31, 34, 37].

The most energy efficient semi-active WuRx proposed to-date is presented by Roberts et al. [92]. The energy is harvested from the RF signal and then the received voltage is boosted using resonant tank before supplied to the active part of the circuit. This 915 MHz band WuRx achieved a communication range of 1.2 m at transmission power of 0 dBm. The whole CMOS based WuRx provides a data rate of 100 kbps using OOK modulation while consuming
only 98 nW in active state. However, the WuRx does not support node addressing as per the implementation.

Yet another ultra-low power WuRx intended for WBAN is presented in [85]. The proposed design uses Gaussian On-Off Keying (GOOK) and Pulse Width modulation (PWM) for decoding and encoding the preamble signal, respectively. This receiver has higher power consumption of 2.67 µW than that proposed by Roberts et al. [92] in listening mode, but achieves a longer communication range of 10 m for WuTx output power of 10 dBm. The WuRx also operates in a sub-GHz frequency band (433 MHz) and has receiver sensitivity of -51 dBm. The address decoding is handled by the MCU and the authors have not provided any details of its related power consumption.

To increase the flexibility of WuR, multi-band WuRs have also been exploited to allow interoperability between different frequencies used in WSNs. Huang et al. [96] propose a radio-triggered WuRx able to operate selectively at 915 MHz and 2.4 GHz band. After input matching, an envelope detector suppresses the fundamental tone to the required frequency followed by a baseband amplifier for filtering and amplifying the WuS. This WuRx consumes 51 µW for 100 kbps OOK modulation featuring receiver sensitivity of -75 dBm in the 915 MHz band and -64 dBm in 2.4 GHz band, respectively.

Oh et al. [97] presented a tri-band 116 nW WuRx with 31-bit Correlator with interference rejection capabilities. The WuRx front end operates in the 402 MHz MICS band and the 915 MHz and 2.4 GHz ISM band with sensitivities of -45.5 dBm, -43.4 dBm and -43.2 dBm, respectively. The chip consists of an input matching network for filtering and boosting the incoming WuS and a 30-stage passive rectifier for down-converting the RF signal to baseband, which is then sensed by a comparator. Finally, a bank of 124 correlators is implemented to compare the wake-up sequences with a programmable wake-up code. The wake-up interrupt is generated only when a correlation value exceeds a user-programmable threshold.

Recently, another dual-band WuRx that operates in 868 MHz and 2.4 GHz band has been proposed in [86]. The WuRx front end consists of a dual-band antenna and matching network.

Figure 1.10 – Wake-up receiver prototype utilizing Austria Micro Systems AS393x WuRx [15].
with a passive envelope detector. The back-end consists of an interrupt/data generator and an ultra-low power micro-controller for address decoding and generating interrupt to the sensor node. The receiver is tuned to use OOK modulation for WuS with sensitivity of -53.4 dBm and -45.2 dBm at 868 MHz and 2.45 GHz, respectively. Simulation results demonstrate that the proposed solution consumes 1.276 $\mu$W while listening the channel and this power consumption increases to 70.6 $\mu$W when the MCU is decoding the address with supply voltage of 1.8 V.

### 1.3.4 Non-RF Based WuRs

While RF based WuRs have been most widely researched, some authors have proposed an unconventional method to communicate with the WuRx by exploiting different transmitting mediums like optical or ultrasonic signals. For this reason it is quite inappropriate to call such devices WuR, but still some solutions are interesting and expose characteristics that are comparable with RF based WuRs discussed so far. In fact the communication range that could be achieved with these type of wake-up transceivers are similar to typical RF based WuRs while also exhibiting similar power demands. The two main drawbacks are that some of these devices require directionality and/or line-of-sight (LOS) communication between transmitter and receiver, making them inappropriate for some applications. The complete list of all the WuRs in this category is presented in Table 1.9.

Hakkinen et al. [101] proposed one of the earliest designs where infrared is utilized to transmit WuS. The WuTx is basically an IR LED that is switched on and off by the micro-controller. On the WuRx side, a photo-detector is used for receiving the signal and a transimpedance amplifier converts this signal into voltage to generate an interrupt. It achieves operational range of up to 30 m with an IR remote controller by matching its carrier frequency with the WuRx. The prototype consumes 12 $\mu$W when listening for the WuS at a supply of 3 V. Unfortunately, the wake-up circuit is very sensitive to external light and is vulnerable to noise while requiring direct LOS between nodes.

The proposal by Mathews et al. [29] utilizes Free Space Optical (FSO) as a secondary wake-up channel. The power consumption of the proposed FSO WuRx is 317 $\mu$W in listening mode and attains a LOS range of 15 m at a transmission power of 16.5 mW. Due to low gain bandwidth of the operational amplifiers, the system suffers from low data rate of 2 kbps. Optical based designs implicitly feature node addressing through directional communication, however, it is not clear how this design would perform when the nodes are not perfectly aligned and how to communicate with multiple nodes, if required.

Another optical based WuRx is presented in [30] called Free-space Low-Power optical Wake-up and has an ultra low power of only 695 pW in standby mode and 12.2 nW in active mode. The WuR supports three different light sources for extending communication range. Using 0.5 W LED the wake-up range is 0.2 m, 6 m with 3 W LED with focus and extends to 50 m when a 3 mW green laser is utilized as WuTx. In contrast to [29], FLOW features a 16-bit
node addressing capability. However, similar to [29], the WuR system requires direct LOS for transmitting WuS and supports very low bit rate of 91 bps. Moreover, to achieve long range communication, proper physical alignment between the optical WuTx and WuRx is also required.

Sanchez et al. [28] have presented an asynchronous acoustic-triggered wake-up modem for underwater sensor networks. Using this technique, the WuRx is programmed to react to acoustic signals at a certain frequency, reactivating the node if needed. The WuRx consumption is 10 $\mu$W in listening mode. The authors have also integrated AS3933 for 16-bit node address recognition. With a transmission power of 108 mW, an underwater communication range of 240 m has been achieved.

An ultrasonic WuRx working at 40.6 kHz is proposed in [25]. It uses piezoelectric transducer that converts the mechanical energy into electrical energy for generating wake-up interrupts. The design is based on heterodyne architecture and the overall receiver power consumption is 4.8 $\mu$W in listening mode. When exciting the transmitter with an electrical signal power of 16 $\mu$W, it achieved an operational range of 8.6 m. However, the WuRx has very low bit rate of 250 bps. Another prototype using ultrasonic signals is presented by Lattanzi et al. [26]. Unlike [25], this design supports out-of-band addressing scheme for selective awakening. It uses off-the-shelf components and requires 1.748 $\mu$W in listening state and around 14 $\mu$W when active. This design is suitable for ranging applications that require distance up to 10 m. The WuTx takes 0.5s to transmit an 8-bit address and requires 75 $\mu$W of power at bit rate of 16 bps.

The design by Hoflinger et al. [27] presents an acoustic WuRx operating at 18 kHz for controlling devices and appliances at home. The audio signal is sent using a smart-phone speaker and a micro-electromechanical system (MEMS) microphone is used to detect the audio signal on the WuRx. The microphone transducer converts this acoustic signal into an electrical signal, which is then fed into AS3933 WuRx IC that detects a valid frequency of 18 kHz and triggers the micro-controller. A wake-up range of 7.5 m was achieved using this setup. The WuRx consumes 56 $\mu$W in listening mode while the consumption hikes to 440 $\mu$W in active state when receiving the signal using PWM modulation. This design was further improved in [102], which operates at 20 kHz audio signals and features node addressing. To reduce the power consumption compared to [27], the power amplifier and the microphone are duty cycled using the micro-controller. Using this technique, the proposed design attains a power consumption of 45 $\mu$W in listening mode and 420 $\mu$W in active mode. An average wake-up range of 10 m using smart-phone as a sender was achieved.

Recently, Carrascal et al. [103] have developed a visible light communication (VLC) based WuR system. This system uses an off-the-shelf indoor solar panel as a receptor and energy harvester to power the WuRx. The WuRx is also coupled with AS3933. At the transmitter side, a 10 W LED is modulated using OOK at a frequency of 21 kHz to transmit WuS. In an indoor environment, with short bit duration the prototype achieved 7 m range while with longer bit duration maximum achievable range was 14 m. This VLC based WuR consumes 19.2 $\mu$W in
listening mode and \( \sim 95 \, \mu W \) when receiving and decoding the WuS. The transmission power required to achieve the above range was 87.9 mW at a data rate of 1.12 kbps. The proposed system is suitable for indoor applications only and allows to harvest energy from the indoor lights for energy-autonomous operation of the WuRx.

### 1.3.5 Summary

In Section 1.2, we considered different physical layer characteristics of wake-up receivers, each designed and tested in separate ways. We next discuss some of the advantages, disadvantages, and features for each category.

From the application point of view, RFID-based WuR systems are suited for mid-range applications such as health monitoring, inventory monitoring, or environmental applications \[83\]. Nonetheless, the maximum communication range achieved so far has been 30 m using an active RFID tag \[71\]. As active RFID tags are costly and require more power, such WuR designs may not be suitable for applications that require extended lifetime with minimum maintenance. Moreover, the communication range of RFID devices are related to antennae size: the bigger the antenna the more power can be transmitted thus the longer the range. For WuR based applications that demand small form factor, this could be a hindrance and may force designers to opt for other technologies such as system-on-chip, which may be suitable for a wide range of applications. In addition, for passive RFIDs and EH-WuRx not all energy is absorbed by the receiving end resulting in a phenomenon known as backscattering. Thus, WuS are transmitted at high power and usually take a few seconds to accumulate and recharge the capacitors for powering up the circuits. This, in turn, affects the wake-up range and the latency of the system as a whole.

Most active WuRs use CMOS technology and a heterodyne approach. While these heterodyne-based WuRs offer superior sensitivity and data rate, most lack node addressing capabilities and information on their operational range. This category of WuR also features the highest power consumption of up to a few milliWatts \[68, 70\] as the heterodyne approach requires some active components such as IF-amplifiers and mixers. It has also been noticed that some of these designs operate in lower non-ISM bands such as 45 MHz \[52\] or 1.9 GHz \[41\] making them inadequate for medical applications. By contrast, lower operation frequency may enable the design of transceivers that consume less power than transceivers in higher frequencies. Moreover, it enhances security compared to traditional wireless technologies for WBAN by making the radio signal more difficult to eavesdrop.

It has also been observed that the use of a secondary MCU for address decoding allows faster prototyping at the receiver back-end. On the other hand, the introduction of this extra hardware adds to the overall power overhead and may not be applicable for applications that have strict power requirements. However, due to advancements in miniaturization, the power consumption of these MCUs has drastically reduced over the years making it possible to integrate with WuRx while still achieving power consumption below 10 \( \mu W \).
The choice of modulation scheme also affects the overall WuRx performance. If a complex modulation technique like FSK is utilized, this demands complex circuitry at the RF front-end such as the use of active demodulators, mixers, and amplifiers that require extra power. Therefore, simple modulation techniques such as OOK and ASK presents an opportunity to simplify the WuRx circuitry and to achieve low power consumption. Most of the WuRxs reported are compatible with only one of these two modulations. As a consequence, the WuRx architecture implemented in wireless nodes can limit the interoperability with other transmitters.
1.3. State-of-the-Art Wake-up Radios

Table 1.8 – RF based wake-up radio prototypes.

<table>
<thead>
<tr>
<th>No.</th>
<th>Year</th>
<th>Authors</th>
<th>Device</th>
<th>Source</th>
<th>Address</th>
<th>Channel Mod</th>
<th>Signal Detection</th>
<th>RF Front End</th>
<th>A/D Tech</th>
<th>S.V [V]</th>
<th>D.R [kbps]</th>
<th>Sens [dBm]</th>
<th>R [m]</th>
<th>Pwr [µW]</th>
<th>Implement</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2002</td>
<td>Rabaey et al. [40]</td>
<td>Passive</td>
<td>O-D-B</td>
<td>OKR</td>
<td>ANT, MN, LNA</td>
<td>-</td>
<td>CMOS</td>
<td>1.9</td>
<td>100</td>
<td>-75</td>
<td>10</td>
<td>380</td>
<td>Simulation</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>2002</td>
<td>Gu et al. [74]</td>
<td>Passive</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>OOK</td>
<td>-</td>
<td>0.433</td>
<td>40</td>
<td>-50</td>
<td>8</td>
<td>3</td>
<td>Simulation</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>2004</td>
<td>Pritchard et al. [100]</td>
<td>Passive</td>
<td>O-D-B</td>
<td>OKR</td>
<td>ANT, MN, RAW</td>
<td>PD, E3, VM, LNA</td>
<td>CMOS</td>
<td>0.5</td>
<td>40</td>
<td>-50</td>
<td>8</td>
<td>3</td>
<td>Simulation</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td></td>
<td></td>
<td>Active</td>
<td>O-D-B</td>
<td>OKR</td>
<td>ANT, MN, RAW</td>
<td>-</td>
<td>1</td>
<td>125</td>
<td>40</td>
<td>-50</td>
<td>8</td>
<td>3</td>
<td>Simulation</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>2007</td>
<td>Rabaey et al. [78]</td>
<td>Passive</td>
<td>O-D-B</td>
<td>OKR</td>
<td>ANT, MN, LNA, RAW</td>
<td>E3, VM, LNA</td>
<td>CMOS</td>
<td>1</td>
<td>125</td>
<td>40</td>
<td>-50</td>
<td>8</td>
<td>3</td>
<td>Simulation</td>
</tr>
<tr>
<td>6</td>
<td>2007</td>
<td>Mark et al. [58]</td>
<td>Passive</td>
<td>O-D-B</td>
<td>OKR</td>
<td>AN, MN, RAW</td>
<td>E3, VM, LNA</td>
<td>-</td>
<td>CMOS</td>
<td>1</td>
<td>125</td>
<td>40</td>
<td>-50</td>
<td>8</td>
<td>3</td>
</tr>
<tr>
<td>7</td>
<td>2007</td>
<td>Yu et al. [43]</td>
<td>Passive</td>
<td>O-D-B</td>
<td>OKR</td>
<td>AN, MN, RAW</td>
<td>E3, VM, LNA</td>
<td>CMOS</td>
<td>0.5</td>
<td>100</td>
<td>-75</td>
<td>10</td>
<td>380</td>
<td>Simulation</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>2009</td>
<td>Pritchard et al. [43]</td>
<td>Active</td>
<td>Broadcast</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>CMOS</td>
<td>1.5</td>
<td>2</td>
<td>100</td>
<td>-75</td>
<td>10</td>
<td>380</td>
<td>Simulation</td>
</tr>
<tr>
<td>9</td>
<td>2009</td>
<td>Deo et al. [53]</td>
<td>Passive</td>
<td>O-D-B</td>
<td>OKR</td>
<td>AN, MN, RAW</td>
<td>ED</td>
<td>0.088</td>
<td>0.088</td>
<td>2</td>
<td>100</td>
<td>-75</td>
<td>10</td>
<td>380</td>
<td>Simulation</td>
</tr>
<tr>
<td>10</td>
<td>2009</td>
<td>Taligadi et al. [44]</td>
<td>Passive</td>
<td>O-D-B</td>
<td>OKR</td>
<td>AN, MN, RAW</td>
<td>ED</td>
<td>0.85</td>
<td>0.85</td>
<td>2</td>
<td>100</td>
<td>-75</td>
<td>10</td>
<td>380</td>
<td>Simulation</td>
</tr>
<tr>
<td>11</td>
<td>2009</td>
<td>Duan et al. [50]</td>
<td>Passive</td>
<td>O-D-B</td>
<td>OKR</td>
<td>AN, MN, RAW</td>
<td>ED</td>
<td>0.9</td>
<td>0.9</td>
<td>2</td>
<td>100</td>
<td>-75</td>
<td>10</td>
<td>380</td>
<td>Simulation</td>
</tr>
<tr>
<td>12</td>
<td>2009</td>
<td>Ansari et al. [18]</td>
<td>Semi-Active</td>
<td>O-D-B</td>
<td>OKR</td>
<td>AN, MN, RAW</td>
<td>ED</td>
<td>0.5</td>
<td>0.5</td>
<td>2</td>
<td>100</td>
<td>-75</td>
<td>10</td>
<td>380</td>
<td>Simulation</td>
</tr>
<tr>
<td>13</td>
<td>2009</td>
<td>Durante et al. [28]</td>
<td>Passive</td>
<td>O-D-B</td>
<td>OKR</td>
<td>AN, MN, RAW</td>
<td>ED</td>
<td>0.5</td>
<td>0.5</td>
<td>2</td>
<td>100</td>
<td>-75</td>
<td>10</td>
<td>380</td>
<td>Simulation</td>
</tr>
<tr>
<td>14</td>
<td>2009</td>
<td>Le-Huy et al. [39]</td>
<td>Passive</td>
<td>O-D-B</td>
<td>OKR</td>
<td>AN, MN, RAW</td>
<td>ED</td>
<td>0.5</td>
<td>0.5</td>
<td>2</td>
<td>100</td>
<td>-75</td>
<td>10</td>
<td>380</td>
<td>Simulation</td>
</tr>
<tr>
<td>15</td>
<td>2009</td>
<td>Lagorce et al. [45]</td>
<td>Active</td>
<td>O-D-B</td>
<td>OKR</td>
<td>AN, MN, RAW</td>
<td>AN, MN, RAW</td>
<td>AS</td>
<td>0.868</td>
<td>0.868</td>
<td>2</td>
<td>100</td>
<td>-75</td>
<td>10</td>
<td>380</td>
</tr>
<tr>
<td>16</td>
<td>2010</td>
<td>Gamm et al. [15]</td>
<td>Active</td>
<td>O-D-B</td>
<td>OKR</td>
<td>AN, MN, RAW</td>
<td>ED</td>
<td>0.8</td>
<td>0.8</td>
<td>2</td>
<td>100</td>
<td>-75</td>
<td>10</td>
<td>380</td>
<td>Simulation</td>
</tr>
<tr>
<td>17</td>
<td>2010</td>
<td>Drago et al. [46]</td>
<td>Active</td>
<td>O-D-B</td>
<td>OKR</td>
<td>AN, MN, RAW</td>
<td>ED</td>
<td>0.8</td>
<td>0.8</td>
<td>2</td>
<td>100</td>
<td>-75</td>
<td>10</td>
<td>380</td>
<td>Simulation</td>
</tr>
<tr>
<td>18</td>
<td>2011</td>
<td>Jurdak et al. [71]</td>
<td>Active</td>
<td>O-D-B</td>
<td>OKR</td>
<td>AN, MN, RAW</td>
<td>ED</td>
<td>0.8</td>
<td>0.8</td>
<td>2</td>
<td>100</td>
<td>-75</td>
<td>10</td>
<td>380</td>
<td>Simulation</td>
</tr>
<tr>
<td>19</td>
<td>2011</td>
<td>Huang et al. [96]</td>
<td>Passive</td>
<td>O-D-B</td>
<td>OKR</td>
<td>AN, MN, RAW</td>
<td>ED</td>
<td>0.8</td>
<td>0.8</td>
<td>2</td>
<td>100</td>
<td>-75</td>
<td>10</td>
<td>380</td>
<td>Simulation</td>
</tr>
<tr>
<td>20</td>
<td>2011</td>
<td>Zhang et al. [76]</td>
<td>Passive</td>
<td>O-D-B</td>
<td>OKR</td>
<td>AN, MN, RAW</td>
<td>ED</td>
<td>0.8</td>
<td>0.8</td>
<td>2</td>
<td>100</td>
<td>-75</td>
<td>10</td>
<td>380</td>
<td>Simulation</td>
</tr>
<tr>
<td>21</td>
<td>2011</td>
<td>Hammoud et al. [41]</td>
<td>Passive</td>
<td>O-D-B</td>
<td>OKR</td>
<td>AN, MN, RAW</td>
<td>ED</td>
<td>0.8</td>
<td>0.8</td>
<td>2</td>
<td>100</td>
<td>-75</td>
<td>10</td>
<td>380</td>
<td>Simulation</td>
</tr>
<tr>
<td>22</td>
<td>2012</td>
<td>Ying et al. [49]</td>
<td>Passive</td>
<td>O-D-B</td>
<td>OKR</td>
<td>AN, MN, RAW</td>
<td>ED</td>
<td>0.8</td>
<td>0.8</td>
<td>2</td>
<td>100</td>
<td>-75</td>
<td>10</td>
<td>380</td>
<td>Simulation</td>
</tr>
<tr>
<td>23</td>
<td>2012</td>
<td>Rabaey et al. [78]</td>
<td>Passive</td>
<td>O-D-B</td>
<td>OKR</td>
<td>AN, MN, RAW</td>
<td>ED</td>
<td>0.8</td>
<td>0.8</td>
<td>2</td>
<td>100</td>
<td>-75</td>
<td>10</td>
<td>380</td>
<td>Simulation</td>
</tr>
<tr>
<td>24</td>
<td>2012</td>
<td>Mark et al. [58]</td>
<td>Passive</td>
<td>O-D-B</td>
<td>OKR</td>
<td>AN, MN, RAW</td>
<td>ED</td>
<td>0.8</td>
<td>0.8</td>
<td>2</td>
<td>100</td>
<td>-75</td>
<td>10</td>
<td>380</td>
<td>Simulation</td>
</tr>
<tr>
<td>25</td>
<td>2012</td>
<td>Rabaey et al. [78]</td>
<td>Passive</td>
<td>O-D-B</td>
<td>OKR</td>
<td>AN, MN, RAW</td>
<td>ED</td>
<td>0.8</td>
<td>0.8</td>
<td>2</td>
<td>100</td>
<td>-75</td>
<td>10</td>
<td>380</td>
<td>Simulation</td>
</tr>
<tr>
<td>26</td>
<td>2016</td>
<td>Nikoofard et al. [67]</td>
<td>Passive</td>
<td>O-D-B</td>
<td>OKR</td>
<td>AN, MN, RAW</td>
<td>ED</td>
<td>0.8</td>
<td>0.8</td>
<td>2</td>
<td>100</td>
<td>-75</td>
<td>10</td>
<td>380</td>
<td>Simulation</td>
</tr>
</tbody>
</table>

Note: The table continues with more entries.
<table>
<thead>
<tr>
<th>No.</th>
<th>Year</th>
<th>Authors</th>
<th>P.Src</th>
<th>Address</th>
<th>Channel</th>
<th>Mod</th>
<th>Signal Detection</th>
<th>RX Front End</th>
<th>A.D</th>
<th>Medium</th>
<th>S.V [v]</th>
<th>Freq [kHz]</th>
<th>D.R [kbps]</th>
<th>Sens [dBm]</th>
<th>R [m]</th>
<th>Pwr [µW]</th>
<th>Implement</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2008</td>
<td>Hakkinen et al. [101]</td>
<td>Active</td>
<td>O-O-B</td>
<td>OOK</td>
<td>Photo Diode</td>
<td>LNA, BPF, COM</td>
<td>-</td>
<td>Infared</td>
<td>3</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>6~30</td>
<td>12 Prototype</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>2010</td>
<td>Mathews et al. [29]</td>
<td>Active</td>
<td>O-O-B</td>
<td>OOK</td>
<td>Photo Diode</td>
<td>LNA, C</td>
<td>-</td>
<td>Optical</td>
<td>3.3</td>
<td>-</td>
<td>2</td>
<td>-53</td>
<td>15</td>
<td>317 Prototype</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>2012</td>
<td>Kim et al. [30]</td>
<td>Active</td>
<td>ID-Based</td>
<td>O-O-B</td>
<td>PWM</td>
<td>LED</td>
<td>LED, C</td>
<td>-</td>
<td>MCCJ</td>
<td>Optical</td>
<td>1.2</td>
<td>-</td>
<td>0.991</td>
<td>0.2~50</td>
<td>0.000695 Prototype</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>2012</td>
<td>Sanchez et al. [28]</td>
<td>Active</td>
<td>ID-Based</td>
<td>O-O-B</td>
<td>OOK</td>
<td>Transducer, MN</td>
<td>BPF</td>
<td>AS3933</td>
<td>Sonar</td>
<td>3.3</td>
<td>85</td>
<td>1</td>
<td>-</td>
<td>240</td>
<td>8.1 Prototype</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>2013</td>
<td>Yadav et al. [25]</td>
<td>Active</td>
<td>O-O-B</td>
<td>OOK</td>
<td>Piezoelectric, MN</td>
<td>LNA, M-IF BB</td>
<td>-</td>
<td>Ultrasonic</td>
<td>0.6</td>
<td>40.6</td>
<td>0.25</td>
<td>-</td>
<td>8.6</td>
<td>4.78 Prototype</td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>2013</td>
<td>Lantanz et al. [26]</td>
<td>Active</td>
<td>ID-Based</td>
<td>O-O-B</td>
<td>OOK</td>
<td>Piezoelectric, MN</td>
<td>LNA, C</td>
<td>MCCJ</td>
<td>Ultrasonic</td>
<td>2</td>
<td>40</td>
<td>0.016</td>
<td>-10</td>
<td>10</td>
<td>1.748 Prototype</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>2014</td>
<td>Hoflinger et al. [27]</td>
<td>Active</td>
<td>O-O-B</td>
<td>OOK</td>
<td>Microphone</td>
<td>LNA</td>
<td>AS3933</td>
<td>Audio</td>
<td>3</td>
<td>18</td>
<td>-</td>
<td>-</td>
<td>7.5</td>
<td>5.6 Prototype</td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>2016</td>
<td>Bannoura et al. [102]</td>
<td>Active</td>
<td>ID-Based</td>
<td>O-O-B</td>
<td>ASK</td>
<td>Microphone</td>
<td>BPF, LNA</td>
<td>AS3934</td>
<td>Audio</td>
<td>3</td>
<td>20</td>
<td>-</td>
<td>-</td>
<td>10</td>
<td>45 Prototype</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>2016</td>
<td>Carrascal et al. [103]</td>
<td>Active</td>
<td>ID-Based</td>
<td>O-O-B</td>
<td>ASK</td>
<td>Solar panel</td>
<td>EDL, AS3933</td>
<td>VLC</td>
<td>2.4</td>
<td>21</td>
<td>1.12</td>
<td>-</td>
<td>7~14</td>
<td>19.2 Prototype</td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>2016</td>
<td>Lim et al. [104]</td>
<td>Active</td>
<td>-</td>
<td>O-O-B</td>
<td>OOK</td>
<td>Photo Diode</td>
<td>PD, COM</td>
<td>-</td>
<td>Optical</td>
<td>-</td>
<td>-</td>
<td>250</td>
<td>-</td>
<td>25</td>
<td>28.1 Prototype</td>
<td></td>
</tr>
</tbody>
</table>

**Key:**
- **P.Src:** Power Source
- **Mod:** Modulation Technique
- **RX Front End:** Receiver Front End
- **A.D:** Address Decoding Capabilities
- **Tech:** Technology Used
- **S.V:** Supply Voltage
- **Freq:** Frequency
- **D.R:** Data Rate
- **Sens:** Sensitivity
- **R:** Operational Range
- **Pwr:** Power Consumption in Active Mode
- **Implement:** Implementation

**Note:** Articles that did not provide values for particular information has been stated as (-) in the Tables.
1.4 Statistical Analysis

Different components of the WuR design impact its final performance and add to its overall power consumption. In this section, we compare different RF based WuR prototypes designed and tested in terms of power consumption, sensitivity, data rate, communication range and the modulation scheme used, regardless of their specific technology. The numbers presented in this section are the actual numbers reported by the authors of each article. This statistical comparison will then be used as a guideline to recommend which prototypes are suitable and meet the requirements of various applications outlined in Section 1.7.

1.4.1 Modulation Schemes

The main goal of incorporating WuR with typical sensor node is to reduce power consumption. In order to achieve this, the WuR design should be of low power, hence, the modulation complexity should be kept low as well. The higher the modulation complexity, the more stringent requirements for receiver and transmitter in terms of circuit complexity and power.

When comparing this with the state-of-the-art low power WuR summarized in Table 1.8, it can be noted that most designs use either envelope detector based On-Off keying (OOK) or non-coherent Frequency-Shift-Keying (FSK). To curb energy consumption by simplifying overall implementation, the designers of the WuR generally favor architectures utilizing OOK modulation schemes. For instance, a simple envelope detector using few diodes and capacitors can be used for signal detection [85, 35, 21]. It is evident from Table 1.8 that most of the concepts that have power consumption below 10 $\mu$W are using OOK modulation.

In contrast, the nonlinear nature of envelope detectors make the OOK receivers more susceptible to interference contributing to higher packet error rate and need for retransmission. One can argue that retransmission is expensive in terms of power, but the burden of this is shifted from high power radio to ultra-low power WuR. The advantage of FSK over OOK is that it is more resilient to fading and interference. Therefore, in view of low power WuRx design, either OOK or FSK modulation scheme should be considered.

There are five reported design concepts that differ from above. The concept presented by Le-Huy et al. [10] uses Pulse-width modulation (PWM) technique since it only requires an integrator with a reset option without increasing the complexity of the receiver architecture. Another benefit of using PWM is that it presents the possibility to control the duty cycle of the transceiver. Shuangming et al. [99] use the Offset quadrature phase-shift keying (O-QPSK) to design an ultra low power System-on-Chip (SoC) based baseband processor with wake-up identification receiver consuming only 28.2 $\mu$W. The concept by Ansari et al. [18] use multi-stage approach for WuSing where CC1000 radio chip is used to perform OOK by turning on and off it’s power amplifier. Then the digital data is encoded using Pulse Interval Encoding (PIE) with different time intervals $T$. In order to successfully decode this data sequence, authors utilize MSP430 series micro-controller. A broadband-IF super heterodyne proposal for a crystal-less 2.4 GHz WuRx is presented by Drago et al. [46]. The WuS is modulated by means
Chapter 1. Ultra Low Power Wake-Up Radios:  
A Hardware and Networking Survey

1.4.2 Sensitivity vs. Power Consumption

Fig. 1.11 shows the comparison between the WuR’s power versus sensitivity. It should be noted that these are all custom ultra-low power radios, including radios of different architecture, different data rate, different operating frequencies; none of which is separated in this plot.

Generally, the power consumption of the WuR is related to its sensitivity. With power consumption, in µW, on the y-axis and the sensitivity, in dBm, on the x-axis, two distinct trends can be observed. First, when looking at sensitivity higher than -40 dBm (to the left on the x-axis) it can be seen that there is no direct correlation between the changing sensitivity to the power of the receiver. However, there is a floor around 2 µW suggesting that there is a minimum power requirement for the radio regardless of sensitivity. With increasing sensitivity from -40 dBm (to the right on the x-axis) there is a linear trend indicating a correlation between sensitivity and power. It can be seen empirically through slope-fitting that a 20 dBm change in sensitivity results in an approximately 10× change in power consumption. The designs

of Pulse Position Modulation (PPM). In order to reduce the power consumption of their design, both the signal front-end and the oscillator are duty-cycled at the pulse level. The WuRx achieves -82 dBm sensitivity and requires up to 415 µW. Recently, Roberts et al. [77] have proposed a Bluetooth Low Energy (BLE) WuRx with energy harvesting capability. They have utilized Code division multiple access (CDMA) modulation scheme referred to as Back-channel for encoding and decoding the WuS. Upon signal detection, the information is fed into a baseband processor that correlates the energy levels with a time-based template that matches the sequence of BLE advertising packets to determine the presence of a wake-up message. This CMOS based design was able to achieve sensitivity of -56.5 dBm while consuming only 236 nW.
1.4. Statistical Analysis

below this slope are regarded as energy efficient as most of them exhibit high sensitivity at low energy cost.

Moreover, as can be seen in Fig. 1.11, the lowest power consumption that has been achieved so far has been 98 nW [92], but not without trading-off the sensitivity (-41 dBm). This design was able to achieve a communication range of only 1.2 m. Out of 75 prototypes that we have surveyed for RF based WuR for those that power consumption and sensitivity values were provided, only 23 prototypes were able to achieve power consumption below 10 µW, where [92, 67] and [97] reached an outstanding power consumption around 100 nW.

Regarding the requirements for different applications in Table 1.12, it can be seen that for short-range communication such as WBAN, five WuR prototypes [92, 97, 85, 79, 77] (marked with green circles) fulfill the power consumption and sensitivity requirements. All these prototypes have power consumption below 0.27 µW with sensitivity ranging between -40 dBm to -56 dBm. For mid-range communication (e.g., smart city and metering), only [55, 63] (marked with a red circle) fulfill all these requirements at the same time. Power and sensitivity of these prototypes are 4.75 µW and 7.25 µW, and -83 dBm and -80 dBm, respectively.

For ultra-low power WuR, the knowledge from Fig. 1.11 is useful for understanding key design trade-offs. For example, most designers [68, 70, 45] try to push the sensitivity as low as possible to achieve better communication range, but this may lead to power-costly design.

In terms of modulation technique, most of these designs utilize OOK modulation. OOK based prototypes have been able to reach the two extreme ends of the power levels, one being the most energy efficient [92] while the other design is not [68]. There are two designs, one based on CDMA [77] and the other using FSK modulation [59] that have also been able to achieve an excellent receiver sensitivity of -56.5 dBm and -87 dBm, respectively with very low power requirements. Both of these prototypes are fabricated using 65nm CMOS process and use correlators for address decoding.

1.4.3 Data Rate vs. Power Consumption

Fig. 1.12 shows the data rate of WuRxS with respect to their power consumption and signal modulation techniques. Since, power is inversely proportional to data rate, it is generally possible to increase the data rate with little power overhead [105], however, communication distance will be short. For example, it does not cost much in terms of power to increase the modulation rate from 1 kbps [31] to 100 kbps [64] in an OOK receiver.

As can be seen, there are fourteen designs that have been able to reach a data rate above 200 kbps. Out of these, five [15, 47, 53, 54, 80] have a power consumption below 10 µW.

From the application perspective, there are few designs [15, 47, 53, 54, 80] (circled in red) that offer high data rate at the same time consuming low power making them suitable for WBAN application scenarios for replacing the high data radio with WuR. Thanks to its high data rate and low power consumption, these WuR utilized as main data radio can have an advantage
Chapter 1. Ultra Low Power Wake-Up Radios: 
A Hardware and Networking Survey

1.4.4 Range and Frequency Usage

So far we have only looked at the modulation technique, receiver sensitivity, and data rate. Another factor that impacts the power consumption of wake up radios is the carrier frequency. The choice of the carrier frequency is an important parameter for the wake-up transceiver. Fig. 1.13 shows the main frequency bands that have been utilized by most of the WuR prototypes together with the min, max, and average power consumption. One of the trends that can be observed is that the average power consumption of transceivers increases from sub-GHz band to 2.4 GHz. This is due to the fact that transceiver circuits running at higher frequencies require more current to achieve the same performance as lower frequencies.

From this survey and referring to Table 1.8, it can be seen that 25 of the prototypes are based on 2.4 GHz while 32 of them are between 433 MHz and 915 MHz. One of the designs that have achieved an outstanding power consumption of 0.0115 $\mu$W operates in 50 MHz [67]. The design is based on CMOS technology and features a data rate of 1 kbps with receiver sensitivity.
of -60dBm. Due to its semi-active design and OOK modulation, this particular prototype managed to surpass state-of-the-art wake-up radios in terms of sensitivity and power trade-off. However, it has only been tested via simulations. Nevertheless, most of the designers have opted to shift from high frequency to sub-GHz as an operating frequency for wake-up receivers. One of the reasons is that at higher frequencies the attenuation rate also increases, i.e., the 2.4 GHz signal weakens faster than a sub-GHz signal. According to the Friis equation, the path loss at 2.4 GHz is 8.5 dB higher than at 900 MHz translating into 2.67 times longer range for 900 MHz transceivers.

Besides the need for higher power for the same link budget, 2.4 GHz band is more prone to interference due to spectrum crunch and devices such as Wi-Fi and Bluetooth operating in the same band. Sub-GHz ISM bands are mostly used for proprietary low-duty-cycle links and are not as likely to interfere with each other. The quieter spectrum means easier transmissions and fewer retries, which is more efficient and saves battery power for wake-up radio based systems.

Furthermore, Fig. 1.14 shows the maximum achievable communication range reported for different WuR prototypes in terms of their power consumption. It should be noted that we do not take into account prototypes that did not report explicitly the communication range of the WuR. From the application point of view, WuR prototypes with communication range between 30 m to 50 m (labeled as cluster A) [17, 15, 34, 35, 21] satisfy the requirements for mid-range applications. For the WBAN case WuR concepts [15, 47, 53, 54, 80] fulfill the sensitivity, data rate and power requirements, if used as a full data radio. However, if utilized just as a secondary radio for triggering the main node’s transceiver, WuR with power consumption below 10 µW should be considered.
Chapter 1. Ultra Low Power Wake-Up Radios: A Hardware and Networking Survey

1.4.5 Summary

The main characteristics of all ultra-low power WuR are sensitivity, data rate, frequency, and power consumption. However, the technology used to design WuR prototypes vary from simple energy detection using discrete components to envelope detection using CMOS, influencing its overall performance. Therefore, for different application requirements the best prototype has to be selected carefully. While some provide high data rate, others are better for high sensitivity or very low power consumption.

It has been observed that to achieve ultra-low-power consumption while maintaining robust operation involves difficult trade-offs between range, data rate, sensitivity, and energy efficiency that must be overcome through a combination of innovative circuit design, novel architectures, and system-level considerations. This section has provided some benchmarking data to help identify what architectures and WuR prototypes might make the most sense given system-level specifications. While optimal implementations depend strongly on the given application, in general the most energy efficient WuR employ low-complexity modulation schemes (e.g., OOK).

1.5 Medium Access Control

Major work on WuR technology has been focused on improving hardware components to achieve better power consumption and physical layer communication characteristics. Nevertheless, to fully exploit the technology, it must be coupled with communication protocols, rounding out the system design. We divide our discussion in two parts, first focusing on medium access in this section, then moving up the protocol stack to routing in the next section. In considering MAC, we address properties both general to wireless medium access and specific to WuR. Table 1.10 summarizes the different WuR based MAC protocols designed so
far while Fig. 1.15 organizes them into a taxonomy.

### 1.5.1 Classification of WuR-based Medium Access

In the last decade, various MAC protocols have been proposed for wireless sensor networks. Most of these energy conservation protocols [106, 107, 5] are single-radio based and use duty cycling mechanisms. In duty cycling mode the nodes periodically wake-up to sample the channel and then go back to the sleep state. However, duty cycling MACs suffer from idle listening and waiting time that increases the data latency and power consumption. Wake-up radios combat this at the hardware level, but they must also be coupled with a MAC protocol to control their use.

The main contrast between traditional asynchronous MAC protocols and MACs designed for use with WuRs is that dual-radios are utilized, one of which is the extreme low-consumption
Chapter 1. Ultra Low Power Wake-Up Radios:  
A Hardware and Networking Survey

WuR. In the former, different power management techniques are applied to the main transceiver for reducing radio-on times. The latter uses different strategies to control the secondary radio while keeping the main radio off during periods of inactivity.

MAC protocols typically divide themselves between on-demand and scheduled, with a majority of existing WuR protocols falling into the former category for flexibility and simplicity as complex, system wide schedules are not required. Further, an on-demand approach well-suits the use of the WuR as a trigger, and avoids heavy resource requirements to build, communicate, and store schedules. Below we focus on several dimensions to on-demand communication, discussing how the WuR paradigm changes their applicability w.r.t. standard wireless communication. Fig.1.16 (notably not drawn to scale) illustrates different WuR based communication schemes that can be adopted for various applications. Two channels are utilized, the WuR channel and the main radio channel. The height of the bar symbolically represents the power consumption of the respective transceivers (WuR and the main radio) in active and inactive states during different radio events while the width represents the radio on-time.

The first concern we address in the taxonomy of Fig. 1.15 requires identifying which pair of nodes is allocated the wireless channel based on who is the communication initiator: the transmitter, the receiver or either (bi-directional).

(i) **Initiator: Transmitter.** In a Transmitter-initiated protocol, the node that has data to send initiates communication (Fig. 1.16(a)). It first sends a wake-up signal, whose receipt triggers the receiver to wake up its main transceiver. Data is exchanged using the main transceivers followed by Tx-ACK if transmission was successful. The nodes then go back into sleep mode.

(ii) **Initiator: Receiver.** In Receiver-initiated systems (Fig. 1.16(b)), the burden of starting a communication event falls to the receiver, specifically with the node, often the sink, announcing its readiness to receive data. After this announcement, it switches to receive (RX) mode and monitors the wireless channel to receive any incoming packets. If we assume the WuRx on the sender side is always active and listening, when it receives the signal it activates its main transceiver to send the data packet. The session ends when the transmit acknowledgment (Tx-ACK) signal arrives at the sender from the destination node, after correctly receiving the data packet. All the nodes then go back to sleep mode. This communication modality is most effective when transmissions are infrequent, and collisions at the receiver are unlikely.

(iii) **Initiator: Bi-directional.** In bi-directional systems, either of the nodes that want to push or pull data can initiate the communication via their respective WuRs. The data packet is still exchanged between main transceivers. This setup is more suitable for enabling multi-hop communication.

Thus far we have ignored the placement of the specialized WuR hardware, assuming that the non-initiator is equipped with the WuRx. Here we detail asymmetric and symmetric options.
1.5. Medium Access Control

(i) **Hardware: Asymmetric.** If only a single hop network is required, an asymmetric scheme is possible, with the WuRx on only one side of the communication link. In a scenario with a powered sink, a Receiver-Initiated solution can be used to pull data to the sink from nodes that are one-hop from the sink. The non-sink nodes must have a WuRx, allowing them to wait in a very low consumption state, then switching to a higher consumption only when the sink is ready to receive their data.

(ii) **Hardware: Symmetric.** For a multi-hop system, each node must alternately serve as receiver and transmitter, resulting in a symmetric system in which all nodes are equipped with a wake-up transceiver. Either receiver- or transmitter-initiated schemes are possible. Fig. 1.16(c) shows a transmitter-initiated case, in which the transmitter sends a wake-up signal to the receiver. The receipt of this signal triggers the activation of the main transceivers for data exchange.

Next we turn to the usage of the wake-up radio itself, concentrating on how and when it is powered. There are three power management techniques that can be applied: always-ON, duty cycling the WuR or energy harvesting.

(i) **Power: Always-On WuR.** Typically, due to the low consumption of the WuRx technology, it can be constantly powered, waiting for a trigger signal. In a transmitter-initiated scenario, this minimizes the latency, as the receiver is immediately aware of the transmitter’s need to initiate communication.

(ii) **Power: Duty Cycled WuR.** To further reduce power consumption, the wake-up radio itself can be duty cycled (Fig. 1.16(d)), meaning the WuRx is periodically put into listen mode to monitor the channel for a wake-up signal. To compensate for the sleeping times of the receiver, the WuTx must send the wake-up signals more than once, until a wake-up acknowledgment (Wu-ACK) is received from the target WuRx. When the WuRx listening period coincides with the wake-up signal transmission, the receiving node switches on its main transmitter and the main data transmission is initiated. If no Wu-ACK is received, the initiator node can re-transmit the wake-up signal. To avoid overhearing by the non-targeted nodes, the wake-up signal carries the destination address.

(iii) **Power: Energy Harvesting WuR.** As mentioned in Section 1.2, in energy harvesting WuR system (EH-WuR), the WuRx is only woken up when “sufficient” energy is harvested from the wake-up signal. Fig. 1.16(e) illustrates the transmitter-initiated scenario where the energy from the WuS is utilized for powering up the trigger circuitry. In this scenario when there is no communication going on, the WuRx is completely switched OFF.

We next consider two, elements that we leave out of our taxonomy, but are nevertheless considered part of the MAC. First, what information is exchanged over the WuR and second, whether the WuR works in the same frequency band as the main radio.
Chapter 1. Ultra Low Power Wake-Up Radios:  
A Hardware and Networking Survey

Figure 1.16 – Various wake-up radio communication schemes.

(a) Transmitter-Initiated MAC.

(b) Receiver-Initiated MAC.

(c) Symmetric WuR System.

(d) Duty Cycled WuR.

(e) Energy Harvesting WuR System.
1.5. Medium Access Control

(i) **Data: Trigger-only.** The most typical use of the WuR is to trigger a higher power radio, used for communicating data. This requires very little logic on the WuR board, and minimizes hardware complexity. As mentioned previously, the trigger can be broadcast, waking up all neighboring nodes, or unicast, with the trigger containing the address of the intended recipient.

(ii) **Data: WuR as main data radio.** As an alternate, the low-power WuR can be responsible for all communication i.e, for sending the wake-up signal and the data packet. The communication is still bidirectional, however, there is no main high power transceiver.

For the next option, we look at the radio itself, specifically the use of the wireless spectrum, divided into channels.

(i) **Spectrum: In-Band.** Few published MAC protocols address only in-band (single channel) communication i.e, both the trigger and the data are exchanged over the same channel or frequency.

(ii) **Spectrum: Out-of-Band.** Multiple channels, instead, can reduce interference and increase bandwidth, but at the expense of additional coordination between senders and receivers both in time, as mentioned previously, and also across the space of the channels. In most of the WuR-MAC protocols, the bandwidth is divided into two channels: one used for control and the other for wake-up signals. Another is the data channel with higher bandwidth allocated for the main radio. For channel reservation, normally RTS/CTS handshake mechanism is performed over the control channel. The RTS/CTS frame includes a preamble, sender/receiver address, channel information for the main transceiver, and packet length. Use of out-of-band approach has following advantages. Firstly, using different channels appropriately can lead to higher throughput. Secondly, communication on different channels or frequency does not interfere with each other allowing multiple transmissions simultaneously, leading to fewer collisions.

In the remainder of this section, we organize our discussion of proposed protocols along the taxonomy of Fig. 1.15, first according to the communication initiator: bi-directional, receiver-initiated, and transmitter-initiated. Within each, we further sub-divide the discussion across symmetric and asymmetric hardware and different power management approaches, also offering the categorization of the protocols along the lines mentioned here.

### 1.5.2 Bi-directional MAC Protocols

The most populated sector for MAC protocols is **bi-directional,** in which any node can initiate the communication. For instance, in a WBAN the traffic is normally categorized into two types: **uplink** where the sensing nodes can communicate with the coordinator node to report urgent data and the **downlink** where the coordinator can send messages to the nodes. In this
### Table 1.10 – Wake-up radio based MAC protocol designs.

<table>
<thead>
<tr>
<th>No.</th>
<th>Protocol</th>
<th>Year</th>
<th>Initiator</th>
<th>Hardware</th>
<th>Power Management</th>
<th>Information Exchange</th>
<th>Channels</th>
<th>Key Novelty</th>
<th>Implement</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Guo et al. [108]</td>
<td>2001</td>
<td>Bidirectional</td>
<td>Symmetric</td>
<td>Always ON</td>
<td>Trigger</td>
<td>Multiple</td>
<td>- Embedding channel information in WuS</td>
<td>Simulation</td>
</tr>
<tr>
<td>2</td>
<td>STEM-T [109]</td>
<td>2002</td>
<td>Transmitter</td>
<td>Symmetric</td>
<td>Duty Cycled</td>
<td>Trigger</td>
<td>Out-of-Band</td>
<td>- All neighbors woken up</td>
<td>Simulation</td>
</tr>
<tr>
<td>4</td>
<td>PTW [110]</td>
<td>2004</td>
<td>Transmitter</td>
<td>Asymmetric</td>
<td>Duty Cycled</td>
<td>Trigger</td>
<td>Out-of-Band</td>
<td>- Broadcast wake-up</td>
<td>Simulation</td>
</tr>
<tr>
<td>5</td>
<td>Miller et al. [111]</td>
<td>2005</td>
<td>Bidirectional</td>
<td>Symmetric</td>
<td>Duty Cycled</td>
<td>Trigger</td>
<td>Multiple</td>
<td>- Wake up scheduling</td>
<td>Simulation</td>
</tr>
<tr>
<td>6</td>
<td>SLAM [112]</td>
<td>2007</td>
<td>Bidirectional</td>
<td>Symmetric</td>
<td>Energy harvesting</td>
<td>Trigger</td>
<td>Multiple</td>
<td>- Energy harvesting by all nodes</td>
<td>Simulation</td>
</tr>
<tr>
<td>7</td>
<td>WUR-MAC [113]</td>
<td>2009</td>
<td>Transmitter</td>
<td>Asymmetric</td>
<td>Always ON</td>
<td>Trigger</td>
<td>Out-of-Band</td>
<td>- CTS / RTS on Wu channel</td>
<td>Simulation</td>
</tr>
<tr>
<td>8</td>
<td>DCW-MAC [114, 115]</td>
<td>2011-14</td>
<td>Transmitter</td>
<td>Asymmetric</td>
<td>Duty Cycled</td>
<td>Trigger</td>
<td>In-Band</td>
<td>- Single transmitter for trigger and data</td>
<td>Simulation</td>
</tr>
<tr>
<td>9</td>
<td>VLPM [116]</td>
<td>2011</td>
<td>Bidirectional</td>
<td>Symmetric</td>
<td>Always ON</td>
<td>Trigger</td>
<td>Multiple</td>
<td>- Bidirectional wake up</td>
<td>Simulation</td>
</tr>
<tr>
<td>10</td>
<td>On-Demand MAC [117, 118]</td>
<td>2011</td>
<td>Bidirectional</td>
<td>Symmetric</td>
<td>Always ON</td>
<td>Trigger</td>
<td>Multiple</td>
<td>- Bidirectional wake-up</td>
<td>Simulation</td>
</tr>
<tr>
<td>11</td>
<td>Blanckenstein et al. [119]</td>
<td>2012</td>
<td>Transmitter</td>
<td>Asymmetric</td>
<td>Always ON</td>
<td>Trigger</td>
<td>In-Band</td>
<td>- Node clustering - TDMA on main radio</td>
<td>Simulation</td>
</tr>
<tr>
<td>12</td>
<td>WMAC [23]</td>
<td>2012</td>
<td>Bidirectional</td>
<td>Symmetric</td>
<td>Always ON</td>
<td>Trigger</td>
<td>Multiple</td>
<td>- TDMA on main radio</td>
<td>Simulation</td>
</tr>
<tr>
<td>13</td>
<td>WUR-TICER [120]</td>
<td>2013</td>
<td>Transmitter</td>
<td>Asymmetric</td>
<td>Energy harvesting</td>
<td>Trigger</td>
<td>In-Band</td>
<td>- Energy harvesting by all nodes</td>
<td>Simulation</td>
</tr>
<tr>
<td>14</td>
<td>GWR-MAC [121, 122]</td>
<td>2014</td>
<td>Bidirectional</td>
<td>Symmetric</td>
<td>Always ON</td>
<td>Trigger</td>
<td>Multiple</td>
<td>- Bidirectional wake up</td>
<td>Simulation</td>
</tr>
<tr>
<td>16</td>
<td>DoRa [123]</td>
<td>2015</td>
<td>Receiver</td>
<td>Asymmetric</td>
<td>Energy harvesting</td>
<td>Trigger</td>
<td>Out-of-Band</td>
<td>- Energy harvesting - Base station wakes up the neighbors</td>
<td>Simulation</td>
</tr>
<tr>
<td>17</td>
<td>AWD-MAC [124]</td>
<td>2015</td>
<td>Receiver</td>
<td>Asymmetric</td>
<td>Always ON</td>
<td>Set of Triggers</td>
<td>In-Band</td>
<td>- Wake up multiple neighbors</td>
<td>Simulation</td>
</tr>
<tr>
<td>18</td>
<td>BATS [125]</td>
<td>2016</td>
<td>Receiver</td>
<td>Asymmetric</td>
<td>Duty Cycled</td>
<td>Trigger</td>
<td>Out-of-Band</td>
<td>- Supports Mobility</td>
<td>Testbed</td>
</tr>
<tr>
<td>19</td>
<td>W-MAC [126]</td>
<td>2017</td>
<td>Bidirectional</td>
<td>Symmetric</td>
<td>Always ON</td>
<td>Trigger</td>
<td>Out-of-Band</td>
<td>- Bidirectional wake-up - Addressed beacon - Supports multi hop</td>
<td>Simulation</td>
</tr>
</tbody>
</table>

### Framework

In this framework, all the nodes can be attached with WuR transceivers providing bi-directional communication [126]. This requires symmetric hardware on all nodes, but affords full flexibility of power management, which we detail here.

1. **Always ON**: The MAC protocols in this category keep the low-power WuRx always ON. As such, it is able to receive the wake-up beacon immediately with reduced latency, however, the energy consumed is non-negligible.

Several existing MAC protocols, VLPM [116], WhMAC [85, 23], On-Demand MAC [117, 118], and GWR-MAC [121, 122], have been proposed for the star topology, applying this schema using existing wakeup radios to WBAN. The authors assume that the wake-up beacon contains the target destination node address allowing other nodes in the network to keep their main radio in sleep state. However, all of these works ignore the fact that different physiological parameters sampled by different sensor nodes generally have significant differences in terms of traffic arrival and data rate. For instance, sensors

---

54
monitoring electrocardiography (ECG) is allocated high data rate while body temperature sensors are assigned low data rate. If the same energy saving strategy is used to cope with all of the sensor nodes, the nodes with high energy consumption rate will quickly exhaust their energy, which eventually reduce the entire network lifetime. In addition, while some of these protocols may work well in a small, single-hop network like a WBAN, they may lack in flexibility to work for more general WSNs with a large number of nodes.

Guo et al. [108] proposed one of the earliest protocols using always-on WuRxs to show the benefit of bi-directional over traditional radios with duty cycling MAC. The receiver assigns the nodes with unique data channels by encoding channel information in the wake-up beacon called *channel based local addressing scheme*. The transmitting node captures this information via its WuRx and switches its data radio to receiver's channel after activating the main node. Through the simulation of their protocol in broadcast mode, the authors showed that power reduction of $10^{-3}$ to $10^{-100}$ times can be achieved with always-on WuRxs compared to duty cycled main radio solutions.

To target real WSN applications, **W-MAC** [126] was proposed for multi-hop network in which nodes alternately act as senders and receivers. W-MAC takes advantage of secondary always-on WuR that is attached to the main mote acting as the communication initiator. Whenever a node has data to send, either generated by the upper layers of the protocol stack or forwarded by neighboring nodes, W-MAC first transmits a wake-up beacon containing the destination node address. To avoid collisions, the WuR and the main radio use different channels for wake-up beacon and data packets. Using simulations with two different routing protocols, W-MAC illustrated that WuR technology has the potential to offer significant energy savings without compromising on reliability and latency.

2. **Duty cycled:** Another bi-directional communication is proposed by Miller et.al in [111]. To avoid costly full wake-ups, the sensor nodes schedule a triggered wake-up with a receiver. This schedule is calculated by the sink node based on the previous traffic patterns and is then disseminated to the network. Each node in the network knows their next wake-up time and when there is nothing to receive, the WuR is switched into duty cycling mode until the next wake-up cycle. The proposed idea is compared to STEM [109] and the simulations show significant reduction in the delivery latency. Nevertheless, schedule sharing requires tight synchronization at the receiver side leading to extra energy overhead to overcome clock drifts. The authors also assume that all the nodes share the same wake-up channel without specific node addressing, thus triggering all the nodes.

3. **Energy harvesting:** **MH-REACH** is a MAC protocol designed for passive RFID-based WuR systems supporting multi-hop wake-up sensor networks [81]. In it, the WuTx on the sink wakes up all nodes in its vicinity. Any node that was woken up offloads its data to the sink, and, if it is a multi-hop node, it also transmits a wake-up signal to wake up
other nodes within its transmission range. If it is an edge node, after transmitting its data to the sink, it returns to the sleep state until the next wake-up event. Although this protocol supports a multi-hop network, the passive devices require wake-up signals of longer duration (between 5s-10s) to accumulate enough energy to fully power-up the circuitry. Therefore, applications must trade-off maximum wake-up range and node lifetime. In addition, due to its broadcast nature of the WuS, all the nodes within 1-hop are activated, thus contributing to overhearing overhead.

A similar energy harvesting based MAC protocol (SLAM) has been proposed in [112]. In SLAM, a few nodes are assigned as guard nodes that monitor the traffic between hops to detect malicious nodes. During periods of inactivity the guard nodes are put into sleep mode and switched on when required via passive WuRxs. Through experiments authors have shown that listening energy can be reduced by to 30-129 times using WuRs while providing a high level of network security.

1.5.3 Receiver-Initiated MAC Protocols

To increase throughput and to shift the burden of energy consumption from the sender to receiver, some authors have proposed receiver-initiated WuR-MAC protocols. Their design is inherently asymmetric, and the full range of power management techniques are applicable.

1. **Always ON:** To extend the life of sensing nodes, AWD-MAC [124] utilizes the receiver-initiated scheme but employs a single channel for communication. Different from the traditional receiver-initiated cycled receiver (RICER) where only one common broadcast beacon is sent, AWD-MAC first sends a set of wake-up beacons in sequence to wake-up multiple neighbors for neighbor discovery. The nodes then reply using random slots with their node IDs and respective data rates. Subsequently, the coordinator node creates a neighbor table to query each node in an asynchronous fashion. AWD-MAC claims that the collisions are removed as only one transmitter node is allowed to send its data at a given time while sharing the same channel. Nonetheless, collisions do occur during the neighbor discovery phase when AWD-MAC sends the broadcast beacon to detect new nodes.

2. **Duty cycled:** The first mobility-based WuRx system using the receiver-initiated paradigm has been proposed in the BATS project [127]. The authors have investigated the potential of ultra-low power WuRs carried by bats to monitor encounters between individuals and to track their routes at high spatial and temporal resolution [128, 129, 125]. Due to limited available energy, the wake-up receivers are duty cycled. To support multiple mobile nodes and to prevent the collisions at the receiver side, the ground node uses Time Division Multiple Access (TDMA)-like communication slots with guard intervals between slots. The communication between the mobile nodes is not synchronized. When the mobile node enters the communication range of the ground node, the latter sends a wake-up beacon. Upon successful wakeup, the mobile node offloads the data.
1.5. Medium Access Control

within its assigned slot. Due to the high mobility of the bat nodes, no carrier sensing techniques are performed prior to transmission allowing mobile nodes to send data before exiting the transmission range. Therefore, if multiple mobile nodes are within the receivers vicinity, data collisions may occur and the packets can be lost.

3. **Energy harvesting: DoRa** [123] offers a WuR-MAC protocol that builds upon the foundation of the receiver-initiated paradigm for the realization of energy harvesting in one hop networks. In the proposed mechanism, no channel reservation or packet acknowledgments are transmitted. The nodes answer to the base station by directly sending the data packet. DoRa also provides out-of-band support and node addressing. However, similar to MH-REACH, a strong wake-up signal is required in order to harvest enough energy to activate the nodes leading to high data latency.

1.5.4 Transmitter-Initiated MAC Protocols

We next consider **transmitter-initiated** MAC protocols where each node chooses its transmission schedule autonomously. In general, this approach puts the energy consumption burden for transmission on the sender, with a much lighter load on the receiver. Both asymmetric and symmetric approaches are possible, and multiple power management techniques have been applied.

We begin with asymmetric:

1. **Always ON:** A transmitter-initiated MAC protocol leveraging always-on WuRx is proposed by Mahlknecht et al. [113]. WUR-MAC is based on multi-channel principle and uses RTS and CTS handshake mechanism. The sender node first transmits the request-to-send packet for selecting the appropriate receiver. The intended node then replies with clear-to-send packet and triggers its main radio for data reception at higher bandwidth. WUR-MAC supports both point-to-point and broadcast communication. Using channel reservation reduces collisions but may impact on the data latency as the transmission is blocked until CTS is successfully exchanged.

Energy efficient node clustering using WuRx for WBAN sensors with similar readings is presented in [119]. To eliminate idle listening and channel contention, an always-on WuRx is attached to a main radio that utilizes TDMA scheme. To achieve clustering, the relevant data information is encoded in the WuTx’s data pattern. The idea is to reduce energy consumption by reducing the number of data packets through clustering nodes with similar sensor readings and allowing only the cluster head to forward data to the sink. This protocol is only tested using simulations where the wake-up addressing mechanism is used to trigger nodes according to the data they have sensed.

2. **Duty cycled:** Similar to STEM-T, Yang et al. [110] propose a Pipelined Tone Wakeup (PTW) scheme that uses two different radio channels, one for data and one for tone detection. In PTW, the WuRx is duty cycled. When a node has packets to send, it
transmits a tone on the wakeup channel and sends the notification packet on the data channel to specify the target node. As the wake-up tone is broadcast, any node within the transmission range of sender will be awakened. From the point of view of application scenarios for opportunistic networking, such an approach could grant fast wake-up in dense and multi-hop scenarios while reducing end-to-end latency, but could be less energy efficient.

Analogous to STEM and PTW, the work in [114, 115] also duty cycles the WuRx statically, but uses in-band approach for communication. In DCW-MAC, the main radio is used for both sending the wake-up beacon and the data, but the authors add dedicated, secondary low-power radio, acting as a WuRx, operating in the same frequency band. The authors through analytical models derive the optimal sleep and listen time for a duty cycled WuRx and compare these models to a non-WuR based system. However, the analysis assumes perfect detection of wake-up signals and energy consumed due to collisions is ignored in the derivation of optimal timing. In addition, the main radio also acts as a wake-up transmitter, hence, frequent switching between RX and TX mode may result in extra energy consumption.

3. Energy harvesting: Le et al. [120] have proposed the WUR-TICER MAC protocol that operates by harvesting energy from the ambient environment. The protocol is based on nano-watt WuRx proposed in [130] embedded with an energy harvesting WSN node. Whenever the transmitter has a packet, it broadcasts a wake-up beacon (WUB) indicating to other receivers that it is ready to send. Since the main radio has been used as a WuTx, WUR-TICER utilizes the same channel for sending the WUB and the data packet. As a result, WUR-TICER achieves a lower packet reception rate than the non-WuR model since the WUB collisions are frequent when two or more transmitter nodes wake-up at the same time and try to send a WUB to the base station. Moreover, the WuR is only simulated in a single-hop energy harvesting WSN with a continuous energy source.

Moving on to symmetric protocols, we find only one:

1. Duty cycled: STEM [109] is one of the first transmitter-initiated protocols that separates the data transmission channel from the wake-up channel by using a dual radio approach on separate frequency bands. Both the radios are high power radios while one of them acts as a WuR. Two variants exist in STEM. In STEM-T, a tone is sent which wakes up all the nodes in the neighborhood. STEM-T resembles the traditional preamble sampling approach but moves the data transmission to a separate channel. In STEM-B, a wake-up beacon is used as a preamble that includes the address of the destination node and the sender. A node thus can determine whether it is the intended receiver or not and the non-target nodes can go back to sleep earlier. Moreover, STEM uses a regular high power radio as a WuR to achieve the same coverage as the main radio. Duty cycling is applied to the WuR while the data radio is switched off unless required. However, both radios are high power radios and the power consumption is not reduced.
1.6 Routing Protocols utilizing WuRs

1.5.5 Summary

To make the wake-up radio based system feasible and energy-efficient, it requires careful design of energy-efficient protocols. The MAC layer plays a crucial role in coordinating how nodes share the common broadcast channel. The main role of this layer is to prevent simultaneous transmissions and data packet collisions at the same time granting energy efficiency, low channel access delays and ensuring fairness among the nodes in the network.

So far various asynchronous MAC protocols have been proposed for WuRs to extend the network lifetime and to increase reliability and throughput. Within this, different communication and power management techniques have been employed. For reducing latency, active WuR based MAC protocols have been proposed. This allows fast response and long communication as the radios are always on. Although this technique provides many advantages, it is less energy efficient as WuRs are always on and dissipate energy. To make this approach effective, energy-efficient WuR hardware design is required. Some works have proposed energy harvesting based MACs and use energy from the wake-up signal itself. The reason is to extend the node lifetime and to only turn on the device upon detection of the valid signal. Other sets of protocols have duty cycled the WuRs. Although this reduces power demand but encounters same problems as traditional MACs such as increased latency.

To enable on-demand communication, asynchronous WuR-MAC protocols have mostly adopted transmitter initiated probing for data transmission. The nodes are only equipped with the WuRxs (asymmetric) while the main radio is utilized as the transmitter. This allows only one-way communication and does not exploit the full potential of WuRs. For the WuR based system to be effective and applicable for various applications bi-directional MACs are more suitable. The main radio-activity is reduced by exchanging control information over the WuRs. This information can include channel or frequency reservation data, which otherwise would have been exchanged over the main radio.

As the number of WuR-MAC protocols grows, there are still many open questions such as the different performance they offer when applied to realistic applications. Most of the protocol evaluation concentrates primarily on simulation results and does not include any results from hardware implementations or testbeds. Moreover, their implementation relies on custom-design software limiting the reproducibility of the obtained results. Some of these works have quantified the benefits of using WuRs in terms of energy consumption through in-lab power measurements, but do not evaluate other relevant metrics, such as latency and end-to-end data reliability. While some of these protocols may work well in a small, single-hop network like a WBAN, it may lack in flexibility to work for more general WSNs with a large number of nodes.

1.6 Routing Protocols utilizing WuRs

In typical WSNs, hundreds or thousands of sensor nodes are scattered or placed throughout a large area. Each sensor has the capability to communicate, collect, and route data to other
Chapter 1. Ultra Low Power Wake-Up Radios:  
A Hardware and Networking Survey

Figure 1.17 – Taxonomy of wake-up radio based routing protocols.

nodes or back to the base station. Since, not all of these sensors are in range of the base station, data is routed in a multi-hop fashion. Over the last several decades, a plethora of routing strategies have been proposed for WSNs [131, 132, 133, 134]. However, most of these studies are based on single radio architecture. The scenario changes when routing is done over WuRs due to the network topology induced by it.

One of the challenges of introducing a WuR as a new component to an existing node with wireless communication is the mismatch between the ranges. By nature, WuR technology has shorter ranges, prohibiting a wake-up signal from triggering a distant node, despite the ability of the higher power radio to effectively reach it. This introduces new challenges for traditional routing protocols. In particular, for WuR based systems, packets need to be routed through longer paths than those of the main radio. This affects the data latency as well as the network lifetime. For applications with stringent consumption requirements, this may not be acceptable. To mitigate this, several WuR based routing protocols have been developed for flooding, multi-hop data collection and dissemination. Table 1.11 summarizes the WuR-based routing protocols that we survey while Fig. 1.17 arranges them in a taxonomy based on whether they address only the routing layer or are also cross-layer.

1.6.1 Routing-Only Protocols

Existing routing-only protocols exploiting wake-up radios can be classified into three categories: topology-based, load balancing, or tree-based.

Topology Based. Under this category, every node in the network maintains routing information such as its end-to-end distance to the sink and also the next hop to reach the sink. This information is usually obtained by the sink using a network wide dissemination of control messages. To forward a packet towards the sink, the node chooses the neighbor that has the shortest path as the next forwarder.

In [135], Stathopoulos et al. present a topology control mechanism for establishing the end-to-end paths in a WSN using the dual-radio system. Each node uses its low bandwidth wake-up radio to request an end-to-end path information to the destination nodes from the central topology controller. The novelty of this work is to use multiple short WuR hops to achieve
a single, long higher power hop by the main radio. This protocol is based on an out-of-band paradigm and supports multi-hop networks. Latency is the main issue here as path discovery using low data rate networks can be time-consuming. Since the topology controller is centralized, this can lead to a single point of failure, crippling the entire network.

The concept of semantic addressing using WuRs, in which a pool of multiple WuRx addresses is assigned to a node and dynamically updated based on its status, has been recently proposed [20]. A dedicated WuRx-enabled communication stack called **FLOOD-WUP** exploiting selective wake-ups and dynamic address assignment is implemented to enhance system performance. FLOOD-WUP enables transmission of commands from the sink to the sensor nodes in a reliable and energy efficient way. Comparing FLOOD-WUP against traditional flooding protocol has shown that nodes using FLOOD-WUP for interest dissemination are 4% energy efficient and require less energy to achieve full network coverage.

**Load Balancing.** Routing protocols designed for load balancing not only select the shortest paths towards the destination but can also consider the available energy of the nodes in the path in an attempt to extend network lifetime. The routing load is distributed over multiple paths in the network to improve packet latency and to minimize dropping packets.

To achieve reliable end-to-end data delivery, a load-balancing, and optimized data flow communication routing tree is proposed by Vodel et al. [136]. **WRTA** is a lightweight routing protocol for data-centric WSN environments that combines complex route path calculations and topology optimization mechanisms for asynchronous communications. In WRTA, the burden of energy consuming calculations such as maintaining routing path and network status is shifted from the sensing nodes to the sink. For load-balancing and route optimization, the shortest path is selected for nodes with a large amount of data depending on the energy level, QoS parameters and bandwidth of the nodes. WRTA was analyzed using both software and hardware experiments. It was observed that for a network with the depth of 3-hops, the proposed routing protocol experiences high packet loss when the number of packet generation increases to 7 packets per node/min.

**Tree-Based.** In tree-based routing, nodes form a tree-like hierarchy with the sink node as the root. Each node (child) at the particular depth of the tree transmits data to a node (parent) in the upper level of the hierarchy. This ensures data transmission in parallel and reduces packet latency significantly.

Recently, the authors in [137] extended the Collection Tree Protocol (CTP), the *de facto* standard for data collection in WSN to work with nodes coupled with WuRs [138]. **CTP-WUR** utilizes WuRs to relay wake-up requests and reduces end-to-end data latency, thereby, extending the achievable wake-up range. CTP-WUR can handle both broadcast and unicast packets. It has been shown through simulations that CTP-WUR performs better, obtaining latencies lower than tens of microseconds and is highly reliable compared to the standard CTP.
### Table 1.11 – Wake-up radio based routing protocols.

<table>
<thead>
<tr>
<th>No.</th>
<th>Protocol</th>
<th>Year</th>
<th>Path Request</th>
<th>Hardware</th>
<th>Addressing</th>
<th>Topology</th>
<th>Implementation</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>EAR [139]</td>
<td>2002</td>
<td>Source</td>
<td>Symmetric</td>
<td>ID-based</td>
<td>Distributed</td>
<td>Simulation</td>
</tr>
<tr>
<td>2</td>
<td>LESOP [140]</td>
<td>2007</td>
<td>Source</td>
<td>Symmetric</td>
<td>ID-based</td>
<td>Distributed</td>
<td>Simulation</td>
</tr>
<tr>
<td>3</td>
<td>Stathopoulos et al. [135]</td>
<td>2007</td>
<td>Source</td>
<td>Symmetric</td>
<td>ID-based</td>
<td>Centralized</td>
<td>Testbed</td>
</tr>
<tr>
<td>4</td>
<td>WRTA [136]</td>
<td>2012</td>
<td>Sink</td>
<td>Symmetric</td>
<td>ID-Based</td>
<td>Centralized</td>
<td>Testbed</td>
</tr>
<tr>
<td>5</td>
<td>FLOOD-WUP [20]</td>
<td>2014</td>
<td>Sink</td>
<td>Symmetric</td>
<td>ID-Based</td>
<td>Distributed</td>
<td>Simulation</td>
</tr>
<tr>
<td>6</td>
<td>CL-RW [141]</td>
<td>2014</td>
<td>Source</td>
<td>Symmetric</td>
<td>ID-Based</td>
<td>Distributed</td>
<td>Testbed</td>
</tr>
<tr>
<td>7</td>
<td>ALBA-WUR [35]</td>
<td>2015</td>
<td>Source</td>
<td>Symmetric</td>
<td>ID-Based</td>
<td>Distributed</td>
<td>Simulation</td>
</tr>
<tr>
<td>8</td>
<td>ZIPPY [17]</td>
<td>2015</td>
<td>Sink</td>
<td>Symmetric</td>
<td>ID-Based</td>
<td>Distributed</td>
<td>Testbed</td>
</tr>
<tr>
<td>9</td>
<td>CTP-WJR [138]</td>
<td>2016</td>
<td>Source</td>
<td>Symmetric</td>
<td>ID-Based</td>
<td>Distributed</td>
<td>Simulation</td>
</tr>
<tr>
<td>10</td>
<td>OPWUM [142]</td>
<td>2016</td>
<td>Sink</td>
<td>Symmetric</td>
<td>ID-Based</td>
<td>Distributed</td>
<td>Simulation</td>
</tr>
<tr>
<td>11</td>
<td>T-ROME [143]</td>
<td>2017</td>
<td>Source</td>
<td>Symmetric</td>
<td>ID-Based</td>
<td>Distributed</td>
<td>Testbed</td>
</tr>
<tr>
<td>12</td>
<td>WHARP [144]</td>
<td>2017</td>
<td>Source</td>
<td>Symmetric</td>
<td>ID-Based</td>
<td>Distributed</td>
<td>Simulation</td>
</tr>
<tr>
<td>13</td>
<td>GREENROUTES [145]</td>
<td>2017</td>
<td>Source</td>
<td>Symmetric</td>
<td>ID-Based</td>
<td>Distributed</td>
<td>Simulation</td>
</tr>
</tbody>
</table>

### 1.6.2 Cross-Layer Protocols

The protocols discussed so far were individually developed for a single layer of the stack i.e., MAC, Network, Transport, and Physical. While they exhibit good performance in terms of the metrics related to a single layer, they are not jointly optimized to maximize overall network performance while reducing energy expenditure. Therefore, a cross-layer design presents a promising alternative by streamlining communication between layers and providing the response based on a complete view of the stack, increasing system utility and energy efficiency.

**Energy-Aware.** The main objective of energy-aware routing protocols is to extend the network lifetime by choosing optimal paths. These paths are chosen depending on the energy budget so that no single path depletes its energy quickly. Rotating among paths leads to increased network lifetime as energy is dissipated equally among all the nodes.

A cross-layer energy aware routing (EAR) protocol using WuRs [139] uses sub-optimal paths to provide substantial gains in network lifetime. In EAR, the MAC layer is responsible for keeping the lists of all its neighbors and metrics such as the neighbor’s position and the energy required to reach it. Then, this list is accessed by the network layer to make decisions regarding packet routing. The energy level information is used as a weight factor when routing the data, avoiding the paths with less residual energy. Finally, to send data the MAC layer transmits a wake-up signal on the broadcast channel, modulating the address of targeted node with the wake-up signal. Even though this method takes energy into account, it does not consider end-to-end latency. Moreover, this protocol has only been evaluated through simulations.

**OPWUM** [142] offers another opportunistic cross-layer MAC protocol leveraging WuRx for selecting the best receiver among its neighboring nodes using energy as a metric. To overcome collisions between wake-up beacons, a clear channel assessment (CCA) is performed using the WuTx. Thereafter, an RTS-CTS is exchanged between the WuTx and WuRx before sending
any data packets via the main radio. One of the features of OPWUM is that all the next hop relay selection phase is carried out using wake-up beacons only. Nonetheless, this proposed protocol has not been tested using real experiments.

Unlike classical approaches, Low Energy Self-Organizing Protocol (LESOP) [140] presents a cross-layer architecture where both Application and MAC layers collaborate directly while Transport and Network layers are excluded to simplify the protocol stack. Inter-node communications are done by exchanging packets and busy tones. The main radio is responsible for handling all data packets while the busy tones are sent using the secondary low power wake-up radios. This protocol is proposed for target tracking applications in large wireless sensor networks. Similar to EAR, this protocol does not investigate the importance of system delay and is tested in simulations only.

**Geographic.** In geographic routing protocols, the data packet is routed towards the destination region using geographically informed neighbor selection heuristics. The key concept is to collect data from the selected region rather than sending it through the whole network hop by hop.

Spenza et al. [35] proposed ALBA-WUR, a cross-layer solution for data collection exploiting semantic node addressing features of WuRx to implement complex relay selection policies. For data routing and path selection, the protocol relies on ALBA-R, a cross-layer geographic protocol that features the integration of awake/sleep schedules, MAC, routing, load balancing, and back-to-back packet transmissions [146]. Simulation results concerning average end-to-end data latency show that the use of WuR technology together with ALBA-R is effective for cutting down the time needed to deliver packets to the destination. However, this delay is dependent on the data rate used to transmit wake-up signals. GREENROUTES [145] is another scheme that also uses semantic addressing. The main difference between ALBA-WUR and GREENROUTES lies in the handshaking procedure, i.e., ALBA-WUR uses only RTS whereas GREENROUTES performs both RTS and CTS before actual data packet transmission.

T-ROME, a cross-layer routing protocol that supports multi-hop communication, is presented in [143]. At the MAC layer, T-ROME uses RTS/CTS messages to reduce packet collisions over the WuR. At the network layer, the data forwarding mechanism of T-ROME is similar to ALBA-WUR but does not flood the whole network. In T-ROME the next hop node is chosen dynamically using link quality estimation over the WuRs to determine if the relay node is within the wake-up range. If so, the data is directly sent to that particular node without passing from each child to its parent. Therefore, T-ROME saves energy by skipping nodes during data transmission. Using small scale testbed, authors have shown that T-ROME outperforms CTP-WUR in terms of number of hops required to reach the sink with reduced latency and power consumption.

Basagni et al. [144] proposed WHARP, a cross-layer forwarding strategy where the channel access using the MAC layer and selection of the next-hop relay over the network layer are performed jointly. Nodes in WHARP leverage the combination of prediction-based techniques and Markov Decision Processes to decide whether to be a potential relay or not, while max-
imizing the node lifetime. Due to the energy-efficient relay selection strategy driven by a Markov Decision Process, nodes using WHARP maintain high operational time. On the other hand, WHARP experiences high network latency due to the cross-layer nature that requires nodes to engage in a time-consuming RTS/CTS handshake before sending a data packet.

**Flooding Based.** In this category, the node that has data communicates it to everyone else in the network using flooding. Multiple copies of the incoming packets are sent by the nodes that are in the broadcast domain which they forward to their neighbors. This technique generates a huge amount of redundant traffic. However, it does not require costly topology maintenance and route discovery procedures.

A practical application of ultra-low power sub-GHz WuR is presented by Sutton et al. [17]. **ZIPPY** is a cross-layer protocol that provides on-demand network flooding for the multi-hop network through the use of ultra-low power wake-up receivers equipped at each node, albeit with reduced per-hop range compared to using high-power transceivers. The ZIPPY protocol features asynchronous network wake-up, neighborhood time synchronization, bit-level data dissemination and carrier frequency randomization leveraging low complexity WuRs. Using ZIPPY reduces the entire network flooding time while maintaining end-to-end latency of only a few microseconds. As in its current implementation, ZIPPY does not address the false wake-ups making it susceptible to erroneous network wide wake-up.

Cross-layer Radio Wake (**CL-RW**) [141] builds on the transmitter-initiated paradigm by coordinating the wake-up beacon transmissions. The proposed mechanism uses an asynchronous scheduler for controlling its WuR, which is a cross-layer information from the MAC layer, to form an operation cycle. This cycle is a network-level duty cycle that is built on top of the duty cycles of individual nodes. Instead of transmitting wake-up beacons independently, each WuTx transmits during its allocated schedule. Therefore, the beacon transmissions in a network are coordinated to form a multi-hop path like a pipeline and the waiting time in each hop is significantly reduced. Furthermore, a node that has generated data can keep the radio off to save additional power. The proposed idea is compared to AS3-MAC [147] and the experiments show significant reduction in the power consumption.

### 1.6.3 Summary

This section has provided a classification of WuR based routing protocols, including also cross-layer approaches. Most of these studies have shown that by combining wake-up capabilities with selective addressing and routing over WuRs, node lifetimes can be extended to decades while achieving data latencies comparable to networks that only use the single main radio.

Most of the routing protocols discussed in this section assume static networks where the sensor nodes and the base station are stationary. An interesting issue to look into will be consideration of node mobility. For diverse applications of WuRs such as smart city or transportation, routing protocols for mobile WSNs will be beneficial to provide real-time delivery and wider coverage. Routing messages in a mobile scenario is challenging since route stability becomes
1.7. Key Application Areas

an important optimization factor, in addition to bandwidth and energy. Use of wake-up radios for mobility purposes requires optimization of transmitter operation, such as the number and time interval over which to transmit wake-up beacons so that they are correctly received by the low power wake-up receiver for controlling main radio operation as proposed in [148]. Novel routing algorithms are needed to handle the overhead of mobility and topology changes in such an energy-constrained environment.

Multichannel routing protocols have recently gained popularity in the context of WSNs, due to their ability to be resilient against interference and collision, providing a significant performance benefit over a purely static approach. Such protocols involve various challenges such as channel selection, hidden terminal problem, and channel hand-over. Thus, routing for multi-channel WSNs over wake-up radios needs to be further studied.

Network security is another aspect that needs to be considered. Routing protocols must be robust against eavesdropping and malicious behavior. An attempt to address this using wake-up radios has been made in [149].

Finally, most of the routing protocols that exploit wake-up radios for the WSNs have been evaluated principally through simulations. To assess the real benefit and the performance of these protocols, thorough testing in real environments with a large network is essential.

1.7 Key Application Areas

Over the decades, the application of WSN has increased, spanning from monitoring natural phenomena such as temperature and humidity to personal health. With the proliferation of low power and cheap semiconductors, WSNs are expected to gain even more popularity [2].

With the understanding of the ultra-low power WuR built in the previous sections, we now briefly discuss multiple emerging application scenarios that can take advantage of it. We then map the different prototypes and protocols suitable for each application. Table 1.12 offers an overview while the remainder of this section provides details.

1.7.1 Wireless Body Area Network (WBAN)

Wireless body are networks (WBANs), find applicability in medical applications and thus require high reliability. To support a variety of applications on or inside the body, systems must have low power consumption and support variable data rates [150]. As an example of the latter, a glucose level monitor requires less than 1 kbps while an ECG can reach 192 kbps [150]. Further, WBAN communication can be periodic, event-driven, e.g., triggered by detection of an alert condition, or on-demand, e.g., in response to an external request by a clinician to retrieve saved data.

WuR technology can be applied in two principle ways. First, it can be used as a trigger to initiate high data rate communication. Alternately, it can be used as a low rate, low consumption
Chapter 1. Ultra Low Power Wake-Up Radios: A Hardware and Networking Survey

Table 1.12 – Wake-up radio based application requirements.

<table>
<thead>
<tr>
<th>Applications</th>
<th>Range</th>
<th>Lifetime</th>
<th>Mode of Data Collection</th>
<th>Network Type</th>
<th>Latency</th>
<th>Data Rate</th>
<th>Addressing</th>
<th>Power Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>WBAN</td>
<td>-</td>
<td>++</td>
<td>Event-driven On-demand</td>
<td>Star/Single-hop</td>
<td>-</td>
<td>++</td>
<td>Yes</td>
<td>Active</td>
</tr>
<tr>
<td>Smart City</td>
<td>++</td>
<td>+</td>
<td>Event-driven On-demand</td>
<td>Node-to-node Multi-hop Mobile</td>
<td>-</td>
<td>+</td>
<td>Yes</td>
<td>Active Passive</td>
</tr>
<tr>
<td>Smart Metering</td>
<td>+</td>
<td>+</td>
<td>On-demand</td>
<td>Node-to-node Mobile</td>
<td>-</td>
<td>+</td>
<td>Yes</td>
<td>Active</td>
</tr>
<tr>
<td>Wildlife Monitoring</td>
<td>++</td>
<td>+</td>
<td>Event-driven Periodic</td>
<td>Node-to-node Multi-hop Mobile</td>
<td>-</td>
<td>-</td>
<td>Yes</td>
<td>Active</td>
</tr>
<tr>
<td>Surveillance</td>
<td>++</td>
<td>+</td>
<td>Event-driven Star Multi-hop</td>
<td>-</td>
<td>++</td>
<td>Yes</td>
<td>Active</td>
<td></td>
</tr>
<tr>
<td>Indoor Localization</td>
<td>+</td>
<td>++</td>
<td>Event-driven On-demand</td>
<td>Star Multi-hop Mobile</td>
<td>-</td>
<td>+</td>
<td>Yes</td>
<td>Active Passive</td>
</tr>
<tr>
<td>Asset Tracking</td>
<td>+</td>
<td>++</td>
<td>Event-driven Periodic</td>
<td>Star Multi-hop Mobile</td>
<td>-</td>
<td>+</td>
<td>Yes</td>
<td>Active Passive</td>
</tr>
<tr>
<td>Wearables</td>
<td>-</td>
<td>++</td>
<td>Event-driven On-demand</td>
<td>Star Node-to-node Multi-hop Mobile</td>
<td>-</td>
<td>+</td>
<td>Yes</td>
<td>Active</td>
</tr>
<tr>
<td>Smart Grid</td>
<td>+</td>
<td>++</td>
<td>Event-driven On-demand</td>
<td>Star Multi-hop Mobile</td>
<td>-</td>
<td>+</td>
<td>Yes</td>
<td>Active Passive</td>
</tr>
</tbody>
</table>

| Requirement Importance  | Low   | Very low | + High | ++ Very high |

data radio [151]. Notably, the short range is not an issue for these applications [152], and the extremely low standby consumption is a major advantage. For example, a receiver sensitivity of -40 dBm is sufficient to receive a signal transmitted with 0 dBm [153]. With low sensitivity demand, energy efficient WuRs can be implemented as a simple star topology with the number of nodes typically ranging from two to ten.

Matching Prototypes. From the list of prototypes in Table 1.8, there are 23 WuR designs that match the criteria for the first scenario. All of these designs are ultra-low-power consuming, below 10 µW, and have node addressing capabilities. For the second scenario where WuRs can be used as a full data transceiver, five concepts [15, 47, 53, 54, 80] are found to be suitable. Four out of five of these are tested prototypes while the design concept by Francois et al. [54] is only in simulation. Nevertheless, all of them have data rate above 200 kbps while exhibiting power demand below 10 µW.

Suitable Protocols. From the system design perspective, there are a few WuR enabled MAC protocols specifically designed for BAN applications. To offer high data rate and low latency, all of these are always-on wake-up MACs. The protocol proposed in [119] is transmitter-initiated while AWD-MAC [124] is receiver-initiated. However, we argue that the MAC protocols suitable for WBAN should be bi-directional so that anomaly can be reported effectively and on-demand. Protocols such as those presented in [23, 116, 117, 121] are best suited for this. For communicating data, WBAN applications require either star or single-hop network, therefore, the complex routing protocol is not essential.
1.7. Key Application Areas

1.7.2 Smart City

The concept of the Smart City is growing in popularity as sensors placed throughout cities are used to support both the public administration as well as citizens directly. A large number of the placed sensors exploit wireless communication and are battery powered, allowing them to be opportunistically placed. Nevertheless, this necessitates low power operation.

Today, a majority of smart city nodes communicate wirelessly over a variety of links such as IEEE802.15.4, IEEE802.15.4g, IEEE802.15.1 (Bluetooth), or low-power 802.11 [154]. WuRs can play a critical role in making these networks more energy-efficient, scalable, and autonomous. For example, a single-hop case can be built in which a mobile data collector, e.g., a bus or garbage truck, is equipped with a WuR. This mobile data collector traverses the city and collects information from WuR based sensing nodes deployed along its route. The sensing nodes will only be activated when the mobile data collector sends the WuS querying these nodes for data (on-demand) [155]. The feasibility of utilizing WuRs for data aggregation and for opportunistic networking in a smart city scenario has been demonstrated in [156].

Infrastructure monitoring is also possible by using WuRs in a multi-hop manner [157]. A stationary or mobile data collector can gather data from a chain of sensors attached to a bridge, tunnel or simply along the streets. WuR enables the higher power sensing nodes to remain in low energy mode when there is no data to send. Instantiating this scenario, however, necessitates a solution for the mismatch between the typical distance of the WuR and that of the primary radio.

Matching Prototypes. In order for the WuR to be suitable for smart city applications, it should support reasonable data rate, long communication range for wider coverage and low power operation. We have identified four prototypes that meet these specifications [15, 48, 35, 21], i.e., prototype numbers 15, 21, 41 and 67 listed in Table 1.8. Most of the prototypes in Table 1.8 do not meet this criterion since either power consumption is high, a factor that limits node lifetime if battery powered or has shorter communication range than 40 m. Non-RF WuRs are not suitable due to the requirements imposed by the hardware such as sensitivity and LOS for optical based systems.

Suitable Protocols The sensors deployed within the smart city may either report periodic or on-demand data with various traffic loads. This adds an additional reliability criterion in addition to coverage and responsiveness. For instance, infrastructure monitoring systems demand fast responsiveness and should be energy efficient. That is, the events should be rapidly detected and reliably communicated in an energy efficient way through a multi-hop network for post-processing. Thus, the protocol should support event-triggered as well as periodic sensing. Various flavors of the surveyed MAC protocols can be adopted. For low latency, broadcast based MAC protocols such as PTW, AWD-MAC, and STEM-B are suitable candidates. Sensors that may rely on energy harvesting technologies can utilize SLAM, WURTICER, and DoRa as main MAC. If a specific node is to be queried bi-directional MACs such as W-MAC are applicable. For periodic sensing where nodes can be switched off during periods
Chapter 1. Ultra Low Power Wake-Up Radios: A Hardware and Networking Survey

of inactivity, duty-cycle wake-up MAC should be considered.

After a certain duration, nodes may fail due to battery depletion or other external factors, therefore, new routes have to be established. Thus, the routing protocols should be adaptive and provide support for multi-hop data collection. For rapid data dissemination, network flooding protocols such as ZIPPY and FLOOD-WUP should be adopted.

1.7.3 Smart Metering

Smart meters enable remote, wireless reading of current meter values, eliminating the need for a technician to enter the home. Typical installations today place a mains powered, wireless communication unit on the meter and a mobile unit carried by a technician in a mobile vehicle. While this saves the time and energy of the technician to visit each meter, the radio itself must be powered to wait for the reading signal.

Instead, a utility meter equipped with a WuRx [158] can be activated on-demand, requiring zero or near-zero consumption in between readings. To be acceptable, the solution must have ultra-low consumption (10+ years battery lifetime at 1 reading per month). Since utility meters are usually placed inside the building, it should also have good radio signal penetration and high sensitivity operating in a sub-GHz frequency. Typically a communication distance of 15 m is required. According to communication standards for smart metering in Europe [159], the maximum allowed effective radiated power (ERP) in 868 MHz band is 25 dBm. A receiver with a minimum sensitivity of -75 dBm will be able to receive packets at a distance of 15 m. The required data rate for smart metering applications is moderate, supporting data rates between 2.4 kbps and 200 kbps. Moreover, the WuR should have addressing ability in order to query specific smart meter with its unique serial number.

Matching Prototypes. From Table 1.8, eight prototypes match the requirements imposed by smart metering application. The designs presented in [15, 48, 35, 21, 37, 17, 42, 43] exhibit power consumption below 60 $\mu$W with good receiver sensitivity and node addressing capabilities while offering tens to hundreds of kbps data rate.

Suitable Protocols. Usually, the communication will be infrequent and demand-driven, i.e., upon a request from the data collector, therefore, polling based (taking-turns) MAC protocols best suits smart metering applications. With regard to routing, various WSN protocols may be considered [160]. However, mostly WuR-enabled meters will communicate to the collector in one-hop, then complex routing protocols are not suitable but require to maintain end-to-end reliability with nodes to be uniquely identified.

1.7.4 Wildlife Monitoring

Use of sensor networks for wildlife monitoring has gained momentum in the recent years. Wildlife monitoring is essential for keeping track of endangered wild animal movement patterns, habitat utilization, population demographics, snaring and poaching incidents and
breakouts. For example, WildScope [161] project attaches sensor nodes on wild animals like deer and foxes to track and to study their interaction and feeding behavior.

Data collection from wildlife has been one of the hindrances in the past, thanks to sensor equipped animal collars it is much easier and cheaper now. These collars have various integrated technologies like GSM and GPS module for tracking, high power transceivers with long range for animal proximity detection and wireless data off-loading. Due to continuous mobility, the collars require battery power with lifetime extending from few weeks to months.

To prolong the lifetime, animal collars mostly use duty cycling MACs such as low power listening (LPL), where the nodes periodically wake-up, transmit the data and then go back to the sleep state. Normally, low sampling intervals ranging between an hour and a day is chosen, since a higher sampling rate would deplete batteries too quickly. However due to the periodic operation, if there are any events of interest such as interactions between animals during this inactive period of the sensor node, it will be missed and not detected at all.

The problems mentioned above motivates the use of WuR technique for lifetime extension in wildlife monitoring scenarios. Similar to health-care, the benefit of WuR for wildlife monitoring purposes can be two-fold: either it can be used as a "contact sensor" or as an initiator for data communication. For example, collars designed in WildScope [161] project use high power CC2420 radio to listen to the beacon channel for a length of time and captures the ID number of the nodes within its proximity. This method for contact detection is expensive in terms of high idle listening power consumption. Instead, WuRs can be used as a "contact sensor" while sniffing the channel for detecting other WuRs in proximity. In this manner, all the proximity beacons between animals can be captured in an energy efficient way. Not only it will reduce power consumption, but it will also reduce the latency of contact detection due to always on feature of WuR. One such example can be found in [129] where researchers have utilized WuRs to monitor contacts and encounters between individual bats.

As a communication initiator, WuRs can be used to trigger nodes in a multi-hop network for offloading data to the base station, where a logical connectivity map can be constructed. Researchers can put data collectors equipped with WuR plus data transceiver and large energy supply near places where animals are expected to aggregate such as water source or ponds. When the animals are within the range of the data collector, the radio on them will be triggered by the WuR. Then the collars can start transmitting the gathered sensor data via the main data transceiver to the data collector. Hence, the collars may last for years and the battery replacement and retrieval cost can be saved.

Matching Prototypes. The salient criteria for WuRs for wildlife applications is that it should be low cost, power efficient and communication range (>30 m) that allows the network to cover a much larger area with few devices. The prototypes that match smart city applications are also suitable here but do not demand high data rates. Nevertheless, the performance of radios in terms of communication range may degrade when moved to environments with varying vegetation, thus radios with high sensitivity plays a key role.
Chapter 1. Ultra Low Power Wake-Up Radios:
A Hardware and Networking Survey

**Suitable Protocols.** As far as MAC protocols are concerned, it should support both event-driven mode for applications like contact sensing as well as the periodic mode for data off-loading. Thus, adaptive MAC approach is required where during inactivity, the collars can save energy by duty cycling the WuRs and during encounters with other collars it can switch to continues listening mode. The MAC protocol should be able to dynamically adapt taking into account the collar activity.

In wildlife applications data is usually collected in delay-tolerant manner where it is stored locally and forwarded to the gateway when encountered with the mobile or fixed collector nodes. Low volume data can be forwarded using proactive routing algorithms that use shortest path such as EAR [139] or CTP-WUR [138].

### 1.7.5 Security and Surveillance Systems

Traditional security systems are based on high power central cameras that process and generate alarms if unauthorized objects or personnel are detected within the premises. Such systems are power hungry due to heavy image processing algorithms and require installation near the stationary power source. For applications such as continuous monitoring of large and wide area facilities, i.e. power plants, border lines, large factories, gas and oil pipelines with no stationary power source, infrastructure for cabling can, therefore, be very expensive.

WuRs with small, low cost and low power camera systems can thus be used to detect unauthorized objects, beyond the perimeter of some critical infrastructure. The monitoring area can be covered with several WuR based camera systems, working independently and stationary. All these units will be wirelessly connected to the main system for decision making. Once an intrusion is detected via wake-up cameras, the more powerful camera system can be triggered for verification and security action. To further reduce the camera activities, low power sensors with WuRs can be added as a separate network tier. The benefits for multi-modal sensing has been proposed in [162] and its extension with WuR is presented in [19]. The authors have presented a two-tier WSN for video surveillance applications where the communication between the PIR sensor nodes and the camera nodes is performed over the wake-up receivers.

**Matching Prototypes.** The coverage and the response latency are the important criteria for this application. The WuRs should be able to react quickly based on the information provided from the sensors thus requiring high data rates. Even WuRs consuming few milliwatts are suitable as long as communication range is greater than 50 m and data requirement is satisfied. The prototype designs by Hambeck et al. [48] and Petrioli et al. [20] are the ones that fulfill these requirements.

**Suitable Protocols.** Although duty cycling the WuRs on camera nodes will reduce power consumption, it also introduces response latency. To keep the latency at bay, an alternative solution is to use MAC protocols that are based on always on WuRs and continuously monitor the channel while keeping power consumption low (e.g., W-MAC). With regard to routing, a cost effective and reliable multi-hop communication network that relays the monitored
1.7. Key Application Areas

information in a timely manner is required so that efficient monitoring of the area can take place.

1.7.6 Indoor Localization

In the recent past, robust and accurate indoor localization for navigating has become one of the challenging areas for the WSN community since the GPS does not work indoors. One of the demanding applications of indoor localization besides navigation in shopping malls, user or robot localization, and environment modeling, is support for rescue teams during emergency scenarios. In life-threatening situations such as fire, rescue teams can often lose their orientation in smoky areas due to low visibility.

To increase the indoor localization accuracy within millimeters, these systems employ external reference points known as landmarks, for instance, Wi-Fi access points or ultra-wide band systems for taking extra measurements like Received Signal Strength Indicator (RSSI) or the Time Difference of Arrival (TDoA). These landmarks consume high energy, and either they require a continuous power supply or the batteries have to be changed frequently if always kept on. In catastrophic scenarios when there is no power available from the grid or if the batteries run out, landmarks will be of no use.

Integrating wake-up technology into these landmarks has the potential to extend the lifetime with improved energy consumption. Simon et al. [163] presented the idea of developing new WuR enabled wireless landmarks such as smoke detectors. During inactivity, these landmarks can be put into sleep state to reduce unnecessary energy wastage.

Matching Prototypes. The requirements for WuRs in the localization case are low power consumption, a communication range of few meters, and data rate in the region of several hundred kb/s. Moreover, there will be many landmarks within a building with devices operating at the similar frequency, therefore, the WuR should provide improved resistance to interference to prevent false wake-ups. WuRs operating in sub-GHz with communication range above 10 m should be preferred.

Suitable Protocols. In indoor localization applications, the navigating node will be frequently requesting the data from the anchor or landmarks deployed within the vicinity for updating the localization information. Thus, always on WuR-enabled MAC protocols are best suited for this. However, bi-directional communication is a must as the information will be shared to and from these landmarks. For emergency applications, the key requirement is to deliver messages in real-time and with a high probability of success, a challenging task in wireless sensor networks. To satisfy this requirement, adaptive or opportunistic routing protocols should be adopted to avoid routing holes (caused by nodes that have failed) or seek real-time and valid paths in emergency situations.
1.7.7 Asset Tracking

To improve operational efficiency in commercial businesses and to deliver quality customer experience, asset tracking during various phases is essential. Businesses as well as customers, both want to identify, locate and manage their assets in a timely manner. Traditionally, this process was done manually by registering product IDs when the items pass through certain warehouses or locations. A slightly faster method was introduced by use of bar codes for tracking items. However, these methods are time consuming and prone to human error. Recently, RFID technology based solutions have become more preferred choice of tracking items that uses radio signals. The items are attached with passive RFID tags and an active RFID reader is used to send signals to acquire data from these tags. Due to passive nature of the tags, the communication range is usually limited up to a few centimeters and to achieve up to few meters, large antennas are required.

To ameliorate above mentioned issues, active RFID tags have been integrated with wireless sensor nodes [164] such that the integrated tags are able to communicate with many wireless devices which are not limited to readers. The RFID system provides the product IDs while other information is communicated using the main node's radio. Consequently, active RFIDs are too costly and power hungry. Therefore, to bridge the gap between RFID and WSNs, RFIDs can be replaced with WuRs. For example, the WuRs can periodically transmit radio beacons that may contain the product ID and the timestamps forming an "smart object". Moreover, using the built-in selective wake-up method, these beacons can also serve as object selector. Thus, allowing specific nodes to be queried on demand.

Malinowski et al. [100] presented the idea of quasi-passive wakeup for asset monitoring. In this work WuRx has been integrated with sensor nodes acting as tags. Whenever the base station queries the tags for events, the wake-up receivers compare the signals against a threshold before activating the main CC2500 radio. If there are no queries, the main radio goes into sleep mode and the WuRx is kept active consuming only 25 $\mu$W of power. Another specialized tag embedded with wake-up radios and sensors has been recently developed [38] for indoor and outdoor asset tracking. The design is extremely power efficient, low cost and supports dual frequency for communication.

Matching Prototypes. To realize wake-up radio based enhanced smart objects long-term operation is an essential requirement. Energy harvesting WuRs such as those proposed in [74, 76, 80] are suitable alternatives for enabling autonomous long-term operation with minimum maintenance cost.

Suitable Protocols. There are two types of nodes utilized in asset tracking; the gateway that is connected to the on-line database and the reader nodes associated with each type of items. To successfully locate these objects, the bi-directional communication mechanism is essential where the gateway can query the reader nodes by requesting information while the reader nodes can respond through their WuRs. On the data collection side, energy-efficient and low-power routing protocol is needed for continuous asset tracking applications. Moreover,
in storage facilities such as warehouses where hundreds of sensor tags equipped with WuRs might be present, packet losses and interference will be an issue. Therefore, robust algorithms to counteract this issue needs to be considered. One possible solution is to use multi-channel protocols with the node-addressing feature.

### 1.7.8 Wearables

Nowadays, wearable electronics have the huge potential to enhance people’s lives every day. New devices like activity trackers, smart bracelet, smart clothes have appeared in myriad, bundled with appealing Apps and motivating people to be always looking forward to new services. Similar to most of the battery operated devices (e.g., smartphones), wearable electronics tackles the need to prolong the battery autonomy as long as possible as well as keeping the size small for comfortable wearing. The challenge is even harder if considering that most of the tasks required by wearable devices are data-streaming oriented (e.g. headphones, trackers, fitness equipment) and energy efficiency is a key for such devices.

The presence of WuR methods would enhance the device reducing remarkably the energy spent in idle time, when the user is not ready or not connected to the specific device, or not requesting for a specific service. Strategies, where wearable devices are combined with ultra-low power wake up radio have been already presented in [165]. Moreover, context aware applications can decide which wearable object need to be activated avoiding overlapping of services when not needed. Typically, wearable objects are connected using a well known and widespread wireless standard (e.g. Bluetooth Low Energy) to a smartphone, that is used as a central device for processing and forwarding the information to the internet. Considering that nowadays, smartphones follow owners almost all the day, the communication range of the WuR is not an issue and very low standby power consumption can be achieved.

**Matching Prototypes.** Wearable electronics share some characteristics typical to the WBANs, and considering the short distance, potentially several WuR designs reviewed in this survey could satisfy the application requirements, such as [15, 47, 53, 54, 80]. Nonetheless, most of the wearable devices offer BLE connectivity and some are equipped with Low-Power Wi-Fi. A WuR technology design in the 2.4 GHz such as one in [77] could facilitate in future the transition towards a comprehensive radio-on-chip which includes a wireless standard and WuR technology.

**Suitable Protocols.** There are a few WuR enabled MAC protocols specifically designed for wearables applications, and to the author’s knowledge none are specifically integrated into a standard like BLE or low-power Wi-Fi. To offer high data rate and low latency, a MAC could intensively be called by the wake-up event. Protocols such as those presented in [23, 116] are suitable for the wearable scenario and offer insights for an integration in Bluetooth radio protocols.
1.7.9 Smart Grid

Microgrids is a new trend for achieving energy efficiency in the distribution of the electrical energy. It is revolutionizing the normal electrical grids within the Smart grids. Realtime control services for monitoring the quality of the power distributed from big power generation plants toward small and distributed network make information and communication technology more crucial than in the past.

One of the main challenges of the smart grid applications is relying on efficient communication infrastructure and service. Communication between measurement points is often realized using heterogeneous technology, both wireless and wired. Among these, power line communications (PLC) is a straightforward non-wireless choice. Several wake-up mechanisms that share similar medium, requirements, and protocols have been already proposed [166, 167].

A wake-up based approach can be implemented with a very simple and low-power device that constantly observes the communication channel and informs a host system whenever activity is detected. Since the power consumption of the wake-up is lower than PLC receiver, the overall energy consumption of the communication in the microgrid is drastically reduced.

Matching Prototypes. Micro-grid communication often includes heterogeneous technologies. Some of the prototypes that suit smart grid requirements are presented in [15, 48, 168], while non-RF wakeup circuits such as those in [166, 167] also exists even if with features tailored for cable communication [167].

Suitable Protocols. Protocols for wireless communication in smart grids may either report periodic or on-demand data with various traffic loads. This requires also fast responsiveness at the lowest energy cost. MAC protocols, therefore, should focus on the latency optimization and on the multi-hop characteristic of the network. In these cases, protocols such as PTW, AWD-MAC, and STEM-B are suitable candidates. For the wake-up mechanism developed over PLC (on cables), the protocol needs tight synchronization and the one proposed in [166] is suitable for the purpose.

1.7.10 Summary

This is certainly not an exhaustive list, with are many other applications that can benefit from WuR technology including building automation, smart lighting, remote keyless entry, aerospace to name a few.

Depending on application demands, the requirements for low power WuRs differ. For some applications, a high data rate is essential, while for others long communication range is of importance.

Table 1.12 provides an overview of different application requirements, which can be used as a reference for system developers to assist in categorizing and choosing the appropriate low
1.8. Open Issues, Challenges and Future Research Directions

power WuR. However, one should note that these requirements are not meant to be strict, but rather offer guidelines that one should keep in mind while designing WuR based systems.

As can be seen, the health-care case has the highest demand for data rate because of possible multimedia applications and most stringent power requirements. The highest requirement for communication range is given in the smart city application case, closely followed by wildlife monitoring applications. Only health care applications have moderate sensitivity requirements owing to the shorter communication range.

Generally, all applications demand node addressing capability in order to query particular nodes.

Further applications can be realized if wake-up radios are designed with standalone devices. An integration into transceivers as a substitution for built-in wake-on-radio mechanisms can further optimize these applications. Finally, low power consumption for WuRs in the ultra-high frequency (UHF) band offers a vast number of new services and applications.

1.8 Open Issues, Challenges and Future Research Directions

This section presents some of the main issues and challenges that must be addressed while designing systems based on WuRs. The challenges are not only related to hardware designs but also to the design and efficiency of upper layers of the stack. We then discuss some of the research directions that can be taken to mitigate these issues as discussed next and presented in Table 1.13.

1.8.1 Hardware Design

The evolution of the WuR technology is mainly driven by advancements in core technology and the demand for ever-less power consumption.

Cost and Technology Integration. Cost is one of the major factors, which is taken into consideration when designing and deploying large scale WSNs. So far, the small form factor and low hardware cost have been the key success indicator for WSNs. With the inclusion of WuR, the overall cost is expected to rise and can become one of the hurdles of this method. Further, the cost of designing ultra-low power WuR is still challenging. Current WuR have a shorter communication range than the traditional radios, making it difficult to align coverage of these two radios. For wide area coverage, high-density deployment will be required leading to higher maintenance costs. Recently, to address this issue Magno et. al [169] have proposed a new IoT node integrated with LoRa technology and energy harvesting wake-up receiver for long and short range networking. Another design that fuses wake-up radio and BLE technology with energy harvesting has appeared in [77].

Most of the presented features, such as addressing and in- or out-of-band communication, need to be implemented in a single chip with the main radio. Keeping a dual radio mechanism
Chapter 1. Ultra Low Power Wake-Up Radios:  
A Hardware and Networking Survey

Table 1.13 – Summary of issues, challenges, and opportunities for wake-up radio based hardware and software designs.

<table>
<thead>
<tr>
<th>Category</th>
<th>Dimensions</th>
<th>Issues / Challenges</th>
<th>Opportunities</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Hardware Design</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Cost and Technology</td>
<td>- short communication range</td>
<td>- small form factor designs</td>
<td></td>
</tr>
<tr>
<td>Integration</td>
<td>- high deployment cost</td>
<td>- cheaper SoC</td>
<td></td>
</tr>
<tr>
<td></td>
<td>- separate radio modules</td>
<td>- single chip packaging</td>
<td></td>
</tr>
<tr>
<td>Power Demand</td>
<td>- always on receivers</td>
<td>- design of energy harvesting WuRs with low latency</td>
<td></td>
</tr>
<tr>
<td></td>
<td>- low receiver sensitivity</td>
<td>- ultra low power transmitter designs</td>
<td></td>
</tr>
<tr>
<td></td>
<td>- non-negligible listening</td>
<td>- novel hardware design with short and</td>
<td></td>
</tr>
<tr>
<td></td>
<td>power</td>
<td>long range capabilities</td>
<td></td>
</tr>
<tr>
<td></td>
<td>- reduced data rate</td>
<td>- design of low power, high sensitivity WuRs</td>
<td></td>
</tr>
<tr>
<td></td>
<td>- high transmission power</td>
<td></td>
<td></td>
</tr>
<tr>
<td>System Architecture</td>
<td>- no unified system and</td>
<td>- WuR designs with multiple integrated sensors</td>
<td></td>
</tr>
<tr>
<td></td>
<td>networking architecture</td>
<td>- modular architecture for easy integration</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- flexible and open source designs</td>
<td></td>
</tr>
<tr>
<td><strong>Software</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Protocol Designs</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Channel Sharing</td>
<td>- static channel assignment</td>
<td>- multichannel MAC and routing protocols</td>
<td></td>
</tr>
<tr>
<td></td>
<td>- asymmetric network</td>
<td>- dynamic spectrum selection</td>
<td></td>
</tr>
<tr>
<td></td>
<td>thus mismatch</td>
<td>- dynamic channel handover</td>
<td></td>
</tr>
<tr>
<td></td>
<td>of transmission ranges</td>
<td>- WuR integration with cognitive radios</td>
<td></td>
</tr>
<tr>
<td>Synchronous WuR-MAC</td>
<td>- time synchronization</td>
<td>- synchronous transmission over WuR</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- efficient time synchronization mechanism</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>with low overhead</td>
<td></td>
</tr>
<tr>
<td>Adaptive Protocols</td>
<td>- static network parameters</td>
<td>- design of traffic adaptive protocols</td>
<td></td>
</tr>
<tr>
<td></td>
<td>- non-adaptive</td>
<td>- dynamic route maintenance</td>
<td></td>
</tr>
<tr>
<td>Mobility</td>
<td>- static nodes</td>
<td>- design of mobility based routing</td>
<td></td>
</tr>
<tr>
<td></td>
<td>- unstable mobile routes</td>
<td>- need for novel topology aware routing</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>with minimum overhead</td>
<td></td>
</tr>
<tr>
<td>Interference and</td>
<td>- high interference with</td>
<td>- multichannel MAC and routing</td>
<td></td>
</tr>
<tr>
<td>Coexistence</td>
<td>nearby in-band devices</td>
<td>- dynamic channel hand-over mechanisms</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- robust wake-up beacon modulation techniques</td>
<td></td>
</tr>
<tr>
<td>Standardization</td>
<td>- none available</td>
<td>- requires standardization of</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- frequency</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- channel availability</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- wake-up beacon format</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- hardware design</td>
<td></td>
</tr>
</tbody>
</table>

using separate components is expensive for IoT device production. This also includes the RF front-end circuits whose WuR performance mostly depends on the chip design. The possibility to have everything pre-assembled or packaged in a well-characterized module or component will pave the way to create a mass diffusion of such technology. An integrated design including the non-volatile baseband processor with wake-up identification receiver and power management module has been recently proposed in [99]. Although the architecture has been tested only using simulations, it opens up new hardware design opportunities.

**Power Demand.** In WuR based systems, always-on WuRs constantly dissipate energy, thus designing a transceiver that consumes orders of magnitude less than the main radio is necessary. The power demand of WuRs is also dependent on other factors such as reception sensitivity and data rate, which dictates the radios performance. All these factors must be considered and the trade-offs among them should be exploited.

While passive wake-up radios are an attractive and alternative means to save energy, it also poses few challenges. Harvested energy is very sensitive to environmental conditions and where energy sources are not always available, the wake-up procedure may be delayed. For delay-sensitive applications, such designs may not be suitable. Therefore, an open issue is how to reduce this delay with passive systems. Recently authors in [170] have investigated how to use energy harvesting based wake-up radios together with error control coding to enhance
the performance of networks while reducing carbon footprint.

Further, passive WuRs have shorter communication ranges than active ones. The wake-up signals are transmitted at high power to achieve long range thus incurring high energy cost. This demands low power wake-up transmitter designs similar to wake-up receivers that are simple to implement, turn ON almost instantly, transmit a short WuS and go back to the sleep state. A few works have proposed techniques such as the use of directional antennas [13], antenna diversity [12], and ultra long range RFID [82] to improve the transmission range of these radios.

At the same time, power consumption and receiver sensitivity will still be the major drivers to determine the future direction of WuRs; because they characterize the operating range of WuR. The transmission range of any radio communication will be the major driver for the coming generation of IoT devices. Low power communication is rapidly evolving towards multi-kilometer ranges and low bit-rate schemes. Long range sub-GHz radios such as LoRa [171] or Sigfox [172] are pioneers of this IoT communication revolution. If WuR technology does not advance with its features, it will be hampered in this market.

**System Architecture.** Currently no unified system and networking architecture exists for WuRs to build applications on top. The integration of different types of sensors, energy harvesters, and RFID tags may necessitate new and modular WuR architectures.

**Protocol Design.** Although the notion of wake-up radio eliminates the complexity that is involved with duty cycling MACs, there are many other challenges that need to be taken into account. Power consumption is also affected by the channel conditions, topology of the network, and the routing protocols utilized. Some of these challenges and issues are discussed next.

**Channel Sharing.** Sharing channels between wake-up and main radios must be studied since these two network layers have mismatched transmission ranges, forming an asymmetric network. Designing protocols that are more responsive to channel changes is still an open issue. There are a few research works that have attempted to address this such as CTP-WUR [138], Guo et al. [108], and WUR-MAC [113]. One solution to opportunistic spectrum access is using cognitive radios. Recently, cognitive radios have been incorporated in sensor networks [173, 174, 175]. Traditional radios assume fixed channel allocation and usually operate in crowded unlicensed bands that are also used by other devices making them prone to interference and collisions. Cognitive radios have the ability to opportunistically select the unused spectrum either in a licensed or unlicensed band. Combining WuRs with cognitive radio may enhance the overall system performance by increasing the communication reliability, alleviating collisions and packet losses, and improving the energy efficiency in dense networks. Due to its dynamic spectrum selection mechanism, multiple overlaid networks can also be realized without channel contention.

A few works have proposed dynamic channel selection by integrating wake-up radio with Wi-Fi modules. Specifically, Takeru et. al [176] utilized a frame length detection mechanism
Chapter 1. Ultra Low Power Wake-Up Radios: 
A Hardware and Networking Survey

with OOK modulation for selecting the appropriate Wi-Fi channel for transmission. Instead of only using wake-up radios for remote triggering, the authors in [177, 178] have also used it for carrier sensing before transmission by integrating it with WLAN. Standardization of wake-up receiver integration with WLAN has also started [179].

**Synchronous WuR-MAC.** Wake-up radios can also be utilized with synchronous MAC protocols for reducing latency and energy consumption [17, 180]. However, such designs require time synchronization among the nodes. WuRs are even more resource constrained devices than typical motes in terms of processing power, memory, available energy, and communication bandwidth. Thus, complex time synchronization protocols and heavy control overheads may not be feasible and requires careful design.

**Adaptive Protocols.** As seen in Section 1.7, WuRs can be utilized for applications that have harsh environments such as structural, animal monitoring or for emergency response where nodes are prone to failures. This may lead to other issues such as transmission failure or long latencies due to poorly designed MAC and routing protocols. To mitigate this, robust and adaptive protocols utilizing WuRs needs to be designed. These protocols should be traffic adaptive, avoid routing holes, and establish new routes dynamically in order to deliver messages reliably and in real-time. WuRs also exhibit shorter communication range than main radios. The design of such protocols is an open research direction.

**Mobility.** Another possible area of research is the consideration of node mobility in wakeup schedule design (e.g., [129, 128]). Most of the existing schemes assume that the sensor nodes and sink are stationary. Asynchronous and non-collaborative synchronous schemes are good candidates for these scenarios because their lack of coordination requirement makes them robust to network topology changes. In the presence of node mobility, schemes that require coordination may not converge to an optimal schedule or may generate excessive overhead. How WuRs will behave in such situations is still unknown.

**Interference and Coexistence.** The propagation impairments of wake-up radio signals in harsh environments such as forest, industrial or inside human-body also needs to be considered while designing WuR based systems. According to our survey, this so far has not been widely studied. An initial study by Lebreton et.al [181] looks into the in-band interference from nearby Wi-Fi devices on a wake-up radio system. The results indicate that wake-up radios are able to maintain high performance in coexistence with external wireless networks while slightly compromising on energy efficiency. Further investigation and study of the aforementioned propagation issues in different settings need to be conducted.

**Standardization.** It is important to remark that there is a clear lack of standardization activities related to the WuR designs such as (i) frequency usage, (ii) available channels, (iii) maximum power below which a radio can be classified as a WuR, (iv) wake-up signal format, and (v) routing topology. To address this, in July 2016, a wake-up radio study group (WUR SG) has been set up within the IEEE 802.11 working group to standardize the above activities [179]. The main aim of this group is to enable an energy efficient data reception for wake-up radios integrated
1.9 Conclusions

with WLANs without increase of latency. An attempt has also been made to standardize the wake-up radio packet structure so that it is compatible with different technologies in the area of medical applications [22].

1.9 Conclusions

The survey in this chapter identifies growing interest across the many facets of the design space of wake-up radios. Available hardware is expanding, with improvements in range, sensitivity, and consumption. Protocol stacks are emerging to exploit the novel properties of this technology, opening new application domains. Future work will require coordinated efforts at all levels to address limitations such as the difference in transmission range between a wake-up receiver and a traditional, higher power receiver. Further, issues such as interference must be studied to understand the reliability and robustness of systems incorporating wake-up receivers. Nevertheless, the potential of wake-up receivers to dramatically reduce the power consumption footprint of wireless, battery powered networks has been clearly demonstrated, offering motivation for future work.
Assessing WUR: Simulation and Testbed
2 WURBench: Toward Benchmarking Wake-up Radio-based Systems

With the proliferation of the Internet of Things devices and the seemingly endless connections among people, the demand for robust, dependable, and long-lasting devices is becoming critical. IoT devices are expected to be network connected at all times, even while simply waiting for an event to happen. Nevertheless, conventional solutions have significant costs even in this standby state, consuming on the order of a few milliwatts, reducing the useful device lifetime.

One emerging solution to extend battery life is the incorporation of a Low-Power Wake-up Radio (LP-WUR) [182]. A LP-WUR consumes only micro-watts of power, usually below 100 $\mu$W allowing it to be always-on, listening for an external trigger either on the same channel used for data communication or on a dedicated, out-of-band channel. Once a trigger is received, the LP-WUR activates the primary device, which can otherwise remain in a deep sleep state, saving energy. This extremely low power constraints of WUR limit the receiver complexity, modulation scheme, and thus the overall receiver sensitivity of WUR designs, reducing the effective communication range.

The potential for LP-WUR is rapidly expanding into domains such as Wi-Fi access points, low-power wide area sensor networks [183, 184], and wildlife monitoring [185]. As LP-WUR moves towards a widely used practical technology, simulations, and experimental deployments must be performed to validate proposed hardware designs, network architectures, and wireless communication protocols.

As studied in the previous chapter, current state-of-the-art WUR prototypes, most of which are custom in-lab designs display significant diversity in their architectures, processing capabilities, energy consumptions, and receiver sensitivities. Given such diversity in platform characteristics, choosing the right prototype and protocol for a specific application scenario is challenging. Nevertheless, the available communication protocols are evaluated in restricted settings with ad-hoc experiments and without comparison to competing approaches, making it almost impossible to identify the appropriate protocol for a given prototype radio in a specific application. Moreover, the results of experiments performed in different settings (e.g., topology, traffic pattern, interference) using different metrics may not hold for others. Currently, there exist no fixed set of accepted testing methods, parameters or metrics to be
applied to a WUR-based system under evaluation. This lack of standardization significantly increases the difficulty to develop new systems and/or apply existing technologies in novel domains.

**Contributions.** To overcome these disparities and offer progress in this direction, this chapter identifies the key parameters of a new evaluation methodology, WURBench, offering a step toward enabling accurate and repeatable profiling of WUR-based systems for IoT applications. Further, this chapter outlines the issues that must be addressed before full WUR benchmarking can become a reality. The main goal of WURBench is to outline a benchmarking framework that will:

- provide a set of recommended practices for performance evaluation.
- offer reliable indicators in terms of key performance metrics, parameters, and tools for researchers to test and fairly compare new solutions against existing ones or baselines when implementations are not publicly available. WUR hardware designers can also utilize this framework to benchmark devices against competitors.
- facilitate a repeatable test environment for WUR-based systems.

The concept of benchmarking is not new and has been applied to areas such as wireless networking [186] and CPUs [187] to compare performance results. Recently, IoT-Connect [188], an Industry-Standard Benchmark for Embedded Systems has been introduced to evaluate micro-controllers with various connectivity interfaces such as Bluetooth, LoRa, and WiFi. A benchmark typically outlines a set of specifications to follow when evaluating the performance of a system, making experiments repeatable and results directly comparable. These specifications include the definition of the parameters for the experimental setup and output metrics reflecting the performance of the benchmarked system. Benchmarking WUR is non-trivial as this not only requires evaluating the WUR prototypes and protocols but also measuring or modeling the wireless environment such as interference sources that have the potential to significantly affect system performance.

**Structure of this chapter.** In the next section, we outline the main specifications to follow in terms of "what to measure" and "recommended practice" when benchmarking wake-up radio hardware. In Section 2.2, we then outline the benchmarking criteria for the wake-up radio system as a whole. We start with the definitions of the key performance metrics to understand the performance of systems exploiting this technology and then, we highlight the different tools that can be used to perform repeatable benchmarking experiments. Finally, we present our conclusions in Section 2.3.

### 2.1 Wake-up Radio Hardware Micro-benchmarking

The design phase of WUR-based low-power networking starts with choosing an appropriate WUR prototype and performing a series of hardware specific micro-benchmarks. Micro-benchmarks are small test applications that iterate through the states of the component being tested e.g., radio, MCU, LEDs. As WURs are mostly custom designed, micro-benchmarking
2.1. Wake-up Radio Hardware Micro-benchmarking

enables the identification of possible performance bottlenecks at the architecture level, allowing hardware designers to compare and assess design trade-offs. Most often these micro-benchmarks are conducted in an ad-hoc fashion limiting comparability. Here, we seek to provide a well-defined structure, defining "what to measure" and "recommended practice" for measuring, allowing results to be compared across prototypes and to verify the fidelity of the test platform.

2.1.1 What to measure?

The first step is the definition of the metrics; hardware performance measured in terms of a set of quantitative variables of interest. At an abstract level, the metrics we define here are hardware-agnostic making them comparable across various WUR prototypes.

(i) Communication range: the achievable distance between the endpoints to establish a baseline for the performance of a given connection. For instance, evaluating transmit power vs. range is important for WUR deployments.

(ii) Successful wake-up rate: is the communication reliability of the WUR module measured in terms of the frame loss, which is the fraction of triggers sent by the sender over those successfully received at the receiver. This metric depends on the effective communication range and the strength of the wake-up signal.

(iii) Energy consumption: right now there is no common way to universally benchmark the energy efficiency of the WUR and one must either independently benchmark or rely on the information provided in the literature. Energy consumption, for instance, may refer to the average power consumption of the WUR in the continuous channel monitoring state, which is of greater importance for the IoT applications. Depending on the nature of the micro-benchmarking, this may also refer to the energy consumed by the WUR while executing different tasks for e.g., signal transmission and processing cost.

2.1.2 Recommended practice

To correctly measure the defined metrics, it is therefore mandatory to layout the steps one needs to follow while conducting these experiments, specifying the experimental parameters when characterizing these micro-benchmarks. The parameters are the configurations that allow controlling the execution of the micro-benchmark. This is a critical piece of the evaluation as comparing WURs without outlining all the configurations bring into question the soundness of the comparison. The main parameters are identified as:

- Physical layer (PHY) settings: Most WURs support various PHY settings that include bit rates, transmission power, and modulation. Often, some of these parameters are not identified or provided, making it difficult to directly compare the prototypes, as seen in chapter 1, Table 1.8. Therefore, the configurations used in each experimental trial must be reported with the results.
Chapter 2. WURBench: Toward Benchmarking Wake-up Radio-based Systems

- **Antenna orientation**: The radiation patterns of the antenna determine the performance of wireless devices [189]. It thus becomes important to state the antenna orientation and type in combination with frequency and transmit power.

- **Trial duration**: Wireless links change over time due to subtle changes in environmental conditions. Therefore, evaluation should be spread throughout a 24 hour period and long duration experiments should be preferred.

- **Firmware**: The firmware version being used in the tests as well as any functions that have been disabled, should be reported together with the results.

- **Environment**: The micro-benchmarking process is incomplete without describing the characteristics of the set-up environment that can be either based on real-life use case or artificial test environments. For a fair comparison, describing and documenting the scenarios for later analysis is critical. The test setups can be divided coarsely into two categories: i) shielded: where cabling or RF shielding techniques are used to attenuate external signals and noise. ii) open-air: environments that mimic the actual use case for the WUR such as indoor or outdoor environment with line-of-sight and non-line-of-sight.

### 2.2 Benchmarking System as a Whole

While it is important to isolate functions for performance testing, benchmarking the software separately may not reveal all the functionalities or vulnerabilities of a system. As software alone may not be able to capture underlying hardware behavior such as interrupts, timings, and design flaws, a complete-as-possible performance test must be conducted. In other words, benchmarking the whole system including the hardware and software interaction provides the most realistic evaluation.

For WSN systems, validation of the whole system is carried out using both, simulations and testbeds, following these key steps: i) defining the application scenario, ii) choosing or implementing a communication protocol, iii) conducting a large set of experiments, iv) measuring the performance in terms of defined metrics, and v) comparing the results. This seemingly simple task of benchmarking is surprisingly challenging for WUR networks as it requires an evaluation of the entire system to capture real-world operational conditions. This requires conducting a large number of experiments that can be tedious and error-prone. Moreover, the complexity of this evaluation is compounded by the lack of control over experimental conditions and limited number of evaluation tools. Furthermore, results obtained from ad-hoc experiments are difficult to compare with the results gathered from different wireless testbeds and simulations, hindering repeatability.

In this section, we focus on enabling benchmarking of WUR-based systems as a whole by presenting a set of testing methods, application scenarios, parameters, and metrics to be applied to a protocol under test either using testbeds or simulation tools.
2.2.1 Application settings

The first critical element to define for a system as a whole is the expected environment in which it will be exploited, as this defines many of they key environmental parameters that influence a deployment. For example, the size of the area to be covered and the density of measurement points affect the network topology. Further, the application needs often direct the choice of the protocol, for example data collection favors unidirectional focus while control systems emphasize latency. These choices must be outlined clearly, as they help focus the applicable metrics which we define in the next section.

2.2.2 Metrics

Next, we define key system performance and techno-economic metrics that we consider in the definition of WURBench. These can be divided into four categories:

(i) **Power consumption**: computed using the amount of time a node keeps its radio on in different states such as Receive (RX), Transmit (TX), Idle, and Sleep. The consumption should also consider duty cycle patterns of both the radios (main + the wake-up radio) to detect even small deviations that may have a substantial effect on the device lifetime in real deployments.

(ii) **Reliability**: defined as the fraction of application data packets successfully received over those sent. This is an indication of the level of service provided to applications in delivering sensed data, especially relevant when WUR is considered an option for safety-critical systems.

(iii) **Latency**: defined as the end-to-end delivery delay from the time of data packet generation to reception.

(iv) **Cost**: one limitation of WUR comes from inherently low-power demand in continuous listening mode, which results in a limitation on the feasible distance between two devices. As such, the WUR needs to be combined with another system, typically with a high power node resulting in a dense network. The relative cost of replacing a standard node with a WUR-based one might incur additional cost. The system cost should, therefore, be calculated not only for the main sensor node but also for the extra WUR hardware.

2.2.3 Evaluation mechanisms

Before proceeding with field tests, simulations and testbeds are the main tools for performance analysis of wireless systems allowing researchers to perform repeatable experiments.

**Simulations.** As WUR technology is still in its relative infancy, many simulators have been extended for evaluating LP-WUR protocols. For benchmarking, simulators offer many advantages over testbeds. Various network topologies such as single- and multi-hop with different
traffic patterns can be implemented and optimized with easy data collection for extracting the metrics. Large-scale networks for scalability analysis can be easily modeled, which otherwise would be too expensive to realize using testbeds. Furthermore, repeatability is easily achievable in simulations. On the other hand, simulators are criticized for not being able to capture all details, especially at the PHY layer, such as path loss, fading, and interference, bringing into question the applicability of simulation results. Nevertheless, simulation can provide valuable results for the initial benchmarking.

**WUR simulators.** Recent interest in WURs demands simulation support to allow systematic exploration of this novel technology. In [190] OMNET++ extensions provide a modular simulation model for WURs. It employs the MiXiM framework and offers reliable primitives for wireless signal propagation, energy consumption, and a complete networking stack. Similarly, GreenCastalia simulates a power model for wake-up receivers [191]. These simulators, however, do not offer code portability from simulation to real system. To fill this gap, one of the contributions of this thesis is an open-source tool, WaCo for assessing and evaluating wake-up radio based systems (refer to chapter 3). WaCo is an extension of COOJA network simulator and has been augmented with WUR hardware. WaCo supports node emulation for MSP430 platforms and uses binary, deployment-ready firmware, providing the ability to move between simulated and real experiments. WaCo offers a full networking stack with various signal propagation models such as Multi-path Ray-tracing and Unit Disk Graph. Moreover, it allows simulation of multiple embedded operating systems and is also the first open-source tool for evaluating WURs (https://github.com/waco-sim). As an example of the ease of comparing protocols with WaCo, Figure 2.1 illustrates the benchmarking of three different MAC protocols; wake-up radio MAC (W-MAC), low-power listening MAC (ContikiMAC), and always-on (NullRDC) MAC for a network of 100 nodes over Collection Tree Protocol. To have a fair benchmarking, the same application is run on top of all the MAC protocols with same settings while varying the network traffic. As expected, WUR solution (w-MAC) not only improves the network reliability but also reduces the overall latency over other MAC protocols, motivating
2.2. Benchmarking System as a Whole

further study of such systems using testbeds.

**Testbeds.** There is an increasing demand for experimentally-supported results to identify issues that cannot be captured through simulation or theory alone. This observation is reflected in the topics of the flagship conferences that increasingly encourage experimentally-driven research for validation.

Various shared and private testbeds exist, including the FIT IoT-LAB [192], FlockLab [193], and Indriya [194]. These allow scheduling experiments remotely, executing protocols directly on hardware, as well as collecting and extracting metrics of interest from the logged data. However, none of these testbeds currently support WUR functionality. We discuss next some of the key functionalities that the testbeds need to offer and how they could be implemented for benchmarking WURs.

(i) *WUR interface*: first and foremost, testbeds must offer hardware with the WUR interface. One cost effective option can be to support only a few nodes as done in [17].

(ii) *Experiment configurations*: the testbeds must provide experiment scheduling capabilities with the ability to configure a number of system parameters such as network topology and size, traffic load and pattern, experiment duration, physical layer settings for the radios including WUR and the main data transceiver. As noted, these configurations must be clearly reported to allow comparison.

(iii) *Monitoring the environment*: test facilities should provide information about the environmental conditions during the experiment. External wireless interference degrades network performance and to investigate this, spectrum analysis is indispensable. The testbed infrastructure should allow recording and replaying of the wireless traces. Tools such as JamLab [195] are key to producing repeatable interference. Temperature, instead, affects the clock oscillation of the devices and the testbed such as TempLab [196] offers temperature profiling for sensor nodes.

(iv) *Data archiving and sharing*: to extend the value of measurements beyond a specific case study, open-source data repositories are necessary. This facilitates archiving, publishing, and comparing of system performance data.

(v) *Result analysis*: testbed infrastructure should be able to extract the key metrics such as power consumption, end-to-end reliability, and data latency. For instance, these metrics can be extracted non-intrusively on testbeds using tools such as IoT-Connect [188] or D-Cube [197], avoiding the probing effects of instrumentation.

Ideally, experiments should be performed in multiple testbeds to achieve statistical significance. However, it is critical to do an “apples to apples” comparison in any benchmarking exercise, and failure to consider all variables can produce results that are misleading or even erroneous.
2.3 Summary

This chapter offers the first steps toward WURBench, a benchmarking framework tailored to the unique properties of the emerging wake-up radio technology. Solidifying this framework and encouraging it in the research community will contribute to the solidification and wide adoption of the wake-up radio technology that promises to revolutionize wireless systems.
Over the few years, a new era of ultra low-power *wake-up radios* has merged, allowing data collection and actuation in an on-demand way, rather than synchronized. Due to it's on-demand paradigm, wake-up radio aims to prolong the lifetime of main sensor nodes by reducing power consumption of nodes in an *idle* state, the so-called idle-listening cost.

While the wake-up radio technology appears quite promising to extend WSN lifetimes, the technology is still in its relative infancy, with major efforts continuing on new hardware designs. While several prototypes exist and show potential in the laboratory environment, it remains difficult to experiment with these devices. Further, the exact potential of this technology on the WSN system as a whole has not been demonstrated, offering a further impediment to the path to commercialization and availability of the devices. Hence, reliable assessment tools are required to systematically characterize the performance of the wake-up radios for real-world WSN deployments. Once these tools are available, we can utilize it to provide a holistic view of the system without physically deploying the nodes.

For these reasons, as a first step, a simulation approach is often preferred over hardware prototyping to reduce cost and also to tune and simplify protocols enabling faster application and system development. This approach also allows one to investigate design trade-offs at a greater system scale and to study the impact of different environmental parameters.

These benefits motivate the recent development of simulation tools for the wake-up radio systems we concisely survey in Section 3.3. Unfortunately, the existing tools are limited to a few that either does not model the wake-up radio with sufficient accuracy or do not consider the entire stack, ignoring the WSN operating systems. Further, most of these tools are implemented using programming frameworks such as MATLAB or C++ making it incompatible for the operating systems designed for the low-power wireless devices (eg., Contiki/C, RIOT/C, and TinyOS/nesC). This requires complete re-implementation to make the executable code compatible for the final deployment.

Some contents of this chapter have been originally published as: **WaCo: A wake-up radio COOJA extension for simulating ultra low power radios**, Rajeev Piyare, Timofei Istomin, and Amy L. Murphy, In Proceedings of the 14th ACM International Conference on Embedded Wireless Systems and Networks (EWSN 2017), Uppsala, Sweden, February 2017, and have been slightly adapted for this thesis.
Contributions. In this chapter, we offer a software-only, simulation environment to enable the exploration of the potential of the wake-up radio technology. This exploration must allow for the development, evaluation, and evolution of protocols across all layers of the software stack as well as the evaluation of the applications in the target environments.

To this end, we present WaCo, a set of extensions to the COOJA simulator [198] and Contiki operating system that allows prototyping of protocols and applications for exploiting a standard WSN mote extended with a new, simulated wake-up radio hardware module. Our design allows the specification of the wake-up radio to be easily modified, allowing researchers to plug in their own wake-up radio module parameters. To the best of our knowledge, WaCo is the first open-source software framework providing these capabilities for the wake-up radio assessment. Specifically, through design of WaCo we:

- provide a new physical layer interface for the wake-up radio module.
- enable support for multiple radio channels that can be used simultaneously and independently.
- extend the power tracker module in COOJA to provide an accurate estimation of the energy consumption for dual-radios.
- provide a visual representation of the network behavior on a unified timeline that captures the events generated by multiple radio transceivers. For instance, the activities of the wake-up radio and the main data radio.
- enable the developers and researchers to emulate the actual, deployment-ready code of ContikiOS.
- allow researchers to plug in their own wake-up radio module parameters.

From an implementation standpoint, WaCo’s ability to directly reuse the firmware simulated into real-world is enabled by our dependence on COOJA and MSPSim, a hardware emulator for the MSP430 based MCUs.

Structure of this chapter. This chapter presents the relevant background and our choice for extending COOJA simulator in Section 3.1 followed by the design of WaCo and ContikiOS extensions in Section 3.2. We also offer related work in Section 3.3 followed by brief summary in Section 3.4.

3.1 Background

Before addressing the novel changes we introduce to the simulation environment, we offer background on COOJA, the simulation environment that we chose to extend in WaCo. For the ultra-low power wake-up radio, we rely on chapter 1 for its background and main characteristics.

WaCo is open-source and can be found at: https://github.com/waco-sim
3.1.1 WSN Simulation Environment: COOJA

The goal of this chapter is to provide an environment in which to explore the potential benefits of the WuR technology without requiring significant hardware investment. COOJA offers a solid environment for such testing, however, it does not support multiple radios on a single hardware platform, nor does it offer modules to simulate WuR. At the same time, COOJA directly supports the integration of multiple hardware models in the form of the platform, offering a clear place for expansion to incorporate the WuR. Further, COOJA exploits MSPSim to allow firmware designed and compiled for real hardware such as the TMote Sky platform to be directly used for simulation. MSPSim emulates the MSP430 MCU and the CC2420 transceiver at the instruction level, offering very fine-grained, low-level simulations.

COOJA also allows developers to test and run their applications using different radio models on fully emulated hardware devices, a functionality not available in other simulators such as Castalia [199] and MiXiM [200]. Further, these simulators do not run the actual operating system, protocols, and the application code, designed for a real target platform, but work at a higher level, focusing only on network related aspects.

Above the hardware, COOJA also enables simulation of systems with various operating systems designed for resource constrained devices like TinyOS and ContikiOS. For this work, we chose ContikiOS [201] as it is actively developed and used in both industry and research. ContikiOS itself contains several protocols from the application to the physical layers, as shown in Figure 3.1b and two networking stacks: Rime and uIPv6. Rime [202] is a light-weight highly-modular stack providing services such as dissemination, point-to-point AODV routing, and multipoint-to-point data collection. The uIPv6 stack is an IPv6 implementation comprising 6loWPAN adaptation sublayer, RPL IPv6 routing, TCP, UDP and CoAP.

To offer a simulation environment suitable for experimenting with the WuR hardware, we made extensions and modifications to both COOJA and ContikiOS, offering descriptions of the hardware as well as the software to exploit it.

3.2 WaCo: Enabling Support of WuR in Simulation

To enable simulations with wake-up radio technology, modifications to both, the simulator (COOJA) and the mote software (ContikiOS) are required. The simulator must be extended with a simulated wake-up radio module (WuR chip) and support for multiple radio channels that can be used simultaneously and independently. Additionally, the visual simulator plug-ins that simplify debugging and measuring system performance need to be added.

The mote software requires changes to the operating system and protocols that interface with the newly added hardware modules. For this, we modify ContikiOS, providing a new physical layer interface for the wake-up radio that uses both the new wake-up radio as well as the standard, higher power, CC2420 data radio for communication.

Next, we provide details of these hardware and software extensions that form the core of WaCo.
Our modified or added extensions are highlighted and summarized in Figure 3.1.

### 3.2.1 COOJA Platform Extensions

In COOJA, node hardware is represented as a platform description, containing models for the micro-controller, radio, sensors, memory, and other hardware components. We chose to extend the existing TMote Sky platform, as it is a popular research platform in the WSN community. Further, because COOJA integrates MSPSim, simulations exploiting the TMote platform can be executed directly with code compiled for real nodes, simplifying the overall development process. Our description of the components specific to the wake-up radio appear in three places in the COOJA framework: i) the platform, ii) the simulator core, and iii) in its plugins.

First, the emulated TMote platform of COOJA is extended with a module, represented by WuR in Figure 3.1. The module implements the hardware interface between the wake-up radio and the CPU, based on I/O ports and a memory buffer. Furthermore, it defines the operational aspects of the wake-up radio: i) initializes the WuR transceiver, ii) configures the radio parameters (transmission power, frequency, data rate), iii) manages the transmission and reception of wake-up packets, and iv) controls the WuR chip. Conceptually, this interface is similar to that of the pre-existing CC2420, with extra interrupt processing to receive an incoming wake-up signal and a transmit operation to send either an addressed unicast or broadcast signal. Additional operations enable reading and writing to the memory of the processor to communicate the destination address received or to be transmitted. While these operations cover the functionality of most wake-up radios found in the literature, our interface can easily be extended. For instance, some radios additionally support duty cycling, necessitating extensions to turn on and off the wake-up radio.

Second, in the core of COOJA, we implemented support for multiple independent radio channels to be used by multiple-transceiver platform. The extension allows simulating an
3.2. **WaCo**: Enabling Support of WuR in Simulation

An arbitrary number of radio channels, with a separate signal propagation model associated to each of them. For our extended platform we use two radio channels. Most COOJA users opt for the Multi-path Ray-tracer Medium (MRM), which is, in fact, the model we apply to the CC2420 radio in our experiments. Nevertheless, for the wake-up radio, we use the Unit Disk Graph Medium (UDGM) that allows configuring the RX success probability of the WuR medium. While UDGM is simplistic when compared to MRM, its use reflects our current goal of creating a generic model of wake-up radios. For MRM to be realistic it should be configured to match characteristics of a specific wake-up radio technology. When this technology is chosen and the characteristics are measured the channel model can be easily changed to MRM.

Third, COOJA uses a plugin architecture to extend its core functionality. Our wake-up extension affects two plugins: the *power tracker* and *timeline*. Accurate power profiling requires augmenting COOJA with a second instance of the power tracker plugin associated with the wake-up radio. Thanks to our extension, it is possible to record, for both radios, the time they spend in different states i.e., listening, transmitting, and inactive state as shown in Figure 3.2.

Moreover, efficient debugging requires a visual representation of the network behavior on a timeline. Therefore, WaCo offers a unified timeline as illustrated in Figure 3.2c that captures the events generated by both radio transceivers. In the COOJA-WaCo timeline, blue indicates transmission and green reception, with the wake-up radio trigger lasting longer than the data exchange due to the slower data rate of the wake-up radio (1 kbps) than the main data radio (250 kbps).

### 3.2.2 Contiki OS extensions

We now turn our attention to extensions for WaCo on the mote software side, as shown in Figure 3.1b. Most significantly, a new physical layer module is implemented for ContikiOS to wrap the interface of the wake-up radio. This interface is analogous to the CC2420 and supports bi-directional data communication, i.e., handling interrupts to the CPU when the node receives the signal over the wake-up channel and triggering transmission of wake-up signals, optionally specifying the destination address.

To exploit both the standard CC2420 radio as well as the wake-up radio, we also implemented a new MAC protocol, W-MAC. As this component offers a significant contribution of its own, we provide its complete description in the next chapter. Here, we simply note our choice to maintain the same interface for W-MAC as for Contiki’s default MAC protocol, thus allowing it to be directly substituted in an existing network stack without requiring any modifications to the upper routing and application layers. While our simulation results presented in chapter 4 show that this approach offers significant improvements over a system without the WuR, alternate MAC protocols that cross the MAC/routing layer boundaries, or that offer additional features could easily be substituted and tested, simply exploiting the dual radio physical layer provided by our module in combination with the CC2420.

(a) Main radio power tracker. (b) Wake-up radio power tracker.
(c) COOJA-WaCo timeline trace and radio events.

Figure 3.2 – COOJA node power tracker and timeline plugin extensions.

3.3 Related Work

Recent interest in WuRs demands simulation support to allow software systems to be designed to exploit their unique features. In [203] MATLAB has been used to evaluate the performance of two different WuRX prototypes and a preamble sampling MAC protocol. The study focused on the analytical aspects to determine the number of nodes required to achieve a fully connected multi-hop network. Moreover, the simulation software employed is not clearly described and does not provide any details related to the channel model for the radios. Another similar comparison [204] using simulation and ideal mathematical analysis relates the number of hops in the network with the related effective wake-up radio range achieved by nodes deployed in a random manner. However, this study does not provide information on how other MAC or routing protocols can be incorporated and what software environment has been used to obtain the results.

Recently, OMNET++ extensions [190] provide a lean and modular simulation model for wake-up radio based systems. Authors have compared four different duty cycling MAC protocols w.r.t. wake-up radios in a single-hop and multi-hop scenario. Similarly, the GreenCastalia [205] simulator has also been extended to simulate a power model for wake-up receivers [191]. Nevertheless, neither of these simulators consider the entire stack, ignoring the operating system and instead focusing more directly on related aspects. Moreover, in these simulators, the number of available protocols specifically targeting sensor networks is limited.

WaCo, instead, directly uses the binary, deployment-ready code, and, therefore, provides developers with the ability to move between simulated and real experiments. WaCo combines the emulation capabilities of COOJA and MSPSim with power tracking models to accurately measure the power consumption of the wake-up radio system. Moreover, it also allows simulation of multiple embedded operating systems such as Contiki and TinyOS, not supported by the previous wake-up radio simulators.
3.4 Summary

Wake-up radio technology is becoming popular as a solution to achieve on-demand communication for extending WSN lifetime. However, the tools supporting the design of the wake-up radio based systems and reproducibility of the obtained results are limited. Most of these tools are closed-source and implemented in a different programming language making it incompatible for the final WSN deployment, thus requiring complete re-implementation. This jeopardizes the accuracy achieved in the simulation to be directly compared with the real-world deployments.

WaCo minimizes this gap between simulation and real hardware by simplifying and accelerating the development of wake-up radio communication protocols for low-power WSNs. Moreover, WaCo is an open-source tool\(^2\) enabling researchers to explore the potential of the novel wake-up radio technology without needing the physical hardware. Not only WaCo quickens the development phase, but it also allows testing large-scale experiments using embedded operating systems, a feature currently lacking in most of the simulators.

\(^2\)WaCo is open-source and can be found at: https://github.com/waco-sim
Extending the battery life of sensor nodes has been one of the primary research focuses since the introduction of wireless sensor networks (WSNs). With increasing interest in the Internet of Things (IoT) paradigm, power saving solutions are increasingly critical to make numerous applications possible.

Approaches to energy savings are varied, but most focus on the software side, specifically addressing the expensive communication activities. For example, work at the Medium Access Control (MAC) layer [7] offers trade-offs in the design space of energy consumption, latency, throughput, and fairness. By duty cycling the radio in strategic ways, putting the radios to sleep when they are not needed, these protocols have the potential to significantly extend system lifetime. Nevertheless, duty cycling systems still suffer from a number of problems: i) idle listening, ii) overhearing, iii) continuous transmissions, and iv) less responsive systems due to data latency during sleep intervals.

Wake-up radio technology promises to directly combat these problems by offering power consumption orders of magnitude less than the typical low power radios used in WSNs. To offer an example, the ultra-low-power radio we consider in this work consumes $1.83 \mu W$ when listening to the channel, while the CC2420 consumes $65.3 \, mW$. While the wake-up technology has been steadily evolving over the last decade in the hardware arena, only a few protocols have been developed to exploit its capability. The benefits of incorporating wake-up radios with MAC protocols in WSN is enormous, but the effect of the same on real-life deployment is still unknown and unaddressed.

**Contributions.** To illustrate concretely that the wake-up radio approach leads to significant gains in many scenarios w.r.t duty cycling protocols, we resort to our extended simulator WaCo, outlined in chapter 3. Using WaCo, we provide an evaluation of a data collection system with standard data collection protocols: Contiki Collect and RPL over a newly developed wake-up radio MAC, W-MAC with two primary goals:

---

Some contents of this chapter have been originally published as: **WaCo: A wake-up radio COOJA extension for simulating ultra low power radios**, Rajeev Piyare, Timofei Istinin, and Amy L. Murphy, In Proceedings of the 14th ACM International Conference on Embedded Wireless Systems and Networks (EWSN 2017), Uppsala, Sweden, February 2017, and have been slightly adapted for this thesis.
Chapter 4. Exploiting WaCo: A Data Collection Case Study

- to demonstrate the benefits of the wake-up radio technology and
- to show effectiveness of WaCo at providing this evaluation.

In this chapter, we focus on one of the approaches of utilizing wake-up receivers, i.e., an always-on mode that allows triggering the main node almost instantly, thereby reducing data latency. This mode provides promising results as we study next in this chapter. Specifically, in this chapter, we:

- propose a low-complexity MAC protocol W-MAC, which exploits the dual-radio interface: the wake-up radio and the main data radio.
- evaluate the performance of networks integrated with wake-up radio over two different data collection protocols. By evaluating different network stacks, we pushed the envelope of energy efficiency achievable using low-power wake-up radios.
- test the scalability of the wake-up radio based system using different network sizes and data rates.
- show that W-MAC outperforms other state-of-the-art duty cycling techniques in terms of energy consumption, reliability, and throughput.
- demonstrate that wake-up radio can reduce the end-to-end data latency by the factor of 200, while still meeting the needs of several representative WSN applications.
- demonstrate that WaCo provides an effective environment for analysis of protocols proposed for the wake-up radio technology.

Structure of this chapter. In Section 4.1, we provide a novel MAC module, called W-MAC which uses the wake-up radio as a trigger for the standard CC2420 radio, and offers the same interface as other popular MAC protocols, allowing it to easily sit below standard routing protocols. This chapter then provides a proof-of-concept and the advantages of the wake-up radio when combined with full networking stacks in Section 4.3. The results show that W-MAC combined with Contiki Collect satisfies realistic workloads offering up to $7.8 \times$ higher reliability, a reduced latency of up to $23 \times$, and energy savings up to $20 \times$ that of ContikiMAC. These savings are even higher when the network is evaluated using RPL. These results simultaneously confirm that wake-up technology has tremendous potential and that our simulator extension provides an effective mechanism for such exploration. Section 4.4 offers a brief overview of the related work followed by concluding remarks and future directions for this research in Section 4.4.

4.1 W-MAC in a Nutshell

This section focuses on the wake-up radio MAC layer which is the extension of the software stack of ContikiOS.

W-MAC has been implemented to offer the same interface to higher layer protocols as the popular ContikiMAC, making it seamlessly interoperable. The major contribution of W-MAC is its coordination of the dual radios offered by WaCo, using an always-on WuRx as a trigger for activating the CC2420. W-MAC offers both communication primitives, unicast for selective
triggering and broadcast for network wide wake-up as described next.

**Unicast Communication.** W-MAC assumes all network nodes host both radios, naturally forming a multi-hop network in which nodes alternately act as senders and receivers. Further, W-MAC is a sender-initiated protocol in which the message source triggers the receiver to wake up.

For simplicity we illustrate the basic operation of W-MAC through an example with the 2-hop network shown in Figure 4.1 while the wake-up radio communication process is described in Algorithm 1. In this example, a message is originated from A, sent to B then to the destination C. When a node has data to send, either generated from the upper layers of the protocol stack or forwarded by another node, W-MAC first transmits a WuS containing the address of the destination node. It should be noted that the two networking stacks of ContikiOS use different layer 2 address sizes, specifically, the Rime stack uses 2 bytes while uIPv6 requires 8 bytes. W-MAC allows exchange of these different network address sizes (node MAC addresses), with corresponding consequences on the energy costs due to increased transmission and demodulation times which we evaluate in-depth in Section 4.3.1.1.

On the WuRx side, the receipt of the WuS matching the receiver’s address triggers the activation of the CC2420 in receive mode. If no data packet is received within a predefined time due to interference or collisions, the receiver switches the CC2420 back into sleep mode, keeping the WuRx actively listening for subsequent signals. Instead, if a packet is received on the CC2420, an acknowledgment is sent, then the receiver’s primary radio is turned off. Upon receipt of the acknowledgment, the transmitter’s primary radio is similarly turned off. On the sender side, after the WuS is transmitted, the node waits for a short period of time during which it expects the receiver’s CC2420 to be switched into receive mode. It then transmits the data on the main, data radio, switches into receive mode to receive the ACK, then goes back to sleep. If the sender does not receive the acknowledgement within a certain time interval, it will return to the beginning of the sequence, re-transmitting the WuS.
Chapter 4. Exploiting WaCo: A Data Collection Case Study

Algorithm 1: Wake-up radio based communication process

Initialization;
for \( i = 1 \rightarrow N \) do
  if \( i \) has new data then
    send wake-up trigger;
    after wake-up sync delay start CSMA/CA process;
    transmit the data;
  else if \( i \) is woken up by wake-up trigger then
    turn on the main radio;
    receive the data packet;
    send the ACK;
  else
    buffer the data;
end

Broadcast Communication. Thus far we have only discussed unicast, addressed transmission, however, the WuR module also supports unacknowledged broadcast. In this case, the receipt of the WuS causes all nodes to switch on their primary radios. As before, the sender waits a short period, then transmits the data. However, unlike unicast mode, broadcast transmissions are not acknowledged, allowing the sender to immediately switch off the CC2420 after transmission and the receiver to switch it off after receipt.

Channel Configuration. As is typically done, the WuR and the main radio use different channels, eliminating the possibility of collisions between the wake-up signal and the data packets. However, collisions can still occur between concurrent wake-up signals or concurrent data transmissions. In our current implementation, channel sensing is only performed by the main radio before transmission, allowing us to avoid most data packet collisions. For this, we use Contiki’s default Carrier-Sense Medium Access (CSMA) mechanism and if the data channel is busy due to an on-going transmission or reception, the transmitting node backs off for a random period before retransmitting the WuS. The drawback of performing carrier sense just before data transmission is the penalty of increasing the on-time for the main radio at the receiver, which was awakened by the WuS, but due to the data channel being busy, the data transmission cannot proceed. To overcome this, an alternative solution could be to use WuR for channel sensing and reservation rather than the main radio [113]. We plan to investigate the benefits of this in the future.

4.2 WaCo Evaluation Settings

To evaluate WaCo, we demonstrate its use through the simulation of a data collection application. This is a common scenario for low-power wireless sensor networks in which high data yield and energy efficiency are paramount. Before offering the details of our experimental
setup, we recall that our objective is to answer the following two questions: i) to what extent can battery-powered networks be made energy efficient by equipping them with WuR technology and ii) can WaCo be used to demonstrate this. In this section, we describe our simulation settings, starting with the lowest physical network topology and progressing up through the routing and application layer.

4.2.1 Network Topology

To avoid the bias of varying network densities, we established a 10 by 10 grid of 100 nodes. The horizontal and vertical distance between nodes is 30 m and we place a single sink node close to the center of the grid as illustrated in Fig. 4.2.

4.2.2 Routing Protocols

The architecture of WaCo and W-MAC allows us to run different routing protocols, as long as they utilize the standard MAC interface. For this work, we experiment with two routing protocols provided with ContikiOS: Contiki Collect and the IPv6 Routing Protocol for Low-Power and Lossy Networks (RPL) [206]. Together they represent the current state-of-the-art in data collection and are staple references for many-to-one scenarios.

Contiki Collect offers an address-free data collection protocol that provides a way for source nodes to forward data packets towards a sink. Nodes in a Contiki Collect network periodically broadcast beacons that announce their distance from the sink node. To send a data packet towards the sink, nodes choose a parent node that is closer to the sink than itself. Contiki Collect uses unicast for all data transmissions and broadcast for route discovery. It employs an expected transmissions (ETX) based route selection technique that minimizes the number of packet transmissions to reach the root. Contiki Collect is part of the Rime stack that supports both single-hop and multi-hop communication primitives.
Chapter 4. Exploiting WaCo: A Data Collection Case Study

Figure 4.3 – MAC protocols compared against W-MAC in WaCo.

**RPL** is the standardized routing protocol for IPv6-based LLNs proposed by the IETF RoLL working group [206]. RPL is designed for networks with significantly higher packet loss rates than those assumed by existing routing protocols such as collection tree protocol [207]. While RPL is optimized for many-to-one (collection) traffic patterns, it also provides mechanisms for point-to-multipoint and point-to-point traffic. Being optimized for low-speed links, RPL focuses on maintaining low control plane overhead.

### 4.2.3 MAC Protocols

The routing protocols described above sit on top of the MAC layer, whose primary objective is to control usage of the radio, including duty cycling. W-MAC provides the same, standard interface as ContikiMAC [8] and NullRDC, making it trivial to experiment with the alternate stacks by simply changing the MAC protocol in use. The most significant difference between W-MAC and the others is its ability to exploit the WuR, while the others use only the CC2420. All the three different MAC protocols compared in WaCo are illustrated in Figure. 4.3.

**ContikiMAC** is the de-facto, asynchronous protocol for ContikiOS. It uses the data packet itself as preamble and has a phase-lock mechanism, where senders record their neighbor’s wake-up phase, using this information to make subsequent transmissions cheaper for the sender. In addition, ContikiMAC performs fast sleep optimization allowing receivers to quickly detect false positive wake-ups. For ContikiMAC the sleep interval was set to 125 ms with phase lock-mechanism enabled.

**NullRDC** serves as a baseline for our comparison. When using it, the main radio transceivers are always kept on i.e., no duty cycling is applied.

### 4.2.4 Simulation Parameters and Metrics

The protocols utilize throughout our evaluation are highly customizable with parameters such as buffer sizes, timeouts, retries, and maximum hop count. Wherever possible, we used the default values, and all parameters are reported in Table 4.1.

In each of our simulation scenarios, every source node generates 6 byte data packets at a fixed inter packet interval (IPI). Each run simulates 40 minutes of runtime with the first 10 minutes
4.2. WaCo Evaluation Settings

Table 4.1 – Simulation Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of Nodes</td>
<td>100</td>
</tr>
<tr>
<td>Simulated Node Type</td>
<td>Tmote Sky, Radio Chip: CC2420</td>
</tr>
<tr>
<td>Distance Between Nodes</td>
<td>30 m</td>
</tr>
<tr>
<td>Topology</td>
<td>Square Grid, Sink Location: Near Center</td>
</tr>
<tr>
<td>Radio Mediums</td>
<td>MRM for Main Radio, UDGM for WuR</td>
</tr>
<tr>
<td>MRM Settings</td>
<td>Noise Mean: -80 dBm, Noise Variance: 4</td>
</tr>
<tr>
<td>UDGM Settings</td>
<td>Tx Range: 50 m, Interference Range: 50 m, Success Ratio: 100%</td>
</tr>
<tr>
<td>Transmission Range</td>
<td>Main Radio: ~65 m, WuR: 50 m</td>
</tr>
<tr>
<td>Routing</td>
<td>Contiki Collect and RPL</td>
</tr>
<tr>
<td>MAC</td>
<td>CSMA</td>
</tr>
<tr>
<td>Radio Duty Cycling</td>
<td>ContikiMAC and NullRDC</td>
</tr>
<tr>
<td>Sleep Interval</td>
<td>ContikiMAC: 125 ms and NullRDC: 0 ms</td>
</tr>
<tr>
<td>Iterations</td>
<td>5 for each parameter permutation</td>
</tr>
<tr>
<td>Random Seeds</td>
<td>New seed for each iteration</td>
</tr>
<tr>
<td>Simulation Duration</td>
<td>10 mins settling, 30 mins of actual</td>
</tr>
<tr>
<td>Inter Packet Interval</td>
<td>[10, 30, 100, 300, 600, 1800] s</td>
</tr>
<tr>
<td>Bit Rate</td>
<td>Main Radio: 250 kbps, WuR: [1, 5, 100] kbps</td>
</tr>
<tr>
<td>WuR Message Size</td>
<td>Rime: 2 bytes, uIPv6: 8 bytes</td>
</tr>
<tr>
<td>Application Layer Payload Size</td>
<td>6 bytes</td>
</tr>
<tr>
<td>Full stack Payload Size</td>
<td>43 bytes</td>
</tr>
<tr>
<td>ACK Packet Length</td>
<td>5 bytes</td>
</tr>
</tbody>
</table>

serving as a burn-in time for the routing protocols to establish routes and for ContikiMAC to establish the phase locks for neighbors. The subsequent 30 minutes are analyzed and results reported for i) power consumption, ii) reliability, and iii) end-to-end packet latency.

**Power consumption** is computed using the amount of time a node keeps its radio on in different states. When multiplied by the consumption of the radio in these states, given in Table 4.2, the result is a measure of energy efficiency. While this mechanism considers only the consumption of the radio, ignoring components such as computation and memory, it remains a reasonable proxy for overall power consumption because the power profile of a typical sensor platform is dominated by the radio. We measure the radio-on timings for both the CC2420 and WuR radios in software, using WaCo’s extended PowerTracker to accommodate the wake-up radio.

**Reliability** is defined as the fraction of application data packets successfully received at the sink over those sent by all the data sources. This is an indication of the level of service provided to applications in delivering sensed data. It is averaged over multiple rounds and represented as a ratio.

**Latency** is defined as the end-to-end packet delivery time from the time of generation of the data packet to reception.

Each point on our plots in the Section 4.3, represents the network-wide average for five simulated runs.
Table 4.2 – Power consumption of the CC2420 and WuR measured in-lab. Idle reflects listening to the channel, but not actively receiving.

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Mode</th>
<th>Power Consumption (mW)</th>
</tr>
</thead>
<tbody>
<tr>
<td>CC2420</td>
<td>RX</td>
<td>65.3</td>
</tr>
<tr>
<td></td>
<td>TX</td>
<td>71.9</td>
</tr>
<tr>
<td></td>
<td>Idle</td>
<td>65.3</td>
</tr>
<tr>
<td>WuR</td>
<td>RX</td>
<td>0.34</td>
</tr>
<tr>
<td></td>
<td>TX</td>
<td>46</td>
</tr>
<tr>
<td></td>
<td>Idle</td>
<td>0.00183</td>
</tr>
</tbody>
</table>

4.2.5 WuR Hardware Parameters

For evaluation purposes, the WuR solution that we adopted is a custom designed ultra low-power module [208]. This is a typical WuR, representative of most wake-up radios found in the literature. Its performance parameters are set to values obtained from actual, in-lab measurements. The WuR board includes two modules: the addressable WuRx and a WuTx. The WuRx module operates in the ISM 868 MHz band and has a receiver sensitivity of -55 dBm with a maximum communication range of 50 m. The power consumption of the WuRx is $340 \mu W$ in receiving mode and of $1.83 \mu W$ when it is in an idle state, listening for the signal. The power consumption of the WuTx is measured to be 46 mW when transmitting at +10 dBm. WuR signals are sent at data rates of 1, 5, and 100 kbps, and consists of 2 bytes of address data for Rime and 8 bytes for uIPv6.

While the experiments presented here reflect the specific properties of our prototype WuR, the results can be generalized to the class of WuR. Further, as new technologies develop, WaCo can be easily extended with the hardware parameters of other modules.

4.3 WaCo for Network Performance Analysis

WaCo enables evaluation of systems with wake-up radios in two ways. First, W-MAC, as a part of WaCo, can be seamlessly substituted for the stock ContikiMAC in a standard stack as it offers the same MAC-layer interface. Further, other MAC protocols employing wake-up radios or cross-layer solutions can be evaluated in WaCo by exploiting its WuR model and the plugins for power profiling and visual debugging.

Here, we concretely demonstrate how WaCo can be used to investigate the performance gains provided by W-MAC when integrated in an networking stack. We concentrate on the many-to-one data collection scenario, which is representative of a significant fraction of existing applications for low-power wireless.

4.3.1 Contiki Collect with Wake-up Radio

We next explore how well the Contiki Collect performs over the W-MAC-layer. Since W-MAC offers the same MAC-layer interface, executing Contiki Collect over W-MAC is largely a matter of changing network configuration settings in ContikiOS during compile time.
We first focus on the evaluation of Contiki Collect in combination with wake-up radio, then discuss the performance differences of the uIPv6 stack with RPL in Section 4.3.2. We also offer insight into the benefits of the addressable mode in Section 4.3.3 by showing simulations that use only broadcast wake-up signals.

### 4.3.1.1 Impact of varying network traffic and WuR bit rate

For this test we varied the network load to determine how the choice of a MAC protocol affects the network capacity.

**Setup:** We explored a range of inter-packet intervals (IPI) from 10 s to 1800 s, with the lower IPIs showing the performance in a highly stressed setting. Together with this, the wake-up radio bit rate was also varied from 1, 5, to 100 kbps to study the trade-off of different data rates on the performance of W-MAC w.r.t. to the state-of-the-art MAC protocols. We also experiment with no data traffic, i.e., zero-IPI, observing the network without any application load to offer a baseline for the maximum power savings achievable. For this evaluation, we fixed the network size to be 100, where 99 are source nodes and a single sink.

**Results.**

**Reliability:** We begin our evaluation by considering the data collection reliability as shown in Figure 4.4a. At the smallest IPI of 10 s, i.e., the highest traffic load, all the protocols suffer from overload showing reliability below 75%. As expected, ContikiMAC demonstrates the smallest network capacity due to its duty-cycling nature. The nodes only have one chance to receive a single packet in each sleep interval, configured as 125 ms in our simulations. NullRDC, instead, is on the other end of the spectrum, showing the highest capacity. This is due to the fact that nodes utilizing NullRDC keep their main radio always-on having a higher probability of receiving the data packets as illustrated in Figure 4.3(c). W-MAC, on the other hand, stands in the middle. At higher bit rates, W-MAC is able to push more data packets through as the wake-up signaling delays (0.16 ms) are much smaller than the periodic channel check rate of ContikiMAC (125 ms). When the WuR data rate is reduced to 1 kbps (WuTX time of 16 ms), the probability of collisions among WuR transmissions increases as the channel is occupied for a longer period of time for signaling. This directly results in a reduced successful packet transmission.

In overload conditions, the queues overflow and packets are dropped. As we decrease traffic with higher IPIs, all MAC protocols achieve near perfect reliability. Notably, each MAC protocol achieves perfect reliability at a different IPI, reflecting the ability to handle higher bandwidth due to the different times required to transmit each packet. W-MAC at higher data rate improves the network capacity over ContikiMAC, handling the IPI of 100 s, while the latter is able to deliver all packets only at 300 s IPI.

**Data latency:** One of the main advantages of utilizing wake-up radios is that it reduces data latency in contrast to duty cycling MACs. Next, we look at the average data latency for varying IPI and data rate as illustrated in Figure 4.4b. NullRDC achieves the lowest network latency.
as packets are tightly packed into the available bandwidth with no additional delays besides CSMA backoffs. W-MAC stands in the middle, albeit closer to NullRDC, due to delays imposed by the signaling of the WuR and the activation of the data radio. The time required to transmit a 16-bit wake-up packet at a data rate of 1 kbps is 16 ms, whereas this duration is reduced to only 3.2 ms and 0.16 ms at the data rate of 5 kbps and 100 kbps, respectively. However, these delays are much smaller than the channel check rate of ContikiMAC, even at the lowest bit rate allowing it to push more packets through. The high traffic load negatively affects latency as higher contention means longer backoffs and longer packet queues.

As we decrease traffic with higher IPIs, all MAC protocols achieve the latencies in line with those of NullRDC. At low load (≥ 300 s), latencies of all three protocols become very similar.

**Power consumption:** We now turn our attention to the power consumption shown in Figure 4.4c. As NullRDC keeps the main radio transceiver always-on, its power consumption is several orders of magnitude higher than the others, averaging 65 mW. Therefore, our plots only show power consumption for W-MAC and ContikiMAC. We also offer two lines showing the consumption for zero-IPI, or no data (ContikiMAC_no-data & W-MAC_no-data). This concretely shows the baseline consumption required to maintain the collection topology, separating out the additional cost to transmit data.

Most importantly, we note that in all scenarios, the addition of the WuR yields significant savings, as summarized in Table 4.3. With no data, W-MACs consumption drops dramatically from 1.01 mW to 0.1 mW. In the setting when both protocols achieve high reliability, namely the IPI of 300 s, the data traffic of ContikiMAC pushes consumption above the baseline to 1.6 mW while W-MAC is very close to its baseline. The reason for this significant difference is that W-MAC reduces idle listening of the power-hungry main radio and does not need long packet trains to deliver the packet as in duty-cycling MAC protocols, thus achieving short transmission times. While ContikiMAC does mitigate this by starting the packet train close to the end of the sleep interval of the destination node, this estimation is not perfect. Instead, the signaling of the WuR ensures tight timing, except in overloaded scenarios. When the system is overloaded, both protocols show increased power consumption because of channel contention that leads to collisions and retransmissions. As the IPI decreases the effect of data traffic gradually reduces until the system lifetime is dominated by protocol overhead.
To understand better how W-MAC achieves such low-power consumption, Figure 4.5 captures the per-node main radio duty cycle (RDC) for the network of 100 nodes at IPI=300 s when both the protocols achieve near perfect reliability. The overall average duty cycle of all the sensing nodes is 2.57% and 0.139% for ContikiMAC and W-MAC, respectively. Looking at the individual nodes, the sink, Node 1 had by far the highest main radio uptime of almost 4.66% for ContikiMAC and 0.39% when using W-MAC. This was expected since, the sink had to process the data of the whole network.

As it can be observed, the RDC of nodes employing W-MAC is more balanced across the entire network whether the node is closer or further away from the sink w.r.t. to ContikiMAC where it fluctuates between 1.5% to 4.6%. Even at the per-node level, W-MAC provides a significant reduction in radio on times, directly translating to less communication cost. For instance, we take an in-depth look at the two sender nodes employing ContikiMAC, Node 65 and Node 74 with highest duty cycles. Node 65 exhibits a radio uptime of 4.29% while Node 74 has 4.42%. Comparing these values with W-MAC, Node 65 has radio uptime of 0.16% and Node 74 with 0.18%, a reduction in main radio activity of ≈96%. This has a key advantage as it increases the node lifetime and hence, ensures that network coverage can be maintained longer.

**Takeaway:** The results show that W-MAC satisfies realistic workloads offering up to $7.8 \times$ higher reliability, a reduced latency of up to $23 \times$, and energy savings up to $20 \times$ that of ContikiMAC. Even at the lowest bit rate, W-MAC outperforms duty-cycling ContikiMAC providing better energy efficiency and network performance. However, a trade-off between wake-up signal duration, power consumption, and coverage needs to be considered while choosing which data rate to adopt.

### 4.3.1.2 Scalability

Scalability is an important characteristic for wireless sensor networks to gauge how it affects the overall network performance. With the proliferation of IoT devices, for scenarios such as smart farms or cities, networks over hundreds of nodes are envisioned. Next, using WaCo we test the scalability of the wake-up radio based system, specifically to estimate the reliability.
Chapter 4. Exploiting WaCo: A Data Collection Case Study

Figure 4.6 – Scalability analysis of WuR based network.

(a) Reliability.  (b) Latency.  (c) Radio duty cycle ratio.

**Setup:** For this evaluation, the number of source nodes in the network was varied from 9 to 100 together with the network traffic i.e., IPIs. The wake-up radio signals are transmitted at 100 kbps and contain 16-bits of address data. The radio on timings for the main radio is obtained using WaCo’s power tracker. Here we only present the results for W-MAC only for 30 minute runs as illustrated in Figure 4.6.

**Results:** First, it can be observed that with a small network of up to 25 nodes, W-MAC is able to achieve reliability close to 100% along all traffic loads. Together with this, the data collection latency also remains below 4 s as there is less competition for the channel and the packets are pushed through without any extra delays apart from the wake-up radio signaling. As network size increases, the reliability of data collection decreases at lower IPIs, as a consequence of more nodes competing for the channel. On the other hand, for lighter traffic loads, i.e., IPI ≥300 s, the reliability remains perfect even with more number of nodes. As expected, the data latency and the radio duty cycle increases with the increasing number of source nodes.

**Takeway:** W-MAC maintains high network performance with tens to hundreds of nodes. For applications that do not demand strict latency but require high system reliability, longer packet intervals should be considered.

### 4.3.1.3 Effect of noise floor

In this section, we study the impact of interference and background noise on the performance of the network. WaCo allows us to explore various noise parameters, which would otherwise not be possible to replicate in real deployments. For instance, using different ambient noise levels makes it feasible to study its impact on the network topology, a useful insight toward the real deployment [209].

**Setup:** As we do not have experimental traces for the noise in a smart city environment or from the WuR testbed, we resort to the multi-path ray tracing model (MRM) provided by WaCo and commonly used in literature. MRM is also the channel model utilized by the main
4.3. **WaCo for Network Performance Analysis**

![Graphs showing reliability, latency, and radio duty cycle ratio.](image)

Figure 4.7 – Effect of noise floor on network performance.

**radio (CC2420) in WaCo.** It models physical layer properties such as background noise and interference through signal-to-interference-and-noise ratio (SINR), the capture, and multipath effects. Moreover, it also allows configuring transmission power, receiver sensitivity, and the antenna gain.

In our evaluation, we varied the mean background noise from -76 to -90 dBm while the standard deviation was kept to 4. As previously studied in [210], the mean noise in an indoor testbed environment can vary from -90 to -98 dBm and a standard deviation of 2-4 dBm. While these values can reach -75 dBm in the cities with the standard deviation between 0-10 dBm. The wake-up radio data rate is fixed to 100 kbps with a network size of 100 nodes.

**Results:** Figure 4.7 shows the protocol performance as a function of noise floor for W-MAC. From a reliability perspective, W-MAC performs well under low noise situations. Below -76 dBm, W-MAC benefits from the improved radio range and link quality making packet delivery more efficient, reaching perfect reliability at low traffic rates. As the noise increases, the theoretical communication range changes and so does the network topology. This decreases the number of neighboring nodes, resulting in a sparse network with poor links thus reducing the probability of successful forwarders of data packets towards the sink.

Similarly, W-MAC suffers from high data latency at high noise levels. A similar trend is also observed for main radio activity. Due to unsuccessful packet acknowledgments, the number of retransmissions increases yielding an increase in energy expenditure.

**Takeway:** In simulations, W-MAC performs reasonably well under low noise conditions. Further investigation is, however, required to experimentally evaluate the capabilities of the WuR system under real noise environments.

4.3.2 **RPL with Wake-up Radio**

To demonstrate the capabilities of WaCo, we also offer an evaluation of IPv6 Routing Protocol for Low-Power and Lossy Networks (RPL), run on top of W-MAC, ContikiMAC, and NullRDC as shown in Figure 4.8.
Table 4.3 – Summary of network-wide performance gain of W-MAC over ContikiMAC. W-MAC offers higher packet delivery ratio, lower energy consumption, and lower end-to-end latency. W-MAC performance meets or exceeds the widely used ContikiMAC link layer.

<table>
<thead>
<tr>
<th>Network Stack</th>
<th>Power [mW]</th>
<th>Reliability</th>
<th>Latency [s]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Contiki Collect</td>
<td>4.75–20.40×</td>
<td>1.00–7.85×</td>
<td>1.45–23.94×</td>
</tr>
<tr>
<td>RPL</td>
<td>3.00–8.40×</td>
<td>1.00–3.69×</td>
<td>9.18–228.50×</td>
</tr>
</tbody>
</table>

**Setup:** We varied the network traffic rate from 10 s to 1800 s together with the wake-up radio bit rate while keeping the network size fixed to grid topology with 100 nodes.

**Results:** Interestingly, W-MAC configured at higher data rate achieved better network performance than that of Contiki Collect. The reliability recorded for all three MAC protocols is higher at the respective IPIs, moreover ContikiMAC shows perfect reliability starting from the IPI of 100 s. However, reliability of W-MAC at low bit rates, i.e., 1-5 kbps is worse than that of Contiki Collect. Even at low traffic, the reliability reaches only up to 75% while perfect reliability is achieved with Contiki Collect. This is attributed to the fact that the wake-up address for nodes in RPL is by default 8 bytes, which is 4× longer than the Rime node address. Therefore, the wake-up radio spends more time in transmitting these addresses due to longer bit duration. As such, the channel is occupied for a longer period of time causing higher number of packet collisions. With a shorter transmission time (higher WuR bit rate), the busy period shrinks and the number of successful transmissions increase improving network reliability.

Similarly, the latency and power consumption in absolute terms are lower than those of Contiki Collect. However, in relative terms W-MAC achieves less improvement over ContikiMAC. This is also due to the longer network address size in RPL w.r.t. Contiki Collect. On average the power consumed by wake-up radios is 1.1× more than that of Contiki Collect in the no-data scenario. In case of network scalability and impact of noise, similar trend to that of Contiki Collect has been observed, therefore, we do not present the results here.

112
4.3. WaCo for Network Performance Analysis

Figure 4.9 – Comparison of addressing and broadcast mode of the wake-up radio.

**Takeway:** As observed from the evaluation, W-MAC over RPL provides better network performance than ContikiMAC. However, slow data rate for wake-up radio is not favorable in this case and may jeopardize the overall network reliability due to high rate of packet collisions.

### 4.3.3 Addressable vs. Broadcast Mode

To further test WaCo, we ran an additional set of simulations with W-MAC, but turning off the addressing mode of the wake-up receiver. Such mode known as broadcasting is often utilized for network flooding or data dissemination in wireless sensor networks.

**Setup:** In this setting, the network broadcast address is contained in the wake-up signal packet, and after the WuRXs receive and decode this address, they subsequently turn on the main radio to receive the packet. For this evaluation, the network size is fixed to 100 nodes.

**Results:** As expected, our analysis shows that the latency and reliability remains the same for addressing and broadcast modes, therefore, we do not show the results. Instead, power consumption increases with broadcast mode only, as shown in Figure 4.9 for both Contiki Collect and RPL. This is also expected, as the act of turning on the main radio, even for a short duration, adds to overall consumption. By comparing these values to Figures 4.4c and 4.8c, we see that the WuR with broadcast still shows savings over a system without the WuR, however, the difference is not as significant.

**Takeway:** W-MAC provides the same level of performance in broadcast mode w.r.t. addressing mode. However, this performance comes at a high energy cost as all devices that receive the wake-up signal trigger their main radios for data exchange.
Chapter 4. Exploiting WaCo: A Data Collection Case Study

4.4 Conclusions

The goals of this chapter were two fold: i) offer a novel MAC protocol that exploits dual radio interfaces and ii) demonstrate the potential of the novel wake-up radio technology using our developed tool, WaCo.

Toward the first goal, we proposed a low-complexity wake-up radio MAC that allows building data collection networks using a standard network stack. To test the performance of the W-MAC and how well it performs in-combination with the upper stacks, we conducted two separate many-to-one data collection campaigns in WaCo using Contiki Collect and RPL. The simulation results of Contiki Collect in combination with wake-up radio indicate a decrease in main radio on-time by 20×, a reduced end-to-end data latency of 23×, and 7.8× higher network reliability than that of ContikiMAC. Even at the lowest bit rate, wake-up radio system outperforms duty-cycling ContikiMAC providing greater channel efficiency. Further, the evaluations also indicate that wake-up radio network scales well from ten to hundreds of nodes, without sacrificing network performance. On the other hand, W-MAC over RPL provides better performance than Contiki Collect. However, the slow data rate of wake-up radio is not favorable in this case and may jeopardize the overall network performance. These results show that the wake-up radio technology has the potential to offer significant energy saving without compromising on reliability and latency. Actually, in terms of reliability, due to the brief transmission duration fostered by using the wake-up radio as a trigger for a higher power radio, systems can support higher throughput.

For the latter, we have demonstrated that WaCo provides an effective environment for the analysis of protocols proposed for WuR technology. We demonstrated this through the use of a standard protocol stack, and also by showing how different features of the wake-up radio such as broadcast vs. addressing mode and the data rate can affect that performance.

The enormous potential demonstrated by the WaCo motivates further research efforts in this direction. Future work will involve further testing and validation of the WaCo environment with different wake-up radio hardware specifications from the literature, the development of novel MAC and routing protocols, possibly incorporating an acknowledgment mechanism at the wake-up radio level, and cross-layer protocols to better exploit the features of this technology through the routing and application layers.
Wake-up Lab: Implementing a Wake-up Radio Testbed

So far the most common method used by researchers to evaluate and benchmark the performance of wake-up radio based systems has been computer simulations [211, 212, 144]. Simulators present several advantages among which perfect repeatability, complete automation, and easy setup are the most notable ones allowing results to be compared against other benchmarks. In addition, cost-effective long term evaluations can be conducted and the simulation setups can be easily re-used by the other developers. Nevertheless, as outlined in chapter 2, simulations have been criticized in the WSN community for not being able to capture and model the dynamics and the complexity of the test environment such as wireless propagation and interference and its inability to correctly model the hardware level details such as delays, errors, and power consumption [186]. These limitations are causing inconsistency between the simulated scenario and the real deployments.

In contrast, testbeds play a critical role in providing an efficient and effective alternative to debug and evaluate protocols and applications. Testbeds are sets of permanently deployed sensor nodes connected to a backbone infrastructure which handles reprogramming and data logging to a database. Usually, testbeds provide a web interface through which users can easily create, schedule experiments, and extract the metrics of interest. Testbeds bring several advantages when benchmarking wireless sensor network applications: the experiments can be performed and debugged in a controlled, yet realistic conditions, a prerequisite for flagship conferences such as ACM SenSys, IPSN, and EWSN. Some testbeds also provide fine-grained power profiling [193] and timing information [213]. This accelerates the deployment and evaluation of proposed wireless protocols and applications before the final system is deployed.

While there are a number of existing sensor network testbeds such as the FIT IoT-LAB [192], FlockLab [193], and Indriya [194], none of these are equipped with wake-up radio functionality. This has hampered the characterization and evaluation of the wake-up radio systems under realistic distributed environment.

Contributions. This chapter makes a step forward in this direction by providing Wake-up Lab, a modular dual-radio prototype that can be easily integrated into a testbed for evaluation of wake-up radio technology. Specifically, in this chapter, we:
Chapter 5. Wake-up Lab: Implementing a Wake-up Radio Testbed

- integrate a cutting edge wake-up radio module provided by our collaborators at ETH Zürich to a Tmote Sky node forming a dual-radio prototype\(^1\).
- enable selective activation of the nodes that only need to take part in data collection.
- enable bi-directional communication over wake-up radio, a key characteristic for achieving multi-hop communication.
- detail the software and the hardware integration of the dual-radio prototype that forms the *Wake-up Lab*.

Structure of this Chapter. The rest of this chapter is organized as follows: We present the hardware integration in Section 5.1 followed by the software design for controlling the dual-radio platform in section 5.2 and its usage in section 5.3. Section 5.4 summarizes this chapter outlining the future directions.

5.1 Hardware Integration: Wake-up radio + Tmote Sky

In this section, we offer a description of the low-level integration of different hardware solutions than going up the stack in section 5.2. As the wake-up radio opted in this work is not a high-complexity radio transceiver such as CC2420, first, we had to integrate it with a standard sensor mote Tmote Sky acting as the primary data radio and for controlling the secondary wake-up radio. This mainly involved providing interconnections across the two independent microcontrollers: MSP430F1611 microcontroller on the Tmote Sky running the ContikiOS stack and the PIC16 for managing the wake-up radio driver. We next describe the main features of the adopted wake-up radio module followed by its integration with the Tmote.

5.1.1 Wake-up Radio module

To compose a modular platform, we adopted a wake-up radio hardware that has been realized in collaboration with ETH Zürich. One of the main features of this module w.r.t the prototypes surveyed in chapter 1 is its ability to selectively receive and transmit wake-up signals. The present wake-up radios have a shorter communication range than the high power radios, making it difficult to align coverage of these two radios. Hence, the bi-directional communication allows creating a multi-hop network in order to remotely trigger the distant nodes. Figure 5.1 illustrates the wake-up radio architecture, where each module consists of an always-on wake-up receiver and a transmitter provisioned with an 8-bit ultra-low power PIC16LF1824T39A microcontroller. The microcontroller is dedicated: \(i)\) for executing the wake-up radio driver, i.e., transmission and reception and \(ii)\) for selective triggering by decoding an address embedded in the RF carrier. This functionality provided by the wake-up radio module is fundamental to prevent false wake-ups and to reduce wasteful power consumption.

The wake-up receiver design consumes on the order of micro-watts, i.e., \(1.83 \mu W\) in the standby mode when listening for the wake-up channel, thereby allowing always-on operation.

---

\(^1\)The content of this chapter is a joint work with Michele Magno and Luca Benini from the Department of Information Technology and Electrical Engineering (IIS), ETH Zürich, Switzerland.

116
5.1. Hardware Integration: Wake-up radio + Tmote Sky

While receiving and decoding the address bits by the PIC16 microcontroller, the power dissipation increases to 340 µW. This wake-up receiver supports a maximum bit rate of 1 kbps using on-off keying (OOK) modulation and operates in the 868 MHz band.

On the transmitter side, the fact that the PIC16LF1824T39A contains an integrated low-power narrow-band transmitter, we have leveraged it for sending the addressed wake-up beacons. To use PIC16 as a wake-up transmitter, it is configured for transmission using OOK modulation where the information bits are sent using ‘1’s and ‘0’s. An OOK 1 sub-bit is produced by transmitting a large amplitude carrier while an OOK 0 sub-bit is produced by sending nothing i.e., the transmitter is turned off allowing the system to save on transmit power when sending ‘0’s. The power required for all OOK transmissions is approximately 46 mW at a transmission power of +10 dBm. The transmitter and receiver are configured to use a separate antenna.

5.1.2 Main Sensor Node: Tmote Sky

For the main sensing node, we adopted the Tmote Sky as it is a popular platform used by the WSN community for research purposes. Moreover, this platform is supported by ContikiOS and COOJA allowing it to execute the same firmware in the simulator and on the real-hardware for controlling the radio transceiver. The platform is designed around the TI MSP430F1611 MCU with a low-power CC2420 radio transceiver offering a 250 kbps data rate and operating in the 2.4 GHz band. It also includes three onboard sensors: humidity, temperature, and light. The Tmote also offers various low power modes that permits it to run for months on a single pair of AA batteries. The average power consumption of Tmote is approximately 58 mW when transmitting and 65 mW when receiving.

The choice of the Tmote provided a rapid prototype to evaluate the effectiveness of the wake-up radio technology and its associated MAC scheme. Moreover, it also offers standard input/output (I/O) pins on the expansion header making it easy to interface a wide variety of external peripherals. We have leveraged this expansion header for connecting and controlling the wake-up radio module.
5.1.3 Enabling communication between Tmote and Wake-up Radio

For operating each radios separately, for instance, to start and stop each radio interface independently, we isolated the two radio stacks on two different MCU cores, mainly MSP430 core of the Tmote to control the CC2420 transceiver and the PIC16 core for managing the wake-up radio driver. These two cores and their respective stacks are illustrated in Figure 5.2.

We further need to provide mechanisms to communicate from the MSP430 core to the PIC16 for selecting the wake-up radio module for beacon transmission prior to sending the main data packet. For selective triggering, each wake-up radio module requires setting its own node_id for address matching and the remote_id of the transmitted packets. As each secondary wake-up radio module is expected to operate in conjunction with the primary data radio, the main mote is responsible for providing remote_id information to the WUR stack for the transmitted packets. The WUR driver reads this remote_id before beacon transmission to select the target receiver node.

Hardware interface. We chose the serial port interface (SPI) that is most widely used on the embedded platforms for communicating the above IDs between the two devices (MSP430 ⇒ PIC16). The MSP430 microcontroller on the Tmote Sky provides an independent SPI hardware module that is shared with the CC2420 transceiver. Therefore, we had to resort to the Software based SPI (SW-SPI) for communicating to the WUR module. A dedicated GPIO port of MSP430 is used to implement the control and data transfer using the expansion header of the Tmote Sky for interfacing the WUR.

For the SPI, a 3-wire handshaking mechanism is adapted to provide coordinated access for writing data over the SW-SPI. These control channels include chip select (CS), serial clock (SCK), and Master out Slave input (MOSI) lines. In the current implementation, Tmote Sky assigned as the master initiates the data transfer by first asserting the CS line for selecting the slave device and then writes the address bytes on the MOSI line of the WUR on each clock edge, SCK. Finally, CS is de-asserted to complete the data transaction. The WUR module does not return any data on the SPI line.
5.2 Software Design: Wake-up Radio Driver

Figure 5.3 – Tmote Sky integrated with the wake-up radio module.

On the WUR module side, we adopt the standard hardware SPI shown in Figure 5.1 to read and store the data. The CS line invokes the interrupt service routine (ISR) indicating to the PIC microcontroller that there is a message available to be read from the SPI buffer. Once a fixed number of bytes has been transferred, the ISR flag is disabled. For triggering the main node from a low-power sleep state, an Interrupt line from the PIC16 microcontroller is interfaced to the Tmote Sky. The entire wake-up radio module is powered by the Tmote Sky, requiring only a single power source. Figure 5.3 shows the complete platform assembled inside an IP 65 enclosure ready for deployment.

5.2 Software Design: Wake-up Radio Driver

From the hardware design, we now turn our attention to the software integration of the whole platform. The goal of this section is to outline and provide the lower-level driver using which other wake-up radio applications and systems can be built. More so, to simplify the control, the radio driver should provide all the necessary application programming interfaces (API) that allow functionalities for setting the WUR module, data processing, and communication.

Wake-up radio API. We developed wake-up radio driver APIs that includes a set of functionalities required by the applications that need to interact with the wake-up radio module. Table 5.1 summarizes these low-complexity APIs. The APIs exposed to the developer includes four main functions:

- WUR module Initialization. In order for the wake-up radio to function as intended, it requires initialization of the SPI interface, interrupt ports, and the RF transmitter performed by board_init when the module is powered on.

- SPI communication handler. SPI_ExchangeHandler is responsible for exchanging bytes from the Tmote via the SPI which is then stored on the processor’s buffer for setting node and packet addresses.

- Send packet. The send_packet function invokes the packet transmission of a spec-
Chapter 5. Wake-up Lab: Implementing a Wake-up Radio Testbed

Table 5.1 – Wake-up radio driver APIs.

<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>void board_init()</td>
<td>Initializes the WUR platform</td>
</tr>
<tr>
<td>uint8_t SPI_ExchangeHandler(uint8_t byte)</td>
<td>Read a byte from the SPI interface into the receive buffer</td>
</tr>
<tr>
<td>void send_packet(unsigned char *data, unsigned char num_bytes)</td>
<td>Transmit wake-up packet of size num_bytes (max 8 bytes)</td>
</tr>
<tr>
<td>unsigned short receive_packet()</td>
<td>Receive data bits and store in the buffer for address decoding</td>
</tr>
</tbody>
</table>

The send_packet function waits for all the data to be sent and signals a complete transaction to the main loop. In the current implementation, the user has to ensure that the first bit of the sent data is one in order for the receiver to correctly decode the packet address.

- Receive packet. The receive_packet() function is responsible for storing and decoding the address bits received over the always-on wake-up receiver interface indicated as Data_In line in Figure 5.1. The PIC16 microcontroller waits in sleep mode for the first bit to be one, which will assert the Wakeup line, triggering the onboard microcontroller from a low-power sleep mode. Once active, the microcontroller observes the state of the Data_In line to decode the OOK address by sampling the state of the GPIO pin for each bit received. After sampling the Data_In line it compares the value received to that stored in the node_id address buffer for triggering the Tmote.

Nevertheless, the APIs ensure that the wake-up radio module can perform a non-blocking send and receive operation. This is done by monitoring the status of the Data_In line and the interrupt service routines.

5.3 Wake-up Lab for Application development

So far we only looked at the node design and its software counterpart. As mentioned earlier, one of the goals of this chapter is to provide a modular dual-radio prototype that can be easily integrated into existing testbeds for evaluation of the wake-up radio technology. The intent is to integrate the developed platform as part of the SoleLab testbed operating at our facility, forming a Wake-up Lab.

SoleLab is an experimental wireless sensor network testbed consisting of 50 Tmote Sky motes deployed in the building of Fondazione Bruno Kessler as shown in Figure 5.4. It provides an intuitive web-based interface where users can remotely upload executables, create, and schedule experimental jobs. Each mote is powered by the USB and is connected to the Ethernet backbone, which facilitates direct capture of data and uploading of new programs.

As each mote in the SoleLab is connected to the testbed infrastructure via a USB interface, it is just a matter of taking our device shown in Figure 5.3 and connecting it to this port. Integrating
5.4. Summary

our developed platform to the SoleLab allows us to easily program these motes and conduct full stack experiments.

To show the proof of concept, we did connect our devices and ran the experiments using the SoleLab. We were able to successfully program our devices and initiate the wake-up transmission for triggering the remote mote. Unfortunately, due to the low sensitivity of our wake-up receivers, measured to be only a few meters, we were unable to successfully receive the wake-up trigger and conduct experiments using the larger set of nodes in the SoleLab. To solve this hardware issue, the new receiver design is underway at ETH.

Alternatively, to illustrate concretely the benefits of the wake-up radio system in a real environment, we have created a 4-node desktop testbed to benchmark the performance in a single and multi-hop data collection scenario, discussion of which is detailed in the next chapter.

5.4 Summary

In this chapter, we focus on the hardware and software integration for realizing a wake-up radio based wireless platform for debugging and evaluating protocols in a controlled, yet "realistic" environment. Towards this, we composed a modular dual-radio prototype that can be easily integrated into existing testbeds. Moreover, the adopted wake-up radio module offers bi-directional communication allowing developers to establish physical "underlay" of wake-up radios to create a multi-hop network in order to remotely wake-up the distant high power radio.

Future work will involve the study of the advantages of this new technology by using the developed prototype to compose a small-scale desktop testbed.
Exploiting Wake-up Lab

Wake-up radio technology has paved the way toward the extension of WSN lifetime by offering low-power on-demand communication. On the other hand, this technology is still in its relative infancy and the exact potential on the WSN system as a whole have not been demonstrated. While several works in the literature show potential, most of these comparison studies are either based on analytical models or simulators such as those presented in [214, 211, 212, 215].

In this chapter, we depart from pure simulation study to testbed evaluation for exploring the potential of the wake-up radio technology in a real office environment. To do so, we use our Wake-up Lab prototype presented in Chapter 5 to compose a small-scale desktop testbed where each Tmote Sky is equipped with a wake-up radio module. Using the testbed, we then provide microbenchmarks for the data collection using unicast and broadcast communication and for the system as a whole through a multi-hop network. Not only we show the limits and characteristics of the wake-up technology, but we also explore the effect of interference on the wake-up radio in the presence and absence of nearby in-band devices. We then validate the performance of our wake-up radio simulator WaCo, presented in Chapter 3 to the actual hardware. As WaCo allows to execute directly the codes compiled for the real nodes, making the comparison between hardware and simulation easier.

Contributions. In this chapter, we advance the state-of-the-art in wake-up radio performance evaluation by demonstrating its practical use in various data collection case studies. Specifically, we

- demonstrate and exemplify that the wake-up radio technology is promising for WSNs through a small-scale indoor desktop testbed.
- demonstrate that using a low-complexity W-MAC, wake-up radio based systems can achieve significant improvements over traditional duty cycling MACs.
- provide microbenchmarks for the unicast and broadcast networking using wake-up radios.
- demonstrate the effectiveness of the wake-up radio technology in enabling multi-hop periodic data collection.
- validate our WaCo simulator by comparing simulation results along the testbed.
Chapter 6. Exploiting Wake-up Lab

- evaluate the effect of external interference on the performance of the wake-up radio and the upper layers under controlled laboratory experiments.
- through in-lab measurements provide energy microbenchmarks for the wake-up radio link primitives.
- show that the wake-up radio prototype exhibits a total power dissipation of 1.83 µW during periods of inactivity.

To the best of our knowledge, this work provides the first in-depth testbed analysis for systematically characterizing the wake-up radio performance in a "realistic" environment over single-hop and multi-hop network with address decoding and re-transmission capabilities.

Structure of this chapter. The rest of the chapter is organized as follows. In section 6.1, we revisit our design of W-MAC, a transmitter-initiated link-layer that uses the dual-radio interface: the wake-up radio and the CC2420 transceiver. Section 6.2 outlines the evaluation methodology for benchmarking the wake-up radio and the description of the testbed environment. Sections 6.3 and 6.4 explore the microbenchmarks. The key microbenchmarks include the performance of the wake-up radio system with multiple senders and receivers and the baseline power consumption w.r.t ContikiMAC. Our microbenchmarks indicate that wake-up radio MAC offers higher network reliability and reduction in main radio-on time of 21× than ContikiMAC. We also show that the system is more responsive and uses fewer transmissions than ContikiMAC, making wake-up radio system more energy efficient. Finally, we also demonstrate that data collection using Contiki Collect over W-MAC achieves similar and in some cases outperforms the state-of-the-art. A maximum end-to-end latency of 78 ms is observed for a 3-hop network when routing using wake-up radios. In section 6.5, we also test the vulnerability of in-band interference and its impact on the wake-up radio technology. Section 6.6 provides the energy microbenchmarks for W-MAC's link primitives including transmission of the wake-up beacon, receive, transmit, and channel listening before concluding the chapter in Section 6.7.

6.1 Revisiting W-MAC

In duty cycled protocols, nodes periodically wake-up to transmit or to receive data. While radio duty cycling significantly decreases consumption, several shortcomings remain such as idle listening, overhearing, and costly continuous re-transmissions at the sender. To overcome these issues faced by traditional MAC protocols, wake-up radio has been a key enabler allowing on-demand asynchronous communication. To enable such on-demand communication, we have implemented W-MAC, a low-complexity MAC protocol for wake-up radio based systems.

W-MAC is a dual-radio transmitter-initiated link layer protocol where the source node initiates the data communication. The main radio channel is used for sending data and control packets, whereas the wake-up radio channel is dedicated for triggering neighbors. Whenever a sensor node has data to communicate, either generated by the upper layers of the protocol stack or forwarded by the neighboring nodes, W-MAC first transmits a wake-up beacon containing
Figure 6.1 – W-MAC: Transmitter-initiated wake-up radio MAC.

the destination node address using its wake-up transmitter as illustrated in Figure 6.1. The sender performs a clear channel assessment (CCA) using its main radio before starting the data transmission. If the CCA indicates that the channel is busy due to an on-going transmission or reception, the sender backs-off for a random period of time before retransmitting the wake-up beacon. On the contrary, if the channel is clear, the data packet is sent directly to the intended receiver after the synchronization delay. The synchronization delay is the time required by the wake-up receiver to receive and decode the wake-up packet and generate an interrupt to the main microcontroller. The primary radio then switches into RX mode to receive the acknowledgment (ACK) from the receiver before transitioning into sleep mode. If the ACK was unsuccessful within a predefined interval, the initiator returns to the beginning of the sequence, retransmitting the wake-up beacon.

On the destination side, the always-on wake-up receiver decodes the embedded address in the wake-up beacon to confirm the receipt and activates the main radio into RX mode. If the data packet is received successfully, an ACK is sent and the main radio is switched off. Instead, if the node was triggered but no data packet is received due to interference or collisions, the main radio is instantly switched off, while the wake-up receiver remains active listening for the wake-up beacons.

As per the dual-radio setup where the nodes have the capability to receive and transmit the wake-up beacon, W-MAC naturally forms a multi-hop network. W-MAC has been implemented in ContikiOS using its Rime stack to provide a lightweight communication between the nodes offering the same interface to higher layer protocols as its default ContikiMAC, making W-MAC seamlessly inter-operable.

6.2 Evaluation Methodology

In this section, we evaluate the wake-up radio and its associated MAC experimentally using our desktop testbed implementation. We first show, in a single-hop unicast data collection scenario that W-MAC increases network reliability and energy efficiency outperforming state-of-the-art solution such as ContikiMAC [8].

Second, we focus on the broadcast mechanism of the wake-up radio and demonstrate that W-MAC even improves energy efficiency in broadcast mode while maintaining high network
Chapter 6. Exploiting Wake-up Lab

reliability. We then ran a full Rime networking stack atop W-MAC, performing multi-hop data collection over the standard Contiki Collect protocol. Results indicate that W-MAC maintains high network reliability and low end-to-end latency because of its ability to trigger nodes on-demand with reduced delays w.r.t duty cycling MACs.

Finally, we inject controlled interference to study the impact of noise on the wake-up receiver and how the upper layers of the communication stack react in terms of re-transmissions.

Validation of WaCo. One of the drawbacks of simulators is not being able to fully capture and model the dynamics and the complexity of the test environment of which low-level hardware timing and the radio interference being the most important ones. Therefore, during our experimental campaign, we used the results obtained from the testbed as a baseline to evaluate the accuracy of our WaCo simulator. To correctly model and compare the results of the wake-up radio hardware used in the testbed, we had to modify two main parameters in the WaCo simulator: i) the transmission rate of the wake-up transmitter to 1 kbps and ii) the synchronization delay to 21 ms between the wake-up transmitter and the receiver. This synchronization delay includes the time required to setup the SPI communication, signaling of the wake-up transmitter to the activation of the main data radio.

The above parameters are obtained from the actual in-lab measurements of the wake-up radio prototype.

6.2.1 Experimental Configurations

We used the wake-up radio platform described in chapter 5 equipped with off-the-shelf 868 MHz 1/2 wave dipole antenna. The antenna orientation of the transmitter and the receiver is oriented at 0°, i.e., facing upwards. The desktop testbed is deployed in an office space where there are multiple wireless access points and mobile devices operating in the vicinity. All the nodes were stationary during the experiments, exposing them to real-time link dynamics from the surrounding environment as well as external interference. Moreover, due to the low sensitivity of our wake-up receiver prototypes, measured to be only a few meters, in all our experiments the nodes are placed half a meter apart to ensure maximum connectivity range.

In all the experiments, the main data radio, CC2420 uses the maximum transmission power of 0 dBm while the wake-up transmitter is configured at +10 dBm. The wake-up beacons are sent using OOK modulation at a data rate of 1 kbps consisting 16-bits of address data. The main radio, on the other hand, transmits at 250 kbps using Offset Quadrature Phase Shift Keying (O-QPSK) modulation. Unless otherwise stated, we set the application payload size to 6-bytes with an inter-packet interval (IPI) of 10 seconds. The total Rime stack payload is 43 bytes including the overhead.
Table 6.1 – Configurations used throughout our indoor testbed experiments.

<table>
<thead>
<tr>
<th>Description</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sensor node</td>
<td>TMote Sky</td>
</tr>
<tr>
<td>Main radio chip</td>
<td>CC2420</td>
</tr>
<tr>
<td>Main radio data rate</td>
<td>250 kbps</td>
</tr>
<tr>
<td>Main radio channel</td>
<td>26</td>
</tr>
<tr>
<td>CC2420 TX power</td>
<td>0 dBm</td>
</tr>
<tr>
<td>Wake-up radio data rate</td>
<td>1 kbps</td>
</tr>
<tr>
<td>Wake-up radio TX power</td>
<td>10 dBm</td>
</tr>
<tr>
<td>Wake-up beacon packet length</td>
<td>16-bits</td>
</tr>
<tr>
<td>Wake-up beacon TX duration</td>
<td>16 ms</td>
</tr>
<tr>
<td>Application Layer Payload Size</td>
<td>6 B</td>
</tr>
<tr>
<td>Full stack Payload Size</td>
<td>43 B</td>
</tr>
<tr>
<td>ACK Packet Length</td>
<td>5 B</td>
</tr>
<tr>
<td>No. of generated data packets</td>
<td>300</td>
</tr>
<tr>
<td>MAC</td>
<td>CSMA</td>
</tr>
<tr>
<td>Routing</td>
<td>Contiki Collect</td>
</tr>
<tr>
<td>Radio Duty Cycling</td>
<td>ContikiMAC, W-MAC</td>
</tr>
<tr>
<td>ContikiMAC Sleep Interval</td>
<td>125 ms</td>
</tr>
<tr>
<td>No. of experimental runs</td>
<td>3</td>
</tr>
<tr>
<td>Maximum hop count</td>
<td>3</td>
</tr>
<tr>
<td>Maximum retransmissions</td>
<td>8</td>
</tr>
</tbody>
</table>

6.2.2 Protocols

As W-MAC allows to run different routing protocols on top, in our experiments we use Contiki Collect, a routing protocol provided by the ContikiOS for periodic data collection in many-to-one scenarios. Contiki Collect uses unicast for all data transmissions and broadcast for route discovery. It supports both single-hop and multi-hop communication primitives enabling us to test the different modalities of the wake-up radio.

The routing protocol described above sits on top of the MAC layer, whose primary objective is to control the usage of the main radio transceiver, including duty cycling. For initial benchmarking, we use ContikiMAC, the de-facto, asynchronous low-power listening protocol for ContikiOS. ContikiMAC uses the data packet itself as a preamble and has a phase-lock mechanism, where senders record their neighbor’s wake-up phase to make subsequent transmissions cheaper for the sender. In addition, ContikiMAC performs fast sleep optimization allowing receivers to quickly detect false positive wake-ups. For ContikiMAC the sleep interval is set to 125 ms with phase lock-mechanism enabled.

The most significant difference between W-MAC and the ContikiMAC is its ability to exploit the wake-up radio, while the latter uses only the high data rate CC2420 transceiver.

6.2.3 In-software Power Measurements

It is not trivial to acquire measurements of the performance metrics that we are interested in, especially power consumption of the nodes from the desktop testbed. While measuring data reliability is quite straightforward by tracking the packet sequence numbers, measuring end-to-end latency and energy is rather difficult. While there are a few testbeds that offer these capabilities such as the FlockLab [193] and D-Cube [197], however, our desktop testbed does
not have power profiling capabilities. Instead, in our experiments, we measure the main node's energy consumption in software using Energest [216]. Energest profiler provides accurate energy estimations in Contiki by measuring the time spent by the node in different states such as low power, active, radio transmission, and reception, which can then be combined with the current draw in each state to estimate the overall energy consumption.

### 6.2.4 Performance Metrics

We evaluate the network performance using the following key metrics:

- **Packet reception rate (PRR):** measured as the end-to-end packet reception rate at the sink node averaged over all the runs. To compute the PRR, each generated packet contains the sequence number. In the case of the multi-hop experiments, also the number of hops the packet has traversed to reach the destination.

- **Radio duty cycle (RDC):** computed as the percentage of the total time the main radio is active using Energest profiler on each node. We use the duty cycle as a proxy for overall energy consumption because the power profile of a typical sensor node is dominated by its radio chip.

- **Latency:** measured from the time packet is generated at the application layer to its reception by the sink node including the wake-up transmission delay. We measure latency in testbed experiments using the Tektronix digital oscilloscope.

All the above metrics are computed based on 300 data packets after 60 seconds to allow the network to stabilize using the static configurations listed in Table 6.1.

### 6.3 Micro-benchmarks

In this section, we concretely demonstrate the benefits of the wake-up radio technology when integrated into a networking stack. We first focus on the results of the single-hop star network for microbenchmarking followed by the multi-hop data collection in section 6.4 for full-stack evaluation.

#### 6.3.1 Multiple Senders: Unicast Communication

We begin our microbenchmarking by considering a many-to-one single-hop data collection scenario. The focus of this experiment is to test the ability of the wake-up receiver to handle and respond to the wake-up requests from multiple senders. This type of communication is prevalent in wireless sensor networks where the routing protocols send data upwards to a single sink node.

**Setup:** For testing multiple sender flows, we use the topology shown in Figure 6.2 with one node configured as a receiver $R$ and varying the number of nodes sending data to it, $T_n$, separated by a distance of 0.5 m. The sender nodes generate and send data packets periodically.
every 10 s, to the receiver with a small, random delay (jitter) to avoid collisions among multiple senders. For each successful packet, the receiver sends an acknowledgment (ACK) to the sender node.

**Results:** Figure 6.3 shows the energy consumption of nodes between one and three senders transmitting to a single receiver using W-MAC and ContikiMAC. These results provide the basis for a fairly accurate estimation of the energy overhead incurred by ContikiMAC. The energy overhead of ContikiMAC is approximately $17 \times$ higher for the receiver node (ID 1) and $21 \times$ for the sender w.r.t to W-MAC. This was expected as ContikiMAC periodically wakes up at every 125 ms (as per the default configuration) to sample the channel even if there is no data to be sent or receive, increasing the overall main radio-on time.

To quantify this, we also measure the idle state time in hardware, referred to as the duty cycle baseline where the nodes do not generate any data packets. The plot in Figure 6.4 shows the percentage of the time node spends in low-power mode (LPM), in active (CPU), and for receiving (RX). The baseline for the ContikiMAC is measured to be 1.60% during the network idle state. On the other hand, mote equipped with a wake-up radio W-MAC only turns on the main radio when there is data to be sent, without any idle listening or continuous channel
Figure 6.4 – Baseline power distribution comparison for the main radio using ContikiMAC and W-MAC in the testbed.

probing by the sender, eliminating any extra energy cost. Although the main CPU wakes-up a few times due to the Contiki’s system timer that uses the tick interrupt to maintain a long term count of seconds, there is no main radio activity observed for W-MAC, with node spending 97% of the time in deep low-power mode.

We gain further intuition into W-MAC’s behavior by looking deeper at the main radio on-time and the network performance in Figure 6.5. It is observed that the energy consumption of the main radio at the sender nodes remains relatively constant while at the receiver it increases as the number of senders increase. To validate the WaCo simulator, we also plot the results obtained from the simulator alongside the testbed in Figure 6.5. Simulation evaluations show similar trends for the energy consumption indicating that the simulator follows very closely to that of the real implementation with minor difference in the simulation w.r.t to the testbed. This is attributed to the fact that in simulation, the data transmissions for the wake-up radio are almost perfect due to the unit disk graph channel model where the nodes within the disk successfully receive all the packets. Hence, no re-transmissions are required for the small scale network such as the one evaluated here.

As per Figure 6.5b, testbed results show a slight decrease in the PDR as the number of senders grow due to the increased contention caused by frequent wake-ups occurring at the receiver. Very few packet drops occurred in this scenario and the loss rate is less than 3% in the testbed experiments while perfect reliability is observed in the simulation due to the aforementioned reasons.

We now turn our attention to the data latency of the wake-up radio MAC presented in Figure 6.5c. In the simulation, the average end-to-end latency for a single-hop network is measured to be 23.12 ms due to the tight timing from the wake-up radio signaling to the activation of the main transceiver. In the testbed, this is measured to be 26 ms due to the extra few milliseconds (2.88 ms) required by the CPU to activate the main data radio. Nevertheless, we
conjecture that the link layer delays imposed by W-MAC are much smaller than the channel sampling rate of ContikiMAC with wakeup interval of 125 ms, where the receiving nodes only have to wait long enough to receive the wake-up signal. As expected, W-MAC’s per packet latency remains constant regardless of the number of senders in a single-hop network.

This microbenchmarking has quantitatively shown that wake-up radio technology provides a robust and reliable system with a latency of 26 ms for 1-hop communication. In contrast to ContikiMAC, a reduction in main radio on time of $21 \times$ is observed for the current system equipped with a secondary wake-up radio. Moreover, the results obtained also confirms that the wake-up radio modeled in the simulator follows very closely with the real-world.

Figure 6.5 – Single-hop network performance of W-MAC in simulation and the testbed over varying number of senders.
Chapter 6. Exploiting Wake-up Lab

Figure 6.6 – Network topology used for the broadcast experiment.

Figure 6.7 – Broadcast primitive performance of W-MAC in the testbed and simulation.

6.3.1.1 Multiple Receivers: Broadcast Communication

A broadcast is a fundamental technique used by various higher layer applications and services such as CTP and RPL for neighbor discovery, network-wide data dissemination, and route selection. All these applications depend on robust broadcast service for reliable operation.

Setup: To further test W-MAC, we ran additional set of experiments with the intent to wake-up all the nodes $R_n$ in range over broadcast communication using the topology shown in Figure 6.6. The receivers were placed at 0.5 m apart from the sender node.

W-MAC’s design of broadcast primitive is identical to unicast communication with one main difference. When the sender $T$ wants to send a broadcast call, it sets the destination address of the wake-up beacon to the broadcast address 0xFFFF, and submits the frame to the wake-up transmitter via SPI for delivery, without requesting any ACK. In this setting, the wake-up beacon reaches all 1-hop neighbors, who subsequently turn on their main radio to receive the data packet.

Results: As expected, our analysis in Figure 6.7a shows that the energy consumption remains similar to that of unicast communication both in the testbed and in simulations, as the receivers still have to decode the address in the wake-up beacon. We argue that the broadcast expense of W-MAC is far lower than ContikiMAC which transmits broadcast packets repeatedly.
for a full wakeup period to maximize the possibility of reception. However, with longer sleep interval, these broadcast transmissions in ContikiMAC are going to be more expensive while the W-MAC’s remain close to the unicast mode.

Observing the PRR, the results in Figure 6.7b indicate that all the receivers were able to decode the wake-up frames with high probability. Since there is only one sender and no contention for transmissions, the amount of packet drops remains very low. The average PRR in the testbed is above 97%. The 1-hop latency remain the same for broadcast mode, therefore we do not show the results. Similar results are obtained from the WaCo as well validating our simulator design and its ability to correctly model the wake-up radio system.

6.4 Testbed Multi-hop Experiments

We next investigate how well the wake-up radio system performs in a more complex scenario when collecting data over the multi-hop network using tree-based collection protocol, Contiki Collect. This experiment also enables us to test the performance of the unicast and broadcast primitives of wake-up radio system collectively. Running Contiki Collect over wake-up radio MAC layer is largely a matter of changing the network configuration settings in the Contiki during the compile time.

In many data collection applications, mainly duty cycling protocols are utilized to extend the node lifetime. Duty cycling, however, affects the overall system response. The longer the sleep interval of the node, the higher the data latency of the network, making the system less responsive. For instance, rare-events of interest such as a sensor parameter exceeding a safe level that needs urgent attention can be missed while the nodes are asleep. On the other hand, in the case of asynchronous on-demand communication offered by the wake-up radios, the sensing nodes can offload the data in a timely manner by directly waking up the neighbors on-demand and then fall back to sleep mode. This not only allows saving energy at the sender but also at the receiver minimizing its radio on-time during subsequent communications by listening just after the wake-up trigger reception.

Setup: We deployed a total of 4 prototype nodes in an indoor office testbed. The nodes were configured to form a 3-hop network, each separated by the distance of 0.5 meters running Contiki Collect atop W-MAC. The network topology is illustrated in Figure 6.8. The initiator node I starts the packet transmission which is relayed over R2 and R1 until it reaches to the sink node S. The inter-packet interval rate, IPI is varied from 0.5 s to 30 s with the lower IPIs showing the performance of the network in a highly stressed setting. The experiments are repeated three times for each IPI. All the experiments were performed with a wake-up address size of 16-bits.

Results: The results of the testbed experiments are shown in Figure 6.9. We begin our evaluation by considering the main radio’s energy consumption of the participant nodes, shown in Figure 6.9a. At the smallest IPI, i.e., the highest load, all the nodes show increased power consumption due to more frequent transmissions and channel contention leading to packet
Chapter 6. Exploiting Wake-up Lab

Figure 6.8 – Testbed network topology for conducting multi-hop data collection.

![Network Topology Diagram](image)

Figure 6.9 – W-MAC’s performance for data collection over 3-hop network with varying Inter-packet intervals in the testbed.

(a) Main radio on-time for participant nodes.  
(b) End-to-end reliability.

re-transmissions. The consumption of the relay nodes, router 1 & 2 is the highest as they must receive and forward all the packets towards the sink, while also acknowledging the receptions. As the IPI decreases the effect of data traffic gradually reduces until the main radio on timing is dominated by the protocol overhead. The reason for this significant performance is that W-MAC reduces idle listening of the power-hungry main radio and does not need long packet trains to deliver the packets as in duty-cycling MAC protocols, thus achieving short transmission times. At the lowest IPI of 30 s, all the nodes have an average RDC of below 0.2% while the initiator node achieves the lowest of 0.052%.

We now turn our attention to the overall network reliability shown in Figure 6.9b. When the system is overloaded at IPI of 0.5 s, nodes suffer from the queue overflow and packets are dropped reaching average reliability of only 30%. As the traffic load decreases, W-MAC achieves perfect reliability. The higher packet reception rate in this small network is attributed to the improved link quality due to no physical obstruction between the nodes. Overall, W-MAC shows high network capacity and its ability to handle high bandwidth over IPI of 5 s.

The maximum end-to-end latency of 78 ms is measured for the 3-hop network. This is due to the reduced per-hop latency exhibited by W-MAC (26 ms as observed in the single-hop network) allowing it to push more packets through.

The results obtained from the testbed experiments confirm that the wake-up radio based system satisfies realistic workloads, achieving energy-efficient and responsive system while offering high data reliability.
6.5. Robustness to External Interference

Up till this point, we could accurately measure and compare the performance of the wake-up radio in the testbed and simulation. As simulators tend not to capture the vagaries of the wireless environment accurately and completely, we now only rely on the testbed for conducting the next set of experiments on the in-band interference.

Full-fledged radio transceivers such as CC2420 allows performing clear channel assessment based on the measured received signal strength indicator (RSSI) value provided by the radio chip, increasing the complexity and power consumption of such transceivers. Due to the simplicity of the custom-built wake-up receiver utilized in this work, we are not able to acquire such information from the receiver to perform channel sensing task at the wake-up radio level. Although, the low-complexity design allows reducing the number of power hungry components leading to a sub-microwatt power dissipation, this also makes the wake-up receiver vulnerable to in-band interference. For instance, two nodes concurrently transmitting an OOK-modulated signal in the same frequency band can result in either a constructive or destructive interference at the wake-up receiver. If the signal is constructive, the wake-up receiver will be able to decode the address bits correctly. In contrast, if the interference is destructive as depicted in Figure 6.10 where bit 0 is flipped to bit 1, this will corrupt the address bits, preventing the successful trigger of the intended node.

**Setup:** To quantify the effects of the in-band interference on the wake-up receiver operation and its ability to correctly decode the wake-up address, we conducted testbed experiments with an external interferer node $I$ as illustrated in Figure 6.11. The transmitter and the receiver node is located half a meter apart. The transmitter node $T$ generates and transmits the unicast packet to the receiver node $R$ at an inter-packet interval of 5 s and waits for the ACK. For generating external interference, we use a CC1350 Launchpad controlled by the Texas Instruments SmartRF studio that continuously transmits an OOK-modulated signal at the data rate of 1 kbps in the 868 MHz band. We start the experiments with the interferer node off and then we turn it on.

**Results:** Figure 6.12 shows the results of the experiment in which we measure the number of link-layer transmissions and the packet reception rate at the receiver node $R$. Initially, when
there is no interferer node (Packet ID from 1 to 15), the wake-up beacons are correctly decoded and all the packets are received by the destination R with 100% reliability. On the link-layer transmission side, we observe that in some cases a maximum of three attempts are required to achieve this high reliability as the office test environment is not purely interference free. Although the system performs better under these “ideal” conditions, it degrades dramatically when the interferer node is turned on, indicated as jammer activated. First, the number of transmission attempts for the data packet increases to eight, the maximum allowed by the link-layer. This indicates that the main node tries harder to send the packets through under heavy interference w.r.t to the "ideal" conditions expecting ACK from the receiver. As a result of wake-up packet corruption, the receiver is not able to decode the address bits correctly leading to PDR of 0%. A similar trend is observed when the interferer node is turned off and on again through the experiment.

These results suggest that harsh in-band interference can be detrimental to the performance of low-complexity wake-up receivers. One solution to mitigate such interference is employing collision avoidance mechanism with random back-off or frequency randomization techniques at the wake-up receiver level. However, this will require the re-design of the current wake-up receiver architecture to incorporate extra components such as active band-pass filters and amplifiers to measure RSSI. There is a trade-off as this may negatively affect the overall listening consumption of the wake-up receiver due to additional active components. Our results suggest
Table 6.2 – Measured power consumption using WaCo and hardware.

<table>
<thead>
<tr>
<th>Events</th>
<th>WaCo (µJ)</th>
<th>Hardware (µJ)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Wake-up beacon transmission</td>
<td>736</td>
<td>802</td>
</tr>
<tr>
<td>Address reception and decoding</td>
<td>8.42</td>
<td>12.05</td>
</tr>
<tr>
<td>Main radio data transmission</td>
<td>143.24</td>
<td>162.17</td>
</tr>
<tr>
<td>Main radio data reception</td>
<td>135.32</td>
<td>154.92</td>
</tr>
</tbody>
</table>

that further research efforts in this direction are required to deal with interference for the wake-up radio based systems.

6.6 Energy Micro-benchmarks

W-MAC’s services are built by multiple sets of link primitives including transmission of the wake-up beacon, receive, transmit, and channel listening by the wake-up receiver. To validate the energy costs of the proposed wake-up radio system, we conducted the in-lab power benchmarking of the whole system.

Setup: We set up two Tmote Sky motes, one connected to the wake-up receiver and the other to a wake-up transmitter running unicast data collection. Figure 6.13 shows these primitives as well as their power consumption traces collected using an N6705B DC Power Analyzer with a 3.3 V power supply.

Results: The power profiling begins with both motes in a low-power mode, i.e, LPM4 with the wake-up receiver active, listening to the channel. During this state, the power consumption of the whole mote is measured to be approximately 142 µW out of which 1.83 µW is consumed by the WuRX. The transmitter mote then initiates the 16-bit wake-up beacon transmission containing the receiver address. The power required for OOK transmissions is approximately 46 mW at +10 dBm.

On the wake-up receiver side, 340 µW is consumed while receiving and decoding the address bits by the PIC16LF1824T39A microcontroller. Once the address bits are matched, the mote is triggered for exchanging data over the CC2420 transceiver. The main radio consumes 65.3 mW while receiving and 71.9 mW for data transmission at an output power of 0 dBm, which concurs with the data-sheet of the CC2420 transceiver.

As can be observed from these traces, the wake-up radio exchange lasts longer than the main data exchange due to the fact that we ran our experiments with a wake-up radio bit rate of 1 kbps supported by the designed prototype. The traces also show the latency of wake-up signaling to data reception measured as 26 ms.

Further, as listed in table 6.2, the consumption we measure in-lab is close to that measured in WaCo simulator, validating our simulation environment. Small differences account for variations in hardware configurations, such as keeping the wake-up receiver active for a short period after the reception ends.
6.7 Conclusions

This chapter evaluates and benchmarks the performance of the wake-up radio technology under three realistic data collection scenarios: single-hop unicast, broadcast, and 3-hop network using an indoor desktop testbed.

In a single-hop multiple flow setting, wake-up radio MAC outperformed the duty cycling ContikiMAC in terms of data transmission costs (reduction by 21×) and packet delivery latencies while achieving similar network reliability. Similar network performance is also observed for the broadcast data collection.

We then tested the whole setup for data collection in a network of 3-hops. Here we found
out that the trends for the energy consumption and latency are about the same as the single hop case. As the number of hops increase, the data latency increases linearly for the wake-up based network. For a 3-hop network, a maximum end-to-end latency of 78 ms was measured for a 16-bit wake-up packet. W-MAC also exhibited high network capacity achieving perfect reliability above inter-packet interval of 5 s.

We further validated our WaCo simulation framework by inserting the parameters extracted from the actual hardware and replicating similar network topologies as of the testbed. Using the same firmware as deployed on the nodes, we confirmed the results from the experimental setup that indicate similar performance gains.

Finally, to quantify the effects of the in-band interference on the wake-up receiver, we also tested the experimental setup under controlled harsh interference. As expected, the performance of the wake-up receiver degrades significantly due to destructive interference causing data corruption. This not only affected the reliability of the system but also the energy consumption of the nodes by increasing the number of link layer transmissions. This, however, needs further attention when designing and developing low power wake-up receivers in order to mitigate the effect of in-band interference. Use of complex modulation techniques such as Amplitude Shift Keying (ASK) or Frequency Shift Keying (FSK) that offers better noise immunity compared to OOK presents an opportunity here. However, the use of complex modulation techniques demands complex circuitry at the RF front-end such as the use of active demodulators, mixers, and amplifiers that require extra power. As studied recently in [217], it would be interesting to also analyze the effect of the packet length on the interference and at the MAC layer for the wake-up based systems, which we plan to study in the future.

Our practical testbed evaluations in this chapter establish that there is still plenty of room at the upper stack of the network to improve the efficiency of the wake-up radio based systems for guaranteeing predictable performance. Moreover, this work also paves the way for further research in the design of the wake-up radio hardware to enable channel sensing and frequency hopping capabilities for achieving channel-efficient asynchronous links over wake-up radios.
Exemplifying WUR: Energy Neutral and LPWAN
Plug into a Plant: Plant Microbial Fuel Cell and Wake-Up Radio in Synergy Toward Energy Neutral Sensing

Wireless sensor systems have witnessed extraordinary growth over the past decade, particularly for monitoring infrastructures, natural environments, habitats, logistics, and most significantly for security. In most cases, sensors are deployed in remote locations with limited access to a continuous power source and yet require lifetime from several months to a few years. As such, sensing devices are typically battery-powered.

To extend battery life, sensor nodes exploit low power consumption states such as standby or sleep, duty cycling between active and inactive modes to save power. Although the current drawn in the sleep mode is relatively low, over a long period or with many nodes deployed, a large amount of energy is wasted. To quantify this, consider that for the popular Tmote node [6], the CC2420 radio module consumes 20 $\mu$A when powered down and the MSP430F1611 MCU consumes 2.0 $\mu$A in the sleep state. Over a day, this amounts to a baseline total of 5.718 joules, while additional tasks such as sensing, computation, and communication significantly increase this value.

Motivated by this analysis, in this chapter, we turn our attention to reducing overall node consumption to a point at which these devices can be powered using energy harvesting sources, reaching the goal of energy-neutrality. Toward this, we consider three innovative techniques across the spectrum, namely, an extremely low-power switch composed of a wake-up receiver for power management at the hardware side, a new receiver-initiated MAC-level communication protocol at the networking side, plus a microbial fuel cell at the power source side. Specifically, we propose an on-demand sensing system, where an extremely low-power switch is continuously powered by a novel plant-based microbial fuel cell, PMFC. The switch contains an ultra low power wake-up receiver, which can be remotely triggered by a wireless signal. Receipt of the trigger causes a standard Tmote to be woken up to sense then transmit data.

A microbial fuel cell, MFC, converts the chemical energy of organic matter found in sedi-

---

The contents of this chapter have been originally published as: Plug into a Plant: Using a Plant Microbial Fuel Cell and a Wake-Up Radio for an Energy Neutral Sensing System, Rajeev Piyare, Amy I. Murphy, Pietro Tosato, and Davide Brunelli, In Proceedings of the IEEE 42nd Conference on Local Computer Networks Workshops (LCN Workshops), Singapore, October 2017, and have been slightly adapted for this thesis.
ment into electricity by exploiting the metabolism of bacteria [218]. Therefore, PFMCs offer a promising power source for sustainable low power sensing applications in natural environments such as precision farming or greenhouse monitoring. Nevertheless, PFMCs are severely limited in the quantity of energy they can generate, usually on the order of tens of microwatts. Specifically, due to high internal resistance and low power output, PMFCs are not able to directly power electronic devices [219]. Therefore, in this work, an intermittent energy scheme in the form of a boost charger and a capacitor is utilized to harvest and to store the energy. The boost charger converts the low potential generated by PMFC to a higher one for charging the capacitor. The energy stored by the capacitor is then used to continuously supply the on-demand switching system that consumes only 1.8 µW.

The system is completed by a standard sensor node, a Tmote, which is completely off except when it is triggered externally by the always-on wake-up receiver component of the switch. The remote, plant-powered system is periodically queried by the sink node, which collects and stores the data. Once the data transmission is completed, the power is completely cut off from the sensor node while the switch with the wake-up receiver is kept in listening mode. This innovative approach is different from the prior works in that the main node consumes no power during periods of inactivity except for the nano-power switch, allowing the sensing system to be powered with a low-energy source like PMFC.

In this work, we assume the sink node is battery powered and works in a duty-cycle manner where periodically waking up and sending signals to pull data from the sensing nodes. During periods of inactivity, the sink node is also put into a low power mode to save energy. The overview of the full lab setup with various components is shown in Figure 7.1.

Contributions. In this chapter, we offer following contributions:

- design of a system that combines multiple, novel hardware and communication technologies to obtain a fully energy autonomous sensing system.
- exploit a novel microbial fuel cell as a power source for sustainable sensing.
- propose a low-complexity receiver-Initiated MAC, WRI-MAC, for on-demand data collection in energy-constrained environments.
- a concrete in-lab evaluation of the whole system to validate the feasibility of our proposal.

Structure of this chapter. Following a description of the core technologies in Section 7.1, we then outline the overall architecture of our plant-based sensing system composed of the energy harvester, the on-demand switch, and the sensor node in Section 7.2. We offer a description of the various hardware technologies and how we have combined all these components together. Section 7.3 presents the models for estimating energy requirements of the various system components to balance with the energy harvested from the PMFC. We then in Section 7.4, present a new wake-up radio based receiver-Initiated MAC for data collection in a star network. As a proof of concept, the whole system is tested under a controlled laboratory experiment and the results are presented in Section 7.5. The chapter concludes in Section 7.6 with a discussion
7.1. Background

This section offers background on the novel power source used throughout this chapter to achieve a self-sustained and maintenance-free on-demand sensing system for autonomous long-term monitoring.

7.1.1 Plant Microbial Fuel Cell

Microbial fuel cells are bioreactors that exploit a property of particular bacteria, namely producing electrons as a result of their own natural metabolism. These microbes are found in the major part of the soils and sediments on the planet. To extract power, typically two electrodes made of inert material such as graphite are used as illustrated in Figure 7.2. The anode is buried under the sediment while the cathode is placed on top where oxygen is present. Electricity is generated by the oxidation of sediment organics as a byproduct of microbial metabolism. The typical continuous power output that can be exploited with a carefully designed harvesting circuit is approximately 50 to 80 $\mu W$ at 0.3-0.6$V$ level.

It has been shown in previous studies [220] that the addition of flora/plant inside the MFC ecosystem invigorates the performance of the bacteria, boosting the power generation from the MFC and incrementing the output voltage [218]. Further, the plant extends the lifetime
Figure 7.2 – Soil based plant microbial fuel cell.

of the bacteria by providing more nutrients to the soil. Researchers have also shown that harvesting electricity does not affect the plant-growth. The overall result can fit in a standard potted plant requiring only the typical care of occasionally watering to keep the soil moist and the plant alive, which in turn generates a small amount of electricity that can be exploited by the system.

Finally, various factors have been reported to have a major impact on the performance of MFCs [221]. Principally, the ambient temperature greatly affects the microbial activity. Further, dissolved oxygen at the anode impedes power output, and the distance between the electrodes affects the internal resistance of MFCs. Poorly managing these factors can prevent the output power from reaching the optimal value.

7.2 System Architecture

Figure 7.3 offers a high-level overview of our proposed plant-based sensing node system composed of the energy harvester, the on-demand switch, and the sensor node. Notably, the only power source is the plant-based harvester. The system is completed by a sink node composed of a battery powered Tmote. This section offers a high-level description of the primary components and how we use them in this case study.

7.2.1 Energy Harvester

The energy harvester is the core of our system as it is responsible for power generation and distribution to the node. It is composed of three sub-blocks:

PMFC. The PMFC provides the power source, making the remote node design very challenging because of the low power it delivers. The prototype we used is made up of a pot filled with soil from the university courtyard, a standard houseplant able to survive in a very moist environment, and two graphite fiber felt electrodes. After several days of setup time in which the bacteria colony starts growing, the system is able to produce an average output power of
7.2. System Architecture

![System Architecture Diagram](image)

Figure 7.3 – Overall architecture of the wake-up receiver triggered on-demand sensing unit. When a trigger is received, the switch closes to connect the sensing and transmitting node to the PMFC-based energy harvested power supply. Black lines indicate the power supply while blue indicates trigger signaling. PMFC is the only power source for the whole system.

Due to the low output voltage of the PMFC, the energy harvesting module is interfaced with a TI BQ25505 boost charger that shifts the low input voltage to a higher output one to charge the capacitor. This particular IC is chosen as it requires an ultra-low quiescent current of 5 nA, compliant to the PMFC power production, and also embeds a maximum power point tracking (MPPT) that allows maximizing the power extracted from the harvesting source. Through experimental measurements, the efficiency of the boost charger is found to be 80%.

The output of the boost charger is then used to charge a single 22 mF super capacitor that is the main storage bank for our remote node. The intermittent energy harvesting scheme that we adopted uses cyclic charging and discharging of the capacitor to store the energy generated by the PMFC. At full charge, the capacitor is able to provide 33 mJ, enough to power the entire system for a short period. The size of the capacitor in our design can be tuned, depending on the characteristics of the power source and energy requirements of the system.

7.2.2 On-demand Switch

The energy harvester and the sensor node are separated by the on-demand switching mechanism that is the key component for connecting and disconnecting power to the node. As power consumption of the Tmote Sky is typically hundreds of micro-watts (130 µW in sleep mode), much more than the PMFC can continuously supply, it is essential to incorporate a cut-off switch so that enough energy can be accumulated by the capacitor to power the sensor node only when required. The switching system uses two main components to effectively achieve on-demand sensing, namely, an always-on wake-up receiver and a switch. The first component allows detecting the trigger from the sink node and controlling the switch, while the second component bridges the power from the energy harvester to the sensor node.

Wake-up receiver. The WuRX solution that we adopted is a custom designed ultra low-power
Chapter 7. Plug into a Plant: Plant Microbial Fuel Cell and Wake-Up Radio in Synergy
Toward Energy Neutral Sensing

module, representative of most WuRXs found in the literature. The WuRX module operates in the ISM 868 MHz band and has a receiver sensitivity of -45 dBm with a maximum communication range of 20 m. It also features an ultra-low power MCU that is used for decoding an address embedded in the RF carrier for selective triggering. The overall power consumption has been experimentally measured at 1.68 µW in standby mode when listening for the signal, and 1020 µW when it is actively receiving and decoding the preamble or address. Wake-up triggering time was measured as 16 ms for the 16-bit address mode.

Nano-power switch. As mentioned in Section 7, when there are no sensing tasks to be performed, we turn the sensor node completely off as the energy harvester cannot provide enough power to keep the node in sleep-mode. As shown in Figure 7.3, the TPL5110 switch connects the power supply to the Tmote when it receives the signal from the wake-up receiver. The cycle repeats every time the on-demand switch is triggered. The TPL5110 chip incorporates a low power timer with an integrated MOSFET driver for switching and consumes only 35 nA. It also supports a number of modes and can be used as a stand-alone timer for systems that need to wake-up periodically.

7.2.3 Sensor Node

After the switch is triggered, the Tmote is powered ON, performs the sensing and communication then raises feedback to the TPL5110 IC by asserting the GPIO (HIGH) indicating that it is time to disconnect power. We adopted the Tmote Sky as the main remote sensing node as it is a popular platform used by the sensor network community for research purposes. Moreover, it is also supported by ContikiOS and COOJA simulator. The platform is designed around the TI MSP430F1611 MCU with a low-power CC2420 radio transceiver offering a 250 kbps data rate and operating in the 2.4 GHz band. It also includes three onboard sensors: humidity, temperature, and light. The Tmote also offers various low power modes. The average power consumption is approximately 58 mW when transmitting, 65 mW when receiving, and only 163 µW in idle and 15 µW in standby mode. In our setup, the Tmote runs the Contiki Operating System that is used to control the radio transceiver via a custom MAC protocol detailed in Section 7.4.

The choice of the Tmote provided a rapid prototype to evaluate the effectiveness of our proposed MAC scheme. In the future, we will consider alternate, ultra-low power MCUs, such as ARM Cortex-M4 processors [222, 223], and radios that will facilitate the design of an energy neutral wireless sensing system powered by PMFC.

7.3 Energy Requirements and Delivery

To ensure perpetual operation, the energy required by the sensor node for processing, sensing, and communicating $\Delta E_{node}$ must be balanced with the energy harvested from the plant micro fuel cell $\Delta E_{c}$. If $\Delta E_{c} < \Delta E_{node}$, the device will run out of power and fail. Achieving the proper balance requires analysis of the requirements of various hardware components as well as
energy efficient protocols that can operate within a strict energy budget. In this section, we present a model for estimating the energy production and requirements of various system components including:

- the energy required by the sensor node $E_{\text{node}}$,
- the energy delivered by the PMFC in continuous operation $E_{\text{mfc}}$, and
- the energy that can be stored in the capacitor $E_c$ as a function of time.

### 7.3.1 Baseline Power Consumption

In our laboratory, we measured the average sleep power of the Tmote at 130 $\mu W$, a value higher than what the PMFC can continuously supply. Therefore, in our proposed system, the Tmote is completely shut down, except when an external trigger arrives to indicate a sensor should be probed and its value sent. While this technique saves energy, not all components can be turned off, specifically the boost charger, wake-up receiver, and the switch must remain active, together consuming a non-negligible amount of power: 1.8 $\mu W$ (600 nA) in idle mode. This baseline power consumption is illustrated in Figure 7.4a and reported in Table 7.1 as the baseline.

### 7.3.2 Active State Power Consumption

When the sensor system is triggered, a number of actions take place: reception of the wake-up trigger, powering up of the sensor node, sensing, and transmission of the sensed value. Table 7.1 reports these individual costs along with other, common functionality of the Tmote, e.g., CC2420 RX, even though we do not exploit them in this chapter. The total energy consumption of the remote node is the sum of the energy consumption of its individual hardware components $i$, plus the costs of the on-demand switch, processor, radio transceiver, and the sensors. The total energy $E_{\text{node}}(t)$ required by the node is then computed as the product of the power and the time taken to carry out each task:

$$E_{\text{node}}(t) = \sum_i P_i(t) \cdot T = \sum_i V_i(t) \cdot I_i(t) \cdot T$$

(7.1)

Each of these tasks is labeled in Figure 7.4b. In region A, the trigger signal is received over the WuRX and the PIC micro-controller is activated for decoding and address matching. During this process the current consumption of the WuRX increases from 1.68 $\mu W$ to 1.02 mW for a period of 16 ms. If the address is a match, the switch is closed to turn on the Tmote Sky (60 mW spike). The Tmote Sky turns ON and enters into sleep mode (with a current draw of 43 $\mu A$) and then switches to an active mode for at most 640 ms (region B). This period is required to boot up and stabilize the crystal oscillator. Finally, region C shows the power dissipation for sensing and communication. The Tmote Sky activates the on-board temperature sensor and samples the value. It then turns on the CC2420 radio transceiver to send this sensor reading to the sink node before locally signaling the transmission done to the switch. The whole process from triggering to transmission on average takes 680 ms. Hence, the maximum energy required...
7.3.3 Energy Delivered by the PMFCs

Over six months of continuous monitoring in an indoor environment, we observed the average output power that can be extracted from the designed PMFC is $E_{_{\text{mfc}}}=70 \mu W$ as shown in Figure 7.5. This average value can grow up to $200 \mu W$ peak for short periods of time if the plant is healthy, usually followed by a recovery period needed to restore the normal “idle” operating condition. This way of operation is well exploited by our power management system that drains power only when the wake-up receiver is addressed.

7.3.4 Energy Stored by the Capacitor

To supply continuous power to the on-demand switch and peak power to the sensor node we exploit a $22 \mu F$ capacitor, which, when fully charged, supports $3.36$ volts. The energy $E_{_{C}}$ during the active mode is $E_{_{\text{node}}}=3.6 \text{ mJ}$.
stored (accumulated) in the capacitor when it is charged from $V_d = 2.87\, V$ to $V_c = 3.36\, V$ is estimated as:

$$E_c = \frac{1}{2} C(V_c^2 - V_d^2)$$  \hspace{1cm} (7.2)

where, $C$ is the capacitance and the $V$ is the supplied voltage from the boost charger. At full charge, the capacitor provides $E_c = 33\, m\, J$. This stored energy is sufficient to supply the burst energy required for the measurements and transmit operations, quantified as $3.6\, m\, J$ in Section 7.3.2. Further, we note that the size and number of capacitors can be modified depending on the system's power requirements.

### 7.4 Energy Efficient Communication for Sustainable Sensing

From hardware integration, we now move on to the basic operation of the communication, describing our wake-up radio based receiver-Initiated MAC, WRI-MAC, for data collection in a star network. We begin with an introduction to the basic idea followed by presentation of the communication scheme between nodes.

In receiver-Initiated systems (RI-MAC), the burden of starting a communication event falls to the receiver, specifically with a node, often the sink, announcing its readiness to receive data. After this announcement, the sink switches to receive (RX) mode and monitors the wireless channel for any incoming packets. Our choice to adopt a receiver-initiated MAC was motivated by two facts. First, RI-MAC protocols require no network synchronization among sensor nodes thus offering pure asynchronous communication. Second, collisions among senders are eliminated as the receiving node is in charge of pulling data from the individual nodes when required.

---

**Figure 7.5** – Long-term monitoring of PMFC and its output power in an indoor environment.
Table 7.1 – Laboratory power and current measurement of the different components in various states. Idle reflects listening to the channel, but not actively receiving.

<table>
<thead>
<tr>
<th>Component</th>
<th>Current Consumption</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tmote (MCU on, CC2420 TX mode)</td>
<td>19.5 mA</td>
</tr>
<tr>
<td>Tmote (MCU on, CC2420 RX mode)</td>
<td>21.8 mA</td>
</tr>
<tr>
<td>Tmote in deep sleep mode</td>
<td>43 µA</td>
</tr>
<tr>
<td>WuRX in idle state</td>
<td>0.56 µA</td>
</tr>
<tr>
<td>WuRX (Receiving + address decoding)</td>
<td>340 µA</td>
</tr>
<tr>
<td>WuTX (Transmitting)</td>
<td>10 mA</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Component</th>
<th>Energy Budget</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sensor Node + Switch (Active)</td>
<td>3.6 mJ</td>
</tr>
<tr>
<td>WuRx idle + DC-DC Boost + Switch (Baseline)</td>
<td>1.8 µW</td>
</tr>
</tbody>
</table>

**Other Parameters**

- CC2420 bit rate: 250 kbps
- WuR bit rate: 1 kbps
- Application Payload: 6 B
- WuR addressed packet size: 2 B

Figure 7.6 – WRI-MAC in unicast mode using address-based wake-up beacons. A single, separate wake-up beacon (W) is sent to initiate communication with each respective transmitter.

WRI-MAC, proposed in this work, assumes that each remote plant-powered node is equipped with a Tmote Sky and a wake-up receiver while the sink node is equipped with a Tmote Sky and a wake-up transmitter. WRI-MAC operates in both unicast and broadcast modes and has been implemented for ContikiOS.

**Unicast Communication.** In the unicast communication paradigm, as shown in Figure 7.6, the base station first transmits a wake-up beacon through its wake-up transmitter containing the specific node address that it wants to query. The sink then turns on its main data transceiver and samples the channel for 15.6 ms to receive an incoming packet. On average, the packet transmission takes no more than 8 ms, however, we have over provisioned to improve synchronization between source and sink and to reduce packet losses. After reception, the node either repeats the trigger process with other remote sensing nodes or goes back to sleep until the next samples should be collected, a time we refer to as the trigger-interval.
On the side of the remote sensing node, the WuRX is always in the listening state, waiting for a wake-up beacon from the sink. The receipt of the wake-up beacon containing the targeted node's address generates a HIGH pulse on the GPIO of the microcontroller to trigger the power switch, activating the Tmote. The remote node then sends the data packet to the sink using the CC2420 transceiver on the Tmote. As the power available at the remote sensing node is limited by what is produced by the plant and stored in the capacitor, we avoid exchanging acknowledgements (ACKs) from the sink to the sensor nodes.

Unlike traditional receiver-initiated protocols, our proposed MAC not only eliminates idle listening at the receiving end but also the continuous transmission required at the senders.

**Broadcast Communication.** As an alternative, the initiation trigger can be broadcast, with the intent to wake-up *all* nodes in range. Broadcast based wake-up can reduce data latency w.r.t to addressed based systems since the wake-up receiver does not need to decode the wake-up packet. As such, it can trigger its main node more quickly after receiving the preamble.

Figure 7.7 illustrates our proposal for a broadcast-based WRI-MAC’s broadcast mechanism. As shown, the receipt of a broadcast address, set to 11111111, causes the WuRX to disable address decoding mechanism upon detecting ones in the first 2 bits. This beacon provides loose synchronization among the remote sensing nodes, and all transmit at a different, pre-defined interval from trigger reception. After sending the trigger the base station enters a receiving window within which it accepts all incoming data via its main transceiver.

Using the single beacon saves energy at the sink side as multiple beacon transmissions are expensive. However, the energy saved is compensated by keeping the main radio on at the sink for a longer period. Further, this energy is expected to increase with an increase in the number of sensing nodes. Nevertheless, savings is also expected at the remote plant-powered nodes as the trigger need not be decoded, saving approximately 1 mW for 16 ms of decoding time. In the broadcast mode, no ACK frames are exchanged between the sender and receiver.

**Channel Configuration.** As a final note, our wake-up receiver and main transceiver use different channels, 868 MHz and 2.4 GHz, respectively, eliminating the possibility of collisions between the wake-up beacons and the main data packets.

### 7.5 Experimental Setup and Evaluation

Next, we present a proof of concept, controlled laboratory experiment composed of a sink and a single, remote plant-powered sensing node. We have used data collected from several weeks of monitoring the PMFC and we show multiple data collection experiments for system evaluation. To demonstrate system scalability, we turn to simulation, presenting results for multiple sensing nodes.
7.5.1 Output Power Characterization of PMFC

The earlier studies [224] revealed that the best way to exploit the energy from a PMFC is in a bursty fashion, draining a maximum of 300 $\mu$W for short periods and then letting the cell to recover until it reaches the open circuit voltage.

To characterize and validate the amount of power that can be extracted from the PMFC, we conducted two separate laboratory experiments. In the first, the remote node is triggered, then the plant is given ample time to recover back to the open-circuit voltage of 0.36V. The second represents a stress test to identify the maximum trigger rate that can be sustained with our setup.

The PMFC we used, shown in Figure 7.1, was monitored over a period of six months to observe the correlation between the microbial fuel cell and the plant. Data was collected with a National Instruments DAQ, while the power traces are obtained with an oscilloscope equipped with a custom amplifier to measure the power generated by the PMFC. A Keithley SourceMeter SMU2450 was also used to measure the actual power consumption of the sensing node.

**Single-sample test:** Figure 7.8 captures the instantaneous behavior of both the PMFC and the capacitor during the test, namely a single sample.

During a measurement interval of 100s, the sensing node connected to the PMFC is triggered at $t = 2.5s$, shown by the voltage drop at this time. As soon as the switch is closed the capacitor begins to discharge as the stored energy is supplied to the node. At $t = 3.2s$ the sampling and transmitting task ends and the capacitor voltage begins to recover thanks to the energy harvester that returns to the charging mode resulting in the voltage drop across the PMFC until the capacitor is fully charged at $t = 38s$. There are also periodic dips in the PMFC voltage as observed in Figure 7.8 after the initial charging time, e.g., between 40 to 50s. This is due to the capacitor discharging to supply the power to the on-demand switch. To restore this
7.5. Experimental Setup and Evaluation

Figure 7.8 – Instantaneous voltage profile of the PMFC (lower trace) and the buffer capacitor (upper trace). At the beginning the remote node is triggered and the PMFC is left to recover towards an open circuit voltage.

consumed energy, the charger kicks in until the capacitor is fully charged. After this, the PMFC stabilizes again.

It takes around 35s for the 22 \( mF \) capacitor to be fully charged by the PMFC, indicating the maximum trigger interval that we can achieve. As we can see, once the capacitor reaches 3.36V, the PMFC starts to recover until it stabilizes at \( t = 90s \). We note that the size of the capacitor will affect the charging and the discharging time and the amount of data that can be collected. While larger capacitors may be used for powering the node for a longer period, they will also take longer to charge.

**Stress test:** To evaluate the system under stress, we performed a test with a sampling interval shorter than the minimum, 35s interval identified in the previous section. Our objective was to understand how the plant responds in this stressful condition. We used a triggering interval of 30s. The capacitor and PMFC voltage were monitored over this period.

As shown in Figure 7.9, this triggering interval is sufficient to fully restore the capacitor voltage, however, it is not enough for the PMFC to fully recover and reach the open circuit voltage. Essentially, these stress conditions do not allow the flora and bacteria in the soil to recover.

7.5.2 Observations from Controlled in-lab Experiments

Based on our experiments, we offer a few observations, first on reliability, then on system design.
Figure 7.9 – Voltage profile of the PMFC (red) and the buffer capacitor (blue) during stress test. The activation of the node is also highlighted by a dashed black line every time it was triggered.

To test system reliability, we sent 20 trigger signals with a trigger interval of 2 minutes and successfully received all 20 at the wake-up receiver side. Further, all samples were correctly received at the sink. In our setup, the distance between the remote plant powered node and the sink node was approximately 5 meters, therefore these perfect results were expected. Nevertheless, the reliability of the system with longer distances, e.g., closer to the 30m maximum range of our wake-up receiver should be explored.

While performing our experiments, we did experience minor interference from nearby devices operating in the same frequency band causing false wake-ups. This caused sudden peaks in power consumption due to the PIC micro-controller on the wake-up receiver waking up to decode the RF signal. This potentially jeopardizes the stability of the PMFC, as it requires additional time to recover to the steady state.

Regarding system design, we note that our current system is able to support a reasonable data rate, as long as the sensing node is kept completely off in between samples. The PMFC provides sufficient power to constantly sustain the on-demand switch containing the ultra-low power wake-up receiver. Therefore, there is no need to consider alternative designs to further reduce the consumption of the switch, e.g., duty cycling the wake-up receiver.

Nevertheless, our design reboots the Tmote node to the same state in each interval, without requiring any state to be maintained across iterations. This is possible with our simple, star topology as no routing information is required, but more sophisticated topologies or systems may require additional energy storage capacity.

We evaluated only a single capacitor size, however, this can be modified according to the appli-
cation scenario. For applications that require a higher data rate, multiple, smaller capacitors with shorter charging times can be considered. A larger capacitor may also be preferred to allow the node to be powered for longer, e.g., to receive an acknowledgment and possibly retransmit in the case of loss. Nevertheless, this impacts the time required to charge the capacitors, hence, requiring longer trigger delays.

Further, we recognize that environmental conditions also affect the behavior of the microbial fuel cells. A PMFC in an outdoor environment may generate more energy than the one we exploited, thus accelerating the charging time. There is also the possibility to connect multiple PMFCs in parallel to boost energy production [225].

7.5.3 Scalability Analysis using WaCo Simulator

Our laboratory setting is limited by the available hardware, therefore, we turn to the WaCo simulation environment\(^1\) introduced in chapter 3 to test the scalability of the system, specifically to estimate the power demand and the latency of networks with more than one remote node.

**Simulation set-up:** The distance between the sink and the remote nodes is fixed at 30m. Unit Disk Graph Medium (UDGM) with constant loss is used for the radio channel model due to its simplicity where nodes communicate and interfere in fixed-radius circles. The wake-up radio transmission is also fixed to 30m with a success ratio of 100%. The WuR signals are sent at 1 kbps and can contain 2B of address data. To test in a conservative setting based on our observations during the laboratory experiments, the trigger interval per node is set to 2 minutes. The energy consumption for the Tmote is obtained using Contiki’s Powertrace and Energest power profiler. Results for 30 minute runs are reported in Table 7.2.

In all cases, we report the total consumption at the sink to receive the data from all remote nodes. For latency we report the time at the sink between the beginning of the transmission of the wake-up trigger and the termination of the receipt of the data. For unicast, this corresponds to the time between the transmission of the wake-up beacon directed to a specific node and receipt of data from this node. Instead, in broadcast, as only a single trigger is sent, we start a timer at the sink when the trigger is sent and stop it when all data is received. For 3 nodes, the timer is stopped after receipt of the third sensor data packet.

**Results:** In the unicast mode, the power consumption at the sink linearly increases w.r.t the number of remote nodes. This is expected as the sink must send trigger signals to each node, round robin, to collect data. For each remote node, the average consumption remains constant at 33mW. Latency also remains constant, as the probe to each node remains the same.

In the broadcast mode, consumption at the sink is significantly less for 3 and 5 nodes w.r.t. unicast. This is a combination of the fact that the sink transmits only a single trigger and a tuning of the timing at the remote nodes to minimize the idle listening time at the sink. On the other hand, we see the latency increasing. However, it should be noted that this latency

---
\(^1\)The source code of WaCo is available at https://github.com/waco-sim
Table 7.2 – Scalability analysis using different MAC operation modes. The table shows the power consumption and latency w.r.t network size.

<table>
<thead>
<tr>
<th>Mode</th>
<th>No. of Remote Nodes</th>
<th>Sink (mW)</th>
<th>Remote (mW)</th>
<th>Average Latency (ms)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unicast</td>
<td>1</td>
<td>64</td>
<td>33</td>
<td>685</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>195</td>
<td>33</td>
<td>685</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>325</td>
<td>33</td>
<td>685</td>
</tr>
<tr>
<td>Broadcast</td>
<td>1</td>
<td>64</td>
<td>32</td>
<td>685</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>97</td>
<td>32</td>
<td>780</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>105</td>
<td>32</td>
<td>850</td>
</tr>
</tbody>
</table>

reflects the time to receive the data from all the remote nodes, rather than the single readings, as in the unicast case.

7.6 Conclusion

As a step toward sustainable WSNs, in this chapter, we proposed and experimentally validated a novel system that combines the unique power source of a Plant-Microbial Fuel-Cell to produce the electrical energy for sustaining a standard node coupled with a wake-up receiver. By providing a novel receiver-initiated MAC protocol, WRI-MAC that exploits both the wake-up receiver as well as the main, CC2420 radio on the Tmote, we concretely demonstrate in the laboratory that a 30s sampling rate can be sustained for a single remote node, in a star topology. With the combination of the novel hardware and software techniques proposed in this chapter, we are able to reduce the energy consumption of a standard node by three orders of magnitude, enabling fully energy-neutral operation. Our proposed design is energy-efficient and flexible allowing it to be powered with a wide variety of energy sources such as photovoltaic cells, thermal, and kinetic for various wireless sensing applications. Future work is required to experimentally establish scalability and the capabilities of the whole system outside the laboratory.
The recent Internet of Things (IoT) wave is boosting the development of connected smart low-power devices embedded with various sensors and wireless technologies. The most common IoT devices are sensor nodes that measure physical properties such as vibration, pressure and temperature. For these devices to be “smart”, they are equipped with the onboard processing capabilities to extract useful information from the data before they are sent out to the remote host. These smart devices can make decisions or even take actions by controlling actuators [226]. A common feature of these sensor nodes and, in general, of IoT devices is the capability to transmit data and commands wirelessly.

Most IoT devices are battery powered and expected to have a battery life in terms of years rather than hours. Lifetime is crucial for battery-operated devices, as frequent recharging and replacement will be inconvenient or even impracticable due to the sheer number of IoT devices planned to populate the world. Thus, the design of sensor devices and applications poses several challenges, especially in communication and networking. In fact, the wireless transceiver is one of the most power-hungry subsystems especially when there are many nodes in the network and the required range is on the order of hundreds of meters or even kilometers. Therefore, wireless communication needs to be optimized for improving the energy efficiency and lifetime of IoT devices, as studied in [227].

Today, the most widely-used communication standard for sensor networks in IoT is the IEEE 802.15.4, providing both the physical layer and media access control (MAC) layer specifications [228]. However, one of the main drawbacks of this standard is the limited range of the point-to-point link: up to tens or a few hundreds of meters in the best cases. IoT applications requiring long-range coverage in smart cities and smart environments have pushed the research in wireless technologies enabling novel long-range (LR) communication of several kilometers with power consumption similar to that of standard IEEE 802.15.4 transceivers [229]. The data rate of these long-range chip-sets does of course need to be reduced to achieve greater coverage, but for many low-throughput applications such as environmental sensing, these data rates are adequate.

---

The content of this chapter is a joint work with Michele Magno, Luca Benini, and Amy L. Murphy, published as: **On-Demand LoRa: Asynchronous TDMA for Energy Efficient and Low Latency Communication in IoT**, In Sensors, November 2018, and have been slightly adapted for this thesis.
Chapter 8. On-Demand LoRa: Asynchronous TDMA for Energy Efficient and Low Latency Communication in IoT

LoRa is one of these promising low-power long-range technologies, operating in the 868–915-MHz ISM band allowing bit rates between 0.37 kbps and 46.9 kbps and promising ranges up to 20–25 km [230]. LoRaWAN, the MAC protocol for wide area networks, is based on the ALOHA protocol and is ideal for applications with low-traffic and sporadic communication requirements. One of the features making LoRa attractive is its energy efficiency for uplink communication while achieving a long range. In LoRaWAN, the distributed battery-operated sensor nodes send data directly to an always-on gateway. The energy efficiency comes by duty-cycling the main radio transceiver when not transmitting.

8.0.1 Challenges

LoRaWAN is mainly designed and optimized for uplinks where the remote end devices disseminate data to the gateway using periodic communication or according to a specified radio duty-cycle. On the other hand, to control the end devices, query data from the specific node or reconfigure node parameters require feedback from the gateway and are absolutely crucial in many Internet of Things application scenarios. For instance, in precision agriculture, hundreds of networked nodes are envisioned to increase production by monitoring the climate conditions and controlling the irrigation and lighting systems remotely. However, in the current LoRaWAN architecture, a fundamental trade-off between downlink traffic, latency and power consumption arises. We rely on Figure 8.1 to elaborate on this issue. The figure shows the timing diagram of a LoRaWAN Class A end device. The end device first transmits the data to the gateway and then opens up two reception windows (RX1 and RX2), giving opportunities to the gateway for any downlink communication. The transmission slot is scheduled by the end device based on its own uplink requirements. In this communication scheme, the gateway has no control on the end device, and any downlink communication requires waiting or the next scheduled uplink transmission, thereby increasing latency and adversely impacting responsiveness. This affects applications where both low-latency and low-power consumption are required such as structural health [231] and seismic activity monitoring [232]. One may improve the downlink latency by communicating more frequently, but only at the cost of increasing energy at the end devices.

Moreover, LoRa was originally designed to be used as a large-scale star network. As such, this makes it difficult for the nodes to discover other nodes in the network. Without any information regarding neighbors or their transmission schedule, the chance of creating traffic congestion is non-negligible [233]. In fact, even though the amount of data generated by each
sensing node can be low, the large number of devices trying to access the wireless channel at the same time can be unmanageable by techniques such as pure ALOHA. These limitations are becoming a serious bottleneck for the success of LoRa-based networks in realizing large-scale deployments, such as environmental monitoring, geolocation, smart metering and smart grid. Due to the heterogeneous nature of the IoT devices and the huge number of applications, there is a diverse set of requirements that need to be satisfied. These requirements vary in terms of energy efficiency, device lifetime, latency and reliability. Combining different emergent technologies from hardware to software can enable better performance than relying on a single technology.

8.0.2 Approach

Most IoT monitoring applications have data to report which are sparse in time and infrequent such as temperature and humidity with low-throughput requirements. This work targets such applications where the gateway has full control over the network for data collection and node configurations, shifting the communication modality from push to pull based. To facilitate this on-demand communication, we propose a network architecture that adheres to the following requirements:

- **Energy efficient**: during periods of inactivity, i.e., when there are no data to be transmitted, the end devices must reside in a deep sleep state to maximize device and overall network lifetime.
- **Responsive**: data must be delivered to the gateway and vice versa in both a timely and reliable manner.

While LoRaWAN targets mainly the first requirement, the latter is not considered at all. To improve energy efficiency and at the same time overcome latency and collision issues due to duty-cycling, in this work, we consider the ultra-low-power wake-up radios (WuR). WuR has the capability to monitor the wireless channel continuously while consuming power orders of magnitude lower than commodity radio hardware typically utilized in wireless sensor platforms [182]. Recent WuR designs also perform address matching with micro-watts of power while waking up the main system only when required [234], avoiding false wake-ups. Due to these features, WuR allows “pure” asynchronous communication by activating the system via interrupts only when a specific signal referred to as a wake-up beacon (WuB) is detected from the wake-up transmitter (WuTX). To benefit from this technology, sensor nodes are equipped with an extra wake-up receiver (WuRX) circuitry and put into low-power modes, waiting for a remote trigger signal. The wake-up receiver, which is typically used in an always-on manner, identifies the incoming wake-up beacon from the sender node. Upon detection of the valid wake-up beacon, it triggers the main node out of sleep mode to exchange data “instantly”, thus reducing latency. Since the wake-up receiver detects signals with very low current, it significantly reduces the wasteful power consumption of idle listening, improving energy efficiency without compromising latency for IoT communications.
Contributions. In this chapter, we combine a state-of-the-art wake-up radio with LoRa network technology to fulfill the above requirements. Specifically, we offer the following contributions:

- a new network architecture leveraging short- and long-range technologies for enabling low-latency and energy efficient data collection over a two-hop network.
- the design and implementation of a new receiver-initiated on-demand TDMA MAC for managing channel access and packet collisions. The proposed MAC offers two modalities for node triggering and allows slot allocation for combating packet collisions. On-demand TDMA achieves an improvement of at least $1.72 \times$ in terms of latency and an extended node lifetime of $1.4 \times$ with 100% system reliability over the traditional channel sensing scheme for LoRa. This new feature still works with the standard LoRa, but improves overall performance with an on-demand TDMA scheme.
- introduce an analytical model to quantify the data collection latency for on-demand TDMA in broadcast and unicast mode.
- the performance evaluation of the proposed network architecture and MAC using an indoor testbed composed of 11 sensor nodes.

Structure of this chapter. The chapter presents the relevant background and related works in LPWANs and wake-up radios in Sections 8.1 and 8.2 respectively, then outlines the proposed network architecture and MAC design in Sections 8.3 and 8.4 offers the experimental setup followed by an evaluation and the results of the full system design. Finally, Section 8.6 provides concluding remarks and possible future directions.

8.1 Background

We first give the required background on the wireless technologies and techniques that we exploit in this work to achieve energy-efficient sensing and data collection. We begin with background on LoRa followed by ultra-low-power wake-up radios.

8.1.1 Low-Power Long-Range Communication: LoRa

To extend the IoT device connectivity, recently, a new set of wireless technologies appeared that can attain long-range communication, up to several kilometers, with power demands similar to those of ZigBee devices [235]. These technologies are grouped under the label of low-power wide area networks (LPWANs) and operate at sub-GHz frequencies. To achieve a long range, three main modulation techniques can be used: ultra-narrow band, spread spectrum or narrow band.

The key technology players that are trying to fill the gap in the long-range device-to-device (D2D) communication are SigFox, LoRa, NB-IoT and weightless-N [236]. LoRa has been very successful in the market due its open protocol standard and chirp spread spectrum (CSS) modulation technique that allows recovering data from weak signals even below the noise
8.1. Background

Figure 8.2 – Standard LoRa network topology.

floor. Due to this technique, communication is more resilient to interference and is inherently secure. To control the trade-off between the transmission range and data rate, LoRa allows configuring the radios with different spreading factors (SF) within the range \([230, 235]\). A higher SF increases receiver sensitivity and thus range, but decreases the symbol rate. In this work, we use three different SF settings: SF7, SF9 and SF12, which offer data rates of 21.87 kbps, 7.03 kbps and 0.976 kbps, respectively. Further, communications with different SFs do not interfere with one another and create a set of “virtual” channels, increasing the capacity of the gateway.

The LoRa physical layer can be used with any MAC layer, and currently, LoRaWAN is the only standardized MAC. In the LoRaWAN architecture [237], the network is formed by one or more gateways and multiple end devices organized in a star-of-stars topology, as illustrated in Figure 8.2. The end-device changes the channel in a pseudo-random fashion for every transmission and communicates directly to the gateway, which then forwards the data to the server. LoRaWAN defines three classes of end devices (A, B and C). For Class A, each uplink transmission from the end device is followed by two downlink windows, as illustrated in Figure 8.1. Class A is the most power efficient, as nodes only wake up to send data. Class B devices provide the Class A functionality plus open extra receive window at scheduled times. Class C devices are always listening to the channel, except when they are transmitting.

8.1.2 Wake-Up Receivers

Wake-up receivers are a recent hardware technology to improve the energy efficiency in communication. To achieve ultra-low-power consumption, the wake-up receiver works with on-off keying (OOK)-modulated signals. This drastically simplifies the wake-up receiver circuitry where only a few discrete components are sufficient to construct OOK signal detection circuitry [234, 238]. Wake-up receivers are always-on devices that need to operate at a very low power. As a consequence, they usually have a lower sensitivity and bit rate than standard transceivers used in sensor networks. To extend the range of the wake-up receiver communication, most designs operate in the sub-GHz band [239, 234, 238].

In this work, we employ a state-of-the-art wake-up receiver that consumes only 1.80 µW in continuous listening mode [234]. This wake-up receiver has been tuned for the 868-MHz
Chapter 8. On-Demand LoRa: Asynchronous TDMA for Energy Efficient and Low Latency Communication in IoT

band and has a receiver sensitivity of $-50$ dBm, reaching up to 20 m line-of-sight indoors. To trigger the wake-up receiver, we have exploited the OOK modulation feature of the Semtech SX1276 LoRa module to cross communicate between a LoRa transceiver acting as a wake-up transmitter and a wake-up receiver, both operating in the 868-MHz frequency band. In addition, this wake-up receiver design also provides address decoding capabilities at the cost of a few additional micro-watts, by integrating an ultra-low power microcontroller unit. In this work, we rely on the addressing feature of the wake-up receiver to prevent erroneous activation.

8.2 Related Work

Various IoT applications and wireless protocols use a many-to-one communication paradigm where sensing nodes transmit information to a single node or gateway. We next present some of the existing approaches for many-to-one communication in long-range networks followed by wake-up radios.

8.2.1 Long-Range Communication Schemes

Carrier sense multiple access with collision avoidance (CSMA/CA) is one of the most widely-used MAC protocols in wireless local area and short-range networks for efficient many-to-one data collection. However, as the number of devices in LPWAN increases, the carrier sensing mechanism becomes less effective at reliably detecting channel activity, negatively affecting network performance [240]. Therefore, LPWAN technologies such as LoRaWAN and SigFox have adopted pure ALOHA access protocols for uplink communication [241].

ALOHA is an asynchronous protocol where the end devices communicate when they have data ready to send, either scheduled or event-driven. Several studies have evaluated the scalability of ALOHA using simulations by taking into account the physical and link layer information in terms of latency, reliability and throughput [242, 241, 243, 233]. Most of these studies conclude that although ALOHA operates well under low-traffic loads, and it suffers from uplink traffic congestion as the number of network devices increases due to its inability to check whether the channel is busy before transmitting. Pop et al. [244] also studied the effect of bidirectional communication and its impact on the throughput of LoRa networks. The findings indicate that use of downlink traffic, e.g., data and acknowledgments, can corrupt the uplink packets reducing the overall throughput and network reliability. Therefore, for networks that do not require high reliability, use of acknowledgments and retransmission attempts should be chosen carefully.

Recently, an effort has been made in [245] to increase the scalability for LoRa networks by dividing the available bandwidth into single synchronous downlink channel and several asynchronous uplink/downlink channels. The gateway sends a beacon to synchronize the nodes that also carries the information specifying the allowed transmission power and spreading factors to be used by the nodes for uplink transmission. Nodes upon wake-up listen to the
8.2. Related Work

latest beacon to synchronize the information and then transmit the data in an ALOHA manner. Although the proposed RS-LoRa MAC in [245] mitigates packet collisions by nearly 20% compared to standard LoRaWAN, it does not fully eliminate them as the uplink messages could still collide with beacons from neighboring gateways.

Due to these issues, the ALOHA MAC can become a bottleneck for the success and scalability of LoRa networks as deterministic operation cannot be guaranteed. Motivated by the above state-of-the-art, we propose an on-demand TDMA that provides a deterministic performance in terms of network latency w.r.t. to the network size, which is highly desirable for low-power low-latency applications. Moreover, most of the above analyses are restricted to simulation and consider only homogeneous networks, thus limiting the variety of applications to which LoRa can be applied; for instance, applications that require both short- and long-range capabilities [246] to provide different services. To open up new application scenarios in this direction, we exploit the feasibility of combining heterogeneous networks. Furthermore, instead of taking a simulation approach, we evaluate the full network performance via testbed analysis. To the best of our knowledge, on-demand TDMA is one of the first to test receiver-initiated MACs in a wake-up radio setting with a LoRa-WuR testbed.

To improve bandwidth utilization, some studies have proposed a listen before talk (LBT) procedure before transmission [247, 248] for Class A devices. The LBT mechanism allows multiple users to share the same channel by continuously monitoring the channel so as to transmit only when it is free. LoRa transceivers provide a special mode called channel activity detection (CAD) for channel occupancy by detecting a preamble. In case a preamble is detected, the transmitter backs off for a random period of time, then performs channel sensing again. Although LBT allows a high degree of collision avoidance, it increases the power overhead required by the transmitting nodes for channel sensing, a factor crucial for battery-powered sensor nodes. Moreover, LBT also contributes to packet delays due to the exponential back-off mechanism. On-demand TDMA on the other hand efficiently triggers the nodes for data collection over wake-up radio while packets are tightly packed in the available slots without additional delays. As a result, on-demand TDMA achieves high system reliability without compromising energy efficiency and latency.

8.2.2 Wake-Up Radio-Enabled MAC

To extend the lifespan of batteries, most IoT devices resort to duty cycling mechanisms, where the device periodically wakes up from the sleep mode to retrieve new information. Although this allows saving power, it also introduces limitations. First, the device periodically wakes up even if there are no data to exchange, thus wasting energy. This behavior is known as idle listening. Second, using long sleep intervals increases data latency [190].

To address the above issues, a novel hardware-based technology called wake-up radio (WuR) has gained popularity in the WSN community. A recent survey [182] highlights that major work involving WuR has been on improving hardware design to achieve better communication
characteristics with low-energy demand. To reap the benefits of this new technology, a few studies have started to integrate low-power protocols for remote triggering and on-demand communication. Sparse Topology and Energy Management (STEM) [249], based on TI-MAC, uses a dual-radio approach to differentiate data and wake-up channel while relying on the regular high-power radio as a WuR to achieve coverage up to 20 m. Yang et al. [250] propose a pipelined tone wakeup scheme for data and tone detection using separate radio channels. In PTW, the nodes transmit a broadcast tone before sending data. As the wakeup procedure is broadcast, all nodes within range are triggered. From the point of view of application scenarios, PTW enables fast network-wide wake-up; however, it is less energy efficient, as all nodes, upon wake-up, try to transmit concurrently, causing collisions and energy overhead. In this work, we also take advantage of the broadcast scheme for network-wide wake-up with reduced delay; however, to overcome the issue of packet collisions, we allow each node to transmit at different times using an on-demand TDMA scheme.

A few WuR receiver-initiated (RI) protocols shift this burden to the receiver [251, 123]. In RI systems, the task of communication initiation falls to the receiver node, often the sink or the base station, broadcasting a beacon to either request data or to announce its readiness to receive data. The benefits of adopting a receiver-initiated scheme are the following: (i) no collisions, as the receiver is in charge of pulling the data from the individual sensing nodes when required; (ii) asynchronous communication, due to the fact that RI protocols do not require tight network-wide synchronization, so a pure asynchronous communication can be achieved.

The on-demand TDMA scheme developed in this work adopts a receiver-initiated asynchronous MAC via wake-up radios for time synchronization and data collection, reaping the benefits mentioned above. This allows the proposed MAC to be in full control of when to retrieve data from the nodes without congesting the network while achieving 100% reliability.

Other works have also proposed multi-hop data dissemination and collection using WuRs. Wake-up radio MAC (W-MAC) [252] provides bi-directional functionality to support multi-hop communication and can be used for small or large networks. In addition, W-MAC supports addressable and broadcast signaling for reducing false wake-ups, a feature also supported by on-demand TDMA proposed in this work. On the other hand, ZIPPY [253] and BLITZ [180] both depend on synchronous transmissions over wake-up receivers for network flooding and data dissemination. In contrast, on-demand TDMA achieves network synchronization in a fully-asynchronous manner on the order of tens of microseconds, i.e., 95 µs, as opposed to ZIPPY and BLITZ, which rely on frequent synchronous transmissions.

While various WuR MAC protocols have been designed and tested, recently, an attempt has been made to fuse the short-range WuR with a long-range transceiver reaching up to several kilometers to extend the connectivity of devices. The authors in [254] have presented a hardware design that uses dual radios to reduce power consumption for LoRa devices. Through analytical analysis, the study indicates that using the wake-up receiver can substantially decrease uplink/downlink latency for LoRa networks and increase energy efficiency. Although
the work in [254] has been the starting point in this direction, it does not support data collection from multiple nodes within the star network and has not been experimentally evaluated in a testbed with more than two nodes. In contrast, the current work presents an on-demand MAC protocol that leverages a WuR and a LoRa transceiver with the goal of reliably collecting data from multiple sensor nodes without data collisions, yet maintaining high reliability and deterministic operation. We also go beyond prior works and carry out an evaluation via testbed analysis with multiple sensor nodes. Moreover, we also carry out a systematic evaluation with different LoRa radio settings to explore the benefits and the trade-offs it has on energy and overall data latency.

8.3 Energy Efficient Data Communication Network and Protocol

Next, we outline our proposed network architecture and our on-demand TDMA protocol for LoRa communication to achieve the goal of improving energy efficiency without compromising data latency and network reliability. The network architecture is designed to realize many-to-one and one-to-many communication.

8.3.1 Network Architecture

In this work, we propose a heterogeneous IoT network where the gateway is in charge of collecting data from a specific end device or cluster of devices when required. To maintain the low-power state, the end devices offer a dual-radio interface: a wake-up receiver for short range radio (meters) and a LoRa transceiver for long-range communication (kilometers). With pure on-demand asynchronous communication over the wake-up receiver, the end devices need not periodically or continuously listen to the channel, thus overcoming the issue of idle listening and latency with improved energy efficiency. Unlike LoRaWAN, where a gateway communicates directly to the nodes, in the proposed network architecture, we partition the network into clusters. The clusters are comprised of sensor and actuator nodes that form a star topology, and any down-link communication from the gateway to the nodes (one-to-many) must go through the cluster heads due to the short communication range of wake-up receivers, as illustrated in Figure 8.3.

In detail, the network is composed of three different nodes:

1. The end device (ED) is responsible for sensing and equipped with both a wake-up receiver and a LoRa transceiver. The EDs are battery powered and, therefore, will spend most of the time in a low-power mode, i.e., deep sleep state.

2. The cluster head (CH) is in charge of managing the nodes in the cluster and relaying commands from the gateway to the EDs. Depending on the application scenario and the energy availability, the CH can either operate in a duty cycle mode or can always be listening. The nodes designated as CHs are also equipped with both radios. Each CH is assigned a unique ID address allowing the sink to query each CH at a time, thus
Chapter 8. On-Demand LoRa: Asynchronous TDMA for Energy Efficient and Low Latency Communication in IoT

Figure 8.3 – Proposed heterogeneous IoT topology for low-power wide area sensor and actuator networks.

reducing the interference from other clusters.

3. The sink acts as a gateway and is assumed to have no energy constraints and will be wall-powered. Therefore, the sink can be always on and listening for any incoming data. Unlike ED and CH, the sink only offers long-range communication without the wake-up radio interface.

The communication directionality can differ depending on the node classification. The transmission between the sink and the CH is bidirectional (sink ––> CH), allowing the gateway to update network parameters or collect data from the cluster heads and end devices directly. The CH and the EDs also communicate bidirectionally (ED ––> CH), so that the CH can trigger the EDs and the EDs can query neighboring nodes without passing through the gateway every time. Moreover, in urban scenarios, the presence of obstacles such as buildings or foliage could deteriorate the link quality from the ED to the sink [255]. Under this condition, the EDs could relay the messages to their CH, which is expected to have good link conditions, which could then forward the message to the sink. On the other hand, EDs communicate unidirectionally to the sink (ED ––> sink), maintaining a single-hop communication scheme as in the LoRaWAN architecture.

To improve the energy efficiency of the EDs, we propose an on-demand TDMA scheme (Section 8.3.2). This on-demand data request scheme interrupts the EDs through remote triggering using the wake-up beacon. In this scenario, the sink, which can be located a few kilometers away from the clusters, initiates the data collection by first sending the data request to the CH using the LoRa physical layer. The CH then disseminates this to the EDs by transmitting a wake-up beacon, thus exploiting the short-range feature of the wake-up receivers. Since each ED is equipped with a wake-up receiver, either one or all EDs can be triggered by the CH through an addressed or broadcast-based wake-up beacon. Finally, the EDs respond to the request by offloading data directly to the sink (many-to-one), similar to nodes in the LoRaWAN architecture. The end devices then return to the deep sleep listening state until the next wake-up event is detected.

The proposed architecture together with on-demand TDMA leads to interesting performance
8.3. Energy Efficient Data Communication Network and Protocol

gains for uplink/downlink communication in terms of energy and latency in contrast to the LoRaWAN architecture. Next, we present this on-demand TDMA scheme and its inner workings.

8.3.2 On-demand TDMA MAC Design

One of the main contributions of this chapter is a novel on-demand TDMA: a low-complexity, reliable and energy-efficient data collection scheme that leverages dual radios for the LoRa network in IoT applications. In all cases, our goal is to pull data from all nodes in the most energy-efficient and low-latency manner. Therefore, the core of the proposed MAC is receiver-initiated on-demand communication providing network-wide data collection with reduced packet collisions.

We developed two mechanisms to achieve the above goal, explicitly the two primitives of the wake-up radio, namely unicast and broadcast. Unicast is useful for applications that need to gather data only from specific sensor nodes in the network. For instance, the base station triggering a particular sensor located in a house or on a farm for the latest measurements. As an alternative, the trigger can be broadcast with the intent to activate all sensors in range for gathering data such as the status of all fire alarms on the same floor. Next, we discuss these two different modalities offered by our proposed MAC.

8.3.2.1 Unicast TDMA

In unicast mode, we exploit the addressing feature of the wake-up radio to trigger and pull data from a specific ED. This mode can further be extended to pull data from other EDs or nodes in the cluster by sending multiple unicast requests. Initially, the sink commences the data request by transmitting the command beacon to the CH using its long-range transceiver. The sink then switches to receive mode and samples the channel for any incoming data packet. Since the transceiver at the CH is always on and listening, it receives this request "almost immediately" and then proceeds to transmit the addressed wake-up beacon to the specific sensor node as requested by the sink. As introduced in Section 8.1.2, the always-on wake-up receiver on the ED detects this beacon, decodes the embedded address and transitions the main MCU out of deep sleep. The ED then transmits the data packet to the sink using its long-range SX1276 transceiver. After the reception, the sink either repeats the request process with other EDs using the unicast procedure or can fall back to continuous listening mode. The benefit of this modality is that the unicast TDMA not only eliminates the idle listening at the sender (ED) and the receiver (sink), but also reduces the downlink data latency (sink → ED). This reduction is achieved as the sink does not have to wait until the next uplink transmission to acquire fresh data as opposed to LoRaWAN specifications.
8.3.2.2 Broadcast TDMA

In contrast to unicast mode, here, we leverage the broadcast feature of the wake-up radio to trigger and pull data from all the nodes in the cluster using a single broadcast beacon reliably and efficiently. The goal is two-fold. First, the broadcast beacon allows waking up all the nodes in range for data collection reducing latency w.r.t. to the unicast mode where several addressed wake-up beacons are transmitted for activating each different node. Second, transmitting a single beacon saves energy both at the sink and the CH side as multiple beacon transmissions are expensive. We will quantify these savings when presenting our findings under the results in Section 8.5.3. However, the former introduces collisions, as all the nodes may try to transmit data concurrently upon successful wake-up, causing network congestion. To mitigate this, we have implemented a simple, yet robust slot scheduler atop RI-MAC, thus the name on-demand TDMA. The slot assignment per node allows full utilization of the bandwidth while reducing the channel activity and eliminating packet collisions in the network. We show experimentally in Section 8.5.1 that a system reliability of 100% is achievable using the proposed on-demand TDMA technique.

Figure 8.4 exemplifies the operation of on-demand TDMA using short-range and long-range radios. Similar to the unicast case, the sink initiates communication via the CH. This time, the CH transmits a wake-up beacon with a network broadcast address for activating all the EDs within the cluster. After sending the trigger command, the sink opens a receive window within which it accepts all the incoming data from the EDs via its LoRa transceiver.

For proper operation of the on-demand TDMA, all nodes in the cluster need to agree on the time slots; therefore, clock synchronization is required. We leverage the wake-up beacon to provide a fine-grained time synchronization among the EDs achieved by the asynchronous network wake-up, marking the start of an epoch. To measure this synchronization accuracy,
we programmed the two end devices to assert their specific GPIO pin after receiving the correct wake-up beacon. Figure 8.5 illustrates the times at which these end devices raise their GPIO pins. The difference is measured as 95 µs, indicating that clock synchronization on the order of tens of microseconds is achievable using wake-up radios without requiring complex time synchronization algorithms.

![Figure 8.5 – Time synchronization between two end devices over the wake-up receiver.](image)

### 8.3.2.3 Slot Allocation

All the EDs are now active and synchronized. First, each ED enters into a slot reservation phase where the EDs occupy the slot according to their node ID, \(N_{id}\), i.e., the node with ID 1 occupies the first slot, while the node with the maximum ID occupies the last slot. In the current implementation, node IDs are statically assigned a priori at network configuration time. The ED measures the exact time when it was triggered by the CH, denoted as \(WuB\text{Arrival Time}\). Each node then computes the start of its time slot from the \(WuB\text{Arrival Time}\) as:

\[
T_{Next\text{Slot}} = WuB\text{Arrival Time} + (ToA_{pkt} + G_t)N_{id}
\]

\[
ToA_{pkt} = \left( n_{pre} + 4.25 \right) \frac{1}{R_{sym}} + \left[ 8 + \max\left( \left\lceil \frac{8PL - 4SF + 28 + 16CRC - 20H}{4(SF - 2DE)} \right\rceil \right) (CR + 4) \right] \frac{1}{R_{sym}}
\]

The slot size is determined by computing the time on air, \(ToA_{pkt}\), using Eq 8.2 for the LoRa data packet depending on the payload size with a pre-defined guard time, \(G_t\) of 6 ms where:

- \(n_{pre}\) is the programmed preamble length
- \(R_{sym}\) is the symbol rate
- \(PL\) is the payload length in bytes (1-255)
- \(SF\) LoRa spreading Factor (6-12)
- \(H\) is the packet header, 0 when the header is enabled and 1 when no header is present
- \(DE\) data rate optimizer, 1 when enabled, 0 otherwise
- \(CR\) is LoRa Coding rate [1: 4/5, 2: 4/6, 3: 4/7, 4: 4/8].
The guard time guarantees that the window is large enough for the transmission and compensates for clock drift, which may be detrimental with an increasing number of EDs. Finally, the EDs start transmitting the data packets over the LoRa module according to the slot schedule, as illustrated in Equation (8.1).

Moreover, to abate extra delay and transmission consumption, no acknowledgment frames are exchanged between the sender and the receiver. For systems that require 100% reliability at the expense of latency, an acknowledgment can be easily incorporated into our protocol to retransmit in case of packet loss. The above two proposed modalities are evaluated next through experiments, and the results are presented in Section 8.5.

8.4 Evaluation Methodology

This section presents the in-field results of the on-demand TDMA protocol evaluated through a test-bed analysis. Next, we describe the details of the experimental setup, evaluation metrics, and the results.

8.4.1 Testbed Setup

To show the proof of concept, we deployed a total of 11 wireless sensor nodes in an indoor office environment based on the prototypes previously developed by some of the authors of this work [256]. Moreover, since the wake-up receiver hardware is custom designed to work with the LoRa module, a small-scale testbed was the most feasible solution. Of the 11 nodes, 9 are designated as end devices (EDs) responsible for sensing, while one acts as the cluster head and one as a sink. The hardware architecture of the wireless multi-sensor dual-radio platform is shown in Figure 8.6a and includes the following main components: microcontroller, radio transceiver, wake-up receiver, sensors and the power management unit. The platform integrates a multi-purpose SX1276 wireless transceiver from Semtech Corporation that supports different modulations such as (G)FSK and OOK, as well as the LoRa physical layer. To use it as a wake-up transmitter, the SX1276 is configured for transmission using OOK modulation where the information is sent using ‘1’s and ‘0’ s. An OOK 1 sub-bit is produced by transmitting a large amplitude carrier, while an OOK 0 sub-bit is produced by sending nothing, i.e., the transmitter is turned off.

Together with the SX1276 transceiver, each end device is also equipped with a wake-up receiver coupled to an 8-bit ultra-low power PIC12LF1552 MCU for selective triggering. The wake-up receiver consumes 1.80 µW in listening mode and 284 µW when it is actively receiving and decoding the preamble or address. The maximum bit rate supported by the wake-up receiver is 1 kbps. Finally, the platform uses two separate antennas, one for the main transceiver and the other for the wake-up receiver. The nodes are equipped with off-the-shelf 868 MHz 1/2 wave dipole antenna. The antenna orientation of the transmitter and the receiver is oriented at 0°, i.e., facing upwards as illustrated in Figure 8.6b. Table 8.1 depicts the power consumption of the various modes of the platform measured in-lab with a N6705B DC Power Analyzer.
8.4. Evaluation Methodology

Although, all the nodes are equipped with the LoRa and the wake-up receiver, in our evaluations, only the EDs use both radio interfaces. The sink node acting as the base station is connected to the laptop to log data and to verify transmissions. Figure 8.7a captures the testbed deployment, while Figure 8.7b illustrates the topology used in the experiments with different node designations.

8.4.2 Radio Settings

In LoRa networks, switching to a different data rate affects the receiver sensitivity. At high spreading factors, i.e., low data rate, LoRa packets can be received at a much longer range with high reliability. To explore this, we selected three different LoRa radio configurations as summarized in Table 8.2. With Setting 1 (SET 1), we chose the most robust transceiver setting with the lowest data rate (0.976 kbps) and highest spreading factor, leading to the transmission time of 264 ms. Setting 2 (SET 2) was chosen to represent a mid-range data rate of 7.03 kbps with time-on-air of 31 ms, while Setting 3 (SET 3) represents the shortest airtime of 9 ms with bit rate of 21.87 kbps.

All experiments were performed with an 8-byte application payload, a 2-byte wake-up address and the guard time $G_t = 6$ ms. The value of $G_t$ was chosen such that more data slots can be fitted per epoch without overlapping the slots. The transmission power for LoRa and the
Chapter 8. On-Demand LoRa: Asynchronous TDMA for Energy Efficient and Low Latency Communication in IoT

(a) Testbed setup
(b) Testbed topology

Figure 8.7 – Network topology and indoor office testbed deployment.

Table 8.2 – Three different LoRa radio settings (SET) (low, medium, and high data rate) used in our testbed experiments.

<table>
<thead>
<tr>
<th>LoRa Radio Setting</th>
<th>SET 1</th>
<th>SET 2</th>
<th>SET 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spreading Factor</td>
<td>12</td>
<td>9</td>
<td>7</td>
</tr>
<tr>
<td>Coding Rate</td>
<td>4/6</td>
<td>4/5</td>
<td>4/5</td>
</tr>
<tr>
<td>Bandwidth (kHz)</td>
<td>500</td>
<td>500</td>
<td>500</td>
</tr>
<tr>
<td>Data Rate (kb/s)</td>
<td>0.976</td>
<td>7.03</td>
<td>21.87</td>
</tr>
<tr>
<td>Transmission Power (dBm)</td>
<td>10</td>
<td>10</td>
<td>10</td>
</tr>
<tr>
<td>Payload (B)</td>
<td>8</td>
<td>8</td>
<td>8</td>
</tr>
<tr>
<td>Preamble Length (symbols)</td>
<td>8</td>
<td>8</td>
<td>8</td>
</tr>
<tr>
<td>Carrier Frequency (MHz)</td>
<td>868</td>
<td>868</td>
<td>868</td>
</tr>
<tr>
<td>Time-on-air (measured (ms))</td>
<td>264</td>
<td>31</td>
<td>9</td>
</tr>
</tbody>
</table>

OOK signal was fixed at +14 dBm. In contrast to LoRaWAN, in all our experiments, the LoRa radios were configured to use a single channel for communication. Therefore, to provide a fair comparison, we have evaluated the performance of the on-demand TDMA against the LBT protocol with random backoff. The random backoff interval for LBT is restricted to be between {0, 2} s.

8.4.3 Ambient Noise Floor

Before starting an experiment, the noise floor of the ambient environment was measured using the NooElec software-defined radio to check if there were any nearby devices operating in the 868-MHz band. This can be detrimental as both the radios, LoRa and the wake-up receiver used in this work are operating in this frequency band, and any noise on the channel can cause false wake-ups or interference. Figure 8.8 shows the noise activity for the measured band together with the waterfall plots captured from the SDR software. The plot in Figure 8.8a indicates no interference or devices operating nearby in this frequency. As a comparison, Figure 8.8b shows the noise floor during LoRa transmissions with a peak near the center frequency.
8.5. Network Performance Analysis

(a) Noise floor scan before starting experiments.

(b) Noise floor during LoRa transmissions.

Figure 8.8 – Measurement of noise floor in the office testbed environment.

8.4.4 Evaluation Metrics

To evaluate the performance of the proposed protocol, we defined three metrics: (i) packet delivery ratio (PDR), representing how reliable the protocol is, computed end-to-end from the sink to the sensing node. To detect the missing packets, each payload is sent with an increasing sequence number. If the sequence number is seen at the sink, the data packet is counted as received. (ii) Round-trip time latency (RTT) is measured as the time difference from the initial trigger by the sink and the time to receive the data from all the sensing nodes. For instance, in an application such as alarm generation, reducing end-to-end latency is a key objective. (iii) The radio duty-cycle (RDC) is computed as the ratio of time where the radio is active and the interval between two successive wake-ups: \( RDC = (t_{TX} + t_{RX})/T \), where \( t_{TX} \) and \( t_{RX} \) are the time when the radio is on either transmitting or receiving/listening and \( T \) is the total length of the period, expressed in milliseconds. This is a platform-independent metric indicating how energy efficient the protocol is; (iv) Energy consumption is computed as the total energy consumed by the network nodes including the consumption of the wake-up receiver to collect all the pieces of data.

All metrics have been computed based on 500 packets exchanged between the nodes using the static configuration as shown in Table 8.2 over three different trials. We also show the preliminary results of the scalability test and its impact on latency and power in the next section.

8.5 Network Performance Analysis

Next, we present the results obtained from the testbed experiments. Here, we concretely demonstrate the performance in terms of network reliability, latency and energy efficiency of the on-demand TDMA when integrated into the proposed network architecture. For all the experiments, we vary the network load by exploring a range of inter-packet intervals (IPI) from 10 s–60 s. For all the evaluation metrics, we first compare the broadcast TDMA against listen
Chapter 8. On-Demand LoRa: Asynchronous TDMA for Energy Efficient and Low Latency Communication in IoT

Figure 8.9 – Reliability and latency evaluation using different LoRa radio setting and protocols.

8.5.1 Packet Delivery Ratio

We begin our evaluation by considering the data collection reliability of the network, shown in Figure 8.9a, in terms of generated data packets and those successfully received by the sink. The vertical bars indicate the maximum and minimum PDR achieved in each trial. We first observe the PDR of 100% for broadcast TDMA for varying IPIs. The higher PDR is attributed to the facts that: (i) each ED transmits in its own allocated slot, reflecting the ability to handle higher bandwidth without collisions; (ii) the distance between the nodes in our experiments is small; therefore, all the trigger samples were correctly received by the wake-up receivers.

On the other hand, the average PDR for the LBT protocol varies between 83% and 91% for varying traffic. This is because the CAD feature is only useful to detect LoRa preamble symbols. As CAD is not able to detect all transmissions, especially when the preamble has been already sent, this causes packet collisions at the receiver, dropping some packets, resulting in lower reliability. Thus, for the same IPI, different LoRa radio settings could affect the PDR. If the LoRa setting has a long preamble and short data packet, then CAD would be more effective.

8.5.2 Latency

Next, we turn our attention to the network latency as shown in Figure 8.9b. In this set of experiments, all the nodes in the network used the same configuration $\text{Setting} = \{SF, BW, CR, CF, TP\}$. We compared the three different LoRa radio configurations and their impact on the overall network latency using both the protocols, on-demand TDMA and LBT, and two modalities of on-demand TDMA. Since the on-demand TDMA offers a level of determinism that is not attainable with pure ALOHA or LBT MAC protocols, the data latency can be modeled. For latency, the round-trip time is computed as the difference between the sink node transmitting the command packet and receiving all the data from the end devices. These delays include the overhead of 17 ms required by the wake-up receiver to receive a 2-byte wake-up address at
the data rate of 1 kbps, decode the packet and generate a trigger to the main microcontroller. For the unicast mode, the latency, $L_{Ucast}$, is given by Equation (8.3), while for the broadcast mode, $L_{Bcast}$ with slot assignment is expressed in Equation (8.4), where $t_{(S\rightarrow CH)}$, $t_{(CH\rightarrow ED)}$, and $t_{(ED\rightarrow S)}$ is the time required to send the command from the sink to the CH, relaying the trigger from the CH to the ED and transmitting the data packet to the sink, respectively, according to the number of EDs, $N_{nodes}$, in the cluster.

$$L_{Ucast} = \left[ t_{(S\rightarrow CH)} + t_{(CH\rightarrow ED)} + t_{(ED\rightarrow S)} \right] N_{nodes} \tag{8.3}$$

$$L_{Bcast} = t_{(S\rightarrow CH)} + t_{(CH\rightarrow ED)} + [ToA_{pkt} + G_t] N_{nodes} \tag{8.4}$$

**Broadcast TDMA vs. LBT.** For both protocols, the latency increases linearly with the increasing number of EDs for each setting as captured in Figure 8.9b. For higher numbers of nodes, however, on-demand TDMA compares favorably to LBT because nodes are able to send packets without collisions in their own slots. The only extra delay that occurs per transmission is the 6-ms guard time used to compensate for clock drifts. On the other hand, nodes using LBT must compete for the medium and backoff whenever the other node is transmitting. The competition for medium is low with fewer end devices, but as the network size increases, so does the latency due to frequent backoffs. For the highest data rate scenario with the shortest packet transmission time, i.e., SET 3, the latency of LBT is $1.65 \times$ higher than broadcast TDMA, and the performance difference is even more significant for SET 1 and 2, with more nodes and a longer transmission time. For instance, to collect data from 9 nodes, LBT requires $1.72 \times$ longer latency than broadcast TDMA for SET 1.

**Unicast TDMA vs. Broadcast TDMA.** We now turn to the different modalities of the on-demand TDMA to explore the scalability of the system in terms of latency when pulling data from all the sensing nodes. The RTT latency is plotted in Figure 8.9b and also reported in Table 8.3 showing different modes w.r.t. the number of EDs. In the unicast TDMA, the latency linearly increases w.r.t. the number of EDs, independently of LoRa radio settings. This was expected as the sink must transmit a request each time it wants to acquire data from an individual ED and then wait for the data packet, before doing the same with the next node and so on. If the data rate is increased, i.e., SET 2 and 3, unicast mode takes $\approx 1.65$ s and $\approx 1.3$ s, respectively, to collect data from a network size of 9 nodes, whereas broadcast mode provides a dramatic improvement in latency of $3.4 \times$ and $4.7 \times$, respectively. This indicates that if the packet transmission time is reduced, the delay performance of broadcast mode is significantly improved, as this causes a proportional decrease in the TDMA frame duration, which is dependent on the time-on-air of the packet. With a lower number of nodes and depending on the settings used, the broadcast mode is capable of collecting data almost every second if necessary.
8.5.3 Energy Efficiency

Next, we study the energy profile of the on-demand TDMA MAC using radio duty-cycle as a metric to indicate the time radios spend in listening or transmitting. In our evaluation, we took into account the radio listening and transmission timings for the LoRa transceiver, as well as the wake-up receiver module. For LPWANs, the most power-hungry process is radio communication; thus, any optimization to extend battery lifetime shall focus on reducing transmission and reception timings, which impacts the overall energy. The radio on times largely depend on the data rate, as well as on the data size. For instance, with a low data rate, the radio can be on for a longer period of time to transmit the same amount of data w.r.t. to the higher data rate. As for most applications, one cannot control the amount of data generated by the sensing devices; thus, selecting the appropriate data rate would provide significant improvements. To this end, we explore the different data rates and their effects on the power using three different configurations.

**Broadcast TDMA vs. LBT.** Figure 8.10a illustrates the mean radio duty cycle ratio of an ED for both protocols. We note that in all scenarios, broadcast TDMA yields lower RDC than LBT. This is attributed to the fact that each node transmits within its specified slot without any extra transmission, allowing nodes to sleep as much as possible. The nodes also do not perform any carrier sensing, thus minimizing idle listening. On the other hand, the nodes employing LBT must check the channel each time before transmission, leading to extra radio activities.

To this end, we see that broadcast TDMA can significantly improve radio listening and transmission time should it allow EDs to use even lower data rates, still abiding by the duty cycle restrictions imposed by LoRaWAN [241]. The radio activity further reduces at a higher data rate with SET 2 and 3. With longer IPIs, the sleep state begins to dominate the system; hence, the RDC of both the protocols start merging as radios spend most of the time in off mode.

Furthermore, we also estimate the end device lifetime to compare the performance of the
Table 8.3 – Scalability analysis using different on-demand TDMA operation modes w.r.t. to latency, power, and network size.

<table>
<thead>
<tr>
<th>LoRa Radio Setting</th>
<th>SET 1</th>
<th>SET 2</th>
<th>SET 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>TDMA Mode</td>
<td>No. of EDs</td>
<td>Sink (mJ)</td>
<td>CH (mJ)</td>
</tr>
<tr>
<td>Unicast</td>
<td>1</td>
<td>65</td>
<td>36.4</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>325</td>
<td>182</td>
</tr>
<tr>
<td></td>
<td>9</td>
<td>585</td>
<td>327.8</td>
</tr>
<tr>
<td>Broadcast</td>
<td>1</td>
<td>65</td>
<td>36.4</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>119</td>
<td>90.4</td>
</tr>
<tr>
<td></td>
<td>9</td>
<td>173</td>
<td>144</td>
</tr>
</tbody>
</table>

broadcast TDMA against LBT. On the power consumption side, the addition of the wake-up receiver adds an extra 1.8 µW in continuous listening mode. All these additional costs have been accounted for when calculating the end device lifetime. Device lifetime is a critical metric, as it directly affects the network lifetime. Based on the in-lab power measurements presented in Table 8.1, the evaluated sensor node draws an average current of 0.56 µA, leading to a theoretical standby time of 244 years on a 1200-mAh lithium polymer battery, only if batteries could last that long. Next, we estimate the end device lifetime when it is actively participating in data collection rounds with varying packet intervals. As seen in Figure 8.10b, nodes employing the broadcast TDMA scheme can last up to 3 years when polled every minute for data collection. This directly translates into energy savings due to a low radio-on time required by the broadcast TDMA compared to LBT to transmit the same amount of payload (see Figure 8.10a). Even with shorter trigger intervals, i.e., every 10 s, broadcast TDMA provides a lifetime improvement of up to 1.4 × in contrast to LBT. On the other hand, the lifetime of the sensor node is less than a year for the lowest data rate setting w.r.t. both protocols because of the extra power consumed when the radios are active for a long period. Overall, broadcast TDMA demonstrates significant gains in energy that can be achieved over channel sensing schemes for long-range networks.

**Unicast TDMA vs. Broadcast TDMA.** Next, we demonstrate the effectiveness of our proposed MAC by providing a quantitative assessment of the energy consumption to collect data from all the nodes using unicast and broadcast modes. To do this, we conducted testbed experiments where the number of the EDs in the network varied between 1 and 9 with three different LoRa radio settings following the topology shown in Figure 8.7. In all cases, we report the total energy consumption at the sink to receive the data from all the EDs. The power consumption of the CH together with the EDs is reported in Table 8.3.

In unicast mode, the energy consumption at the sink and the cluster head linearly increases w.r.t. the number of EDs across all the LoRa settings. This is expected as the sink must send request signals via CH to each ED in a round robin fashion to collect data from all the nodes, spending around 65 mJ per round. A similar trend is also observed at the CH for receiving and relaying the request commands from the sink to the EDs.

In broadcast mode, consumption at the sink is significantly lower for 5 and 9 nodes w.r.t. unicast. This is a combination of the fact that the sink transmits only a single trigger to pull
data from all the EDs and the fine-tuning of the idle listening time at the sink. The performance gap is more noticeable when the data rate is high, where the sink and the cluster head consume $5 \times$ less energy than unicast trigger to collect data from a network of 9 nodes, i.e., SET 3. The average consumption per ED remains constant per each setting, i.e., 46 mJ for SET 1, as each ED still needs to decode an address embedded in the wake-up beacon for both modes. The extra energy overhead of the protocol is the idle listening at the sink due to the guard time and the continuous listening consumption at the CH. To achieve a data latency in order of a few seconds, this is the trade-off that on-demand TDMA makes by keeping the CH always on for reducing the delay time of the command from the sink to the end devices. In the future, we plan to address this issue by adopting a duty-cycling mechanism at the CH to achieve an energy consumption close to that of end devices.

### 8.6 Conclusions

This chapter presents a new network architecture and an on-demand TDMA MAC protocol leveraging short-range wake-up radios and a LoRa physical layer. On-demand TDMA provides efficient broadcast and unicast service for data transmission and collection, improving the performance of LoRa networks. This work is a stepping stone towards the goal of achieving energy-efficient, yet responsive communication using long-range technology, giving the gateway full control of the network. This is a problem with the LoRaWAN architecture where gateways have minimal or no control over the network and inability to communicate with the end devices upon demand, as discussed in Section 8. The proposed architecture overcomes this drawback by facilitating on-demand triggered communication, where the gateway communicates with the end devices when there are data to be collected. When there is no network activity, the end devices reside in a deep listening state while continuously listening to the wireless channel using an ultra-low power wake-up receiver. The cost of the resulting system, however, slightly increases due to the addition of this extra wake-up receiver module. Nevertheless, on-demand TDMA supports the standard LoRa protocol and can be easily integrated into the LoRaWAN framework for downward data collection.

It has been shown from the testbed experiments that on-demand TDMA significantly improves system scalability and energy efficiency by offering network reliability of 100% with end devices dissipating only 1.83 µW of power during periods of inactivity. We also observed that different LoRa transceiver settings can have significant variations in airtime for a LoRa data packet. Thus, the selection of communication parameters has a tremendous impact on the scalability of a LoRa deployment. While the experimental setup can differ significantly in terms of network architecture, platform design and deployment conditions, our proposed protocol supports a node wake-up delay on the order of milliseconds with a round-trip latency below a second through a two-hop network while sustaining sensing nodes for up to three years.
Wake-up radio is a promising new hardware technology that has shifted the decade-old research in the WSN community from duty cycling to on-demand asynchronous communication. However, this technology is still in its relative infancy and the exact potential of this technology on the WSN system as a whole has not been fully demonstrated.

The goal of this thesis was to demonstrate the potential and benefits of the wake-up radio technology over software-only techniques in alleviating the energy costs of battery-powered wireless sensor nodes. We proposed and evaluated novel techniques and protocols all the way from low-level hardware to the full software stack to strip off the communication costs. Specifically, we started by presenting a comprehensive literature review of the research progress in wake-up radio hardware and relevant networking software. This allowed us to identify the emerging applications that can benefit from wake-up radio technology together with the issues and challenges that need to be addressed before this technology can become an indispensable part of WSNs.

To initiate a standardized evaluation methodology, we proposed WURBench, a set of tools and practices to follow when benchmarking wake-up radio based systems. The goal of WURBench is to make experiments repeatable and results directly comparable whether in simulation or testbed at the same time enabling fair comparisons between new and existing approaches.

As a first concrete step toward benchmarking and assessing the wake-up radio technology, in Chapter 3, we presented WaCo, a simulation tool that accurately models the wake-up hardware and allows measuring the performance in combination with the upper layers of the stack. The simulation results with a realistic physical model and networking stack showed that wake-up radio approach considerably improved the lifetime of the network while maintaining high reliability and low end-to-end latency than duty-cycled protocols. Moreover, WaCo proved to be a valuable instrument for assessing the wake-up radio technology and evaluating the attainable trade-offs between application requirements and robustness of the network, prior to the actual in-field deployment.

To quantify the benefits of the wake-up radio in a realistic distributed environment, we then created a small-scale desktop testbed using a cutting edge wake-up radio hardware integrated
to a standard mote. The testbed experiments confirmed our simulation results. However, results also suggest that harsh in-band interference can be detrimental to the performance of low-complexity wake-up receivers if not properly tamed.

As a step toward sustainable WSNs, we proposed a novel hardware architecture combined with software techniques to reduce the overall node power consumption to a point where an extremely low-power source can sustain the nodes operation. We considered three innovative techniques across the spectrum, namely, an extremely low-power switch composed of a wake-up receiver for power management at the hardware side, a new receiver-initiated MAC-level communication protocol at the networking side, plus a microbial fuel cell at the power source side. The combined effect resulted in a power reduction by three orders of magnitude, enabling fully energy-neutral operation.

To further exemplify the benefits of the wake-up radio, we then integrated it to the long-range (LoRa) network to extended the node lifetime at the same time improving the downlink latency. To fully exploit the wake-up radio, we presented a new network architecture and an on-demand TDMA MAC protocol that leverages short-range wake-up radios and a LoRa physical layer. On-demand TDMA provides efficient broadcast and unicast services for data transmission and collection, improving the performance of LoRa networks. This work is a stepping stone towards the goal of achieving energy-efficient, yet responsive communication using long-range technology. We facilitated this by giving the gateway full control of the network where it communicates with the end devices when there are data to be collected. During periods of inactivity, the end devices reside in a deep sleep state while continuously listening to the wireless channel using a wake-up receiver. By moving away from the realm of pure ALOHA communication to wake-up receivers, we were able to exploit the low power modes of the sensor node more effectively.

Through these contributions, this thesis pushes forward the applicability of ultra-low power wake-up radios, by quantitatively measuring the trade-offs, energy efficiency, reliability, and latency. At the same time, this thesis opens several avenues for future research, briefly characterized in the following.

In this thesis, we only looked at the always-on approach for the wake-up radios. Another promising technique to use is wake-up radio duty-cycling. Using this technique, how many orders of magnitude improvement in the lifetime of the entire network can be achieved in real deployment is still an open question.

Present wake-up radios have a shorter communication range than the high power radios, making it difficult to align coverage of these two radios. Therefore, adaptive relaying over wake-up radios to create a multi-hop “underlay” in order to remotely wake-up the distant high power radio is a promising research direction and is also a subject of our on-going research. It is also in our immediate plans to integrate the wake-up radios to the 50-node testbed in the facility to thoroughly compare the performance and trade-offs that wake-up radio offers. While we showed throughout in this thesis that the wake-up radio works for small networks, it
has not been experimentally validated if the same holds for the large networks.

The hardware experiments conducted in this thesis were using a relatively old Tmote Sky platform that consists of an MSP430 microcontroller and a CC2420 radio chip. Newer platforms build using ultra-low-power MCUs and system-on-chip designs will facilitate even better properties in terms of power and timing they provide.

The recent standardization efforts by the IEEE 802.11 WUR Study Group aims to integrate the wake-up radios with the 802.11 Wireless LAN networks [257]. If the standardization process is successful, it will likely lead to the commercialization of the wake-up receivers featuring receiver sensitivity of at least -80 dBm with a power dissipation of less than 100 µW. This will be 1.6 times higher receiver sensitivity w.r.t. to the wake-up receiver adopted in this work, providing coverage similar to that of the main radio with a justifiable increase in the power dissipation. Off-the-shelf availability of this technology will reduce the time to design and prototype wake-up radio systems, fostering the novel development of energy-efficient applications for the future Internet of Things. Moreover, a possible future step may be to not only use the wake-up radio as a trigger device but also to transmit a few bytes of data for sensing and control.
Bibliography


Bibliography


<table>
<thead>
<tr>
<th>No.</th>
<th>Author(s)</th>
<th>Title</th>
<th>In:</th>
<th>Volume Issue</th>
<th>Pages</th>
</tr>
</thead>
<tbody>
<tr>
<td>108</td>
<td>Chunlong Guo, Lizhi Zhong, and Jan M Rabaey</td>
<td>“Low power distributed MAC for ad hoc sensor radio networks”</td>
<td>Global Telecommunications Conference, (GLOBECOM’01)</td>
<td>Vol. 5</td>
<td>2944–2948</td>
</tr>
<tr>
<td>113</td>
<td>S Mahlknecht and M Spinola Durante</td>
<td>“WUR-MAC: energy efficient wakeup receiver based MAC protocol”</td>
<td>Proceedings of the 8th Conference on Fieldbuses and Networks in Industrial and Embedded Systems (IFAC)</td>
<td>Vol. 42. 3</td>
<td>2009, pp. 79 –83</td>
</tr>
<tr>
<td>119</td>
<td>J. Blanckenstein, J. Klaue, and H. Karl</td>
<td>“Energy efficient clustering using a wake-up receiver”</td>
<td>18th European Wireless Conference</td>
<td></td>
<td>Apr. 2012, pp. 1–8</td>
</tr>
</tbody>
</table>


Bibliography


Bibliography


B. Reynders et al. “Improving Reliability and Scalability of LoRaWANs Through Lightweight Scheduling”. In: IEEE Internet of Things Journal 5.3 (June 2018), pp. 1830–1842.


