Poster: Enabling Contiki on Ultra-wideband Radios

Pablo Corbalán, Timofei Istomin, and Gian Pietro Picco
University of Trento, Italy
{p.corbalanpelegrin, timofei.istomin, gianpietro.picco}@unitn.it

Abstract

We present our Contiki port supporting the ultra-wideband (UWB) DW1000 transceiver directly on the popular DecaWave EVB1000 platform. Our port enables us to easily adapt and leverage the results of a decade of research on wireless sensor networks (WSNs), while exploiting the specific capabilities of UWB. Experimental results with our port confirm decimeter-level ranging accuracy as well as the feasibility of running unmodified Contiki Collect atop UWB.

1 Introduction and Motivation

Ultra-wideband (UWB) communications (concisely described in §2) are becoming increasingly popular thanks to the off-the-shelf and IEEE 802.15.4-compliant DecaWave DW1000 transceiver, which provides centimeter-level ranging (i.e., distance estimation) at a low-power consumption and low cost. As a result, UWB is now been considered a candidate radio technology for many wireless sensor network (WSN) applications, especially those that require or benefit from precise distance estimation and localization capabilities. Nonetheless, WSNs have traditionally focused on narrowband IEEE 802.15.4 radios, raising the question of how much of the last decade research outcome is actually applicable to UWB radios and can be re-used without reinventing the wheel. To address this question, the Contiki operating system [1] can be regarded as a key enabler, as it has received wide adoption from academia and industry, and supports several standard networking protocols such as 6LoWPAN, RPL, and TSCH. The barrier, however, consists in the inability to directly run Contiki atop off-the-shelf UWB platforms.

In this poster, we introduce our Contiki port for the DecaWave EVB1000 platform [2] based on the DW1000 UWB radio. Although there are custom DW1000-based designs that have been ported to Contiki [3], we chose the EVB1000 as it is commercially-available, ready-to-use, and arguably the most popular platform to get started with UWB. We describe the thin layers (see Figure 1) of our port architecture in §3, and present experimental results with our port in §4, confirming decimeter-level ranging accuracy, promising data collection results with unmodified Contiki Collect in a university testbed, and also accurate localization in a co-located demo for autonomous rover navigation [4]. We argue that our port can become an enabler for research revolving around UWB radios as well as help speed up the development of UWB-based prototypes and products. Our port is available at https://github.com/d3s-trento/contiki-uwb.

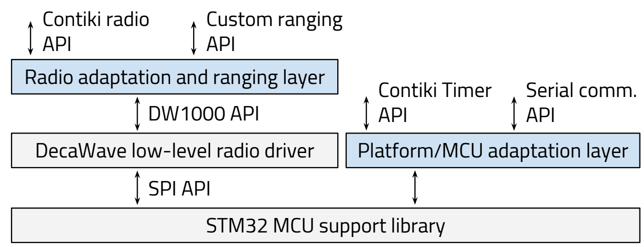


Figure 1. Software architecture.

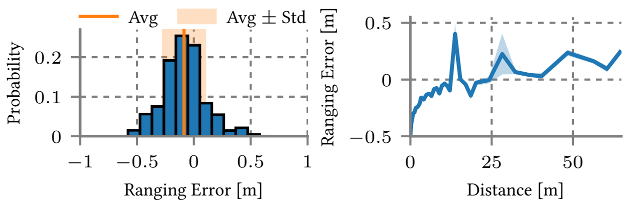
2 Ultra-wideband Communications

UWB radios based on impulse radio transmit a time-hopping sequence of very narrow pulses (< 2 ns) that translate into a large bandwidth (> 500 MHz) in the frequency domain [5]. The resulting large bandwidth leads to an excellent time resolution, enabling UWB radios to precisely estimate the time-of-arrival (ToA) of a signal and distinguish the signal's leading path from multipath components. By precisely timestamping the transmission and reception of a packet, UWB radios can estimate the distance between two devices based on a ranging exchange.

The IEEE 802.15.4-2011 [6] includes an UWB PHY layer based on impulse radio. To transmit data, each data symbol contains two bits: one obtained via pulse position modulation (PPM) and another by changing the phase of the transmitted pulses (BPSK). The DecaWave DW1000 is a standard-compliant UWB transceiver that provides centimer-level ranging at a low-power consumption.

3 EVB1000 Contiki Port

Porting Contiki to a new platform means implementing hardware abstraction modules for the MCU and various peripherals. Our port introduces such modules for the essential hardware of the EVB1000 platform, including timers, the SPI bus, serial-over-USB, and the radio. Figure 1 depicts



(a) Normalized histogram. (b) Error vs. distance.

Figure 2. Ranging error over 25k samples and 36 distances.

a high-level view on the software architecture, with the blue boxes representing the components we implemented.

Hardware Platform. The DecaWave EVB1000 evaluation board [2] includes an STM32F105 ARM Cortex M3 microcontroller, the DW1000 transceiver, an LCD display, and an external PCB antenna provided with the evaluation kit.

Platform Driver. Our port relies on low-level libraries provided by the manufacturers of the MCU and of the radio chip. The platform adaptation component is a thin layer translating the essential timekeeping, interrupt handling, low-power management, and other low-level APIs of Contiki into the calls to the MCU support library. It is also responsible for the configuration of the MCU and integrated peripherals to be used by Contiki.

DW1000 Radio Driver. The DecaWave low-level radio driver provides all necessary functions to operate the DW1000 radio chip. However, Contiki relies on a radio abstraction with its own API and a set of functional requirements that all radio drivers should implement. Therefore, we created an adaptation layer translating the Contiki API calls into the DW1000 API calls. As the standard radio interface of Contiki does not support ranging, we augmented it with a separate, custom-developed ranging API, and implemented the popular single-sided (SS-TWR) and double-sided two-way ranging (DS-TWR) [6] mechanisms at the driver level to eliminate all possible delays related to cross-layer communication in the system.

Porting the DW1000 to other MCUs. We claim that the support we introduce for the DW1000 radio chip can be easily ported to any other microcontroller. Indeed, the low-level radio driver is essentially cross-platform, relying only on a few architecture-dependent functions to communicate with the radio over SPI and to enable/disable radio interrupts.

4 Evaluation

To evaluate the port we tested in isolation its two main features: ranging and communication. However, we also recorded good system performance even in applications combining both [4].

Ranging. To assess the ranging accuracy, we ran a series of indoor tests in a corridor of our university building with two EVB1000 nodes placed in line-of-sight, one initiating ranging sessions with a 100 ms period and the other one responding. We chose frequency channel 4, the data rate of 6.8 Mbps, and the default TX power; for the ranging we used the DS-TWR scheme, requiring a 3-message handshake plus an additional message to send the measured distance back to the initiator. We changed the position of the responder from the initiator, resulting in 36 different distances (0–64m) between them. In every position, we recorded around 700 ranging samples. Note that with the distance increase, we increased the step between consecutive positions.

Figure 2a shows the normalized histogram of the ranging error over all 36 distances and more than 25k samples. The error stays within 50 cm for 99% of the samples while the median error is 13 cm. Figure 2b reports the average ranging error and standard deviation for every ground-truth distance used in the test. The average error remains below 50 cm and is typically within 20 cm. The peaks are probably due to irregularities of the corridor (furniture and large copiers).

Communication. To address the feasibility of running unmodified networking protocols atop UWB radios, we ran a series of 1 hour tests in a 14-node testbed spanning a floor in our building. We used the default Contiki Collect on top of the stock CSMA layer of Contiki, with a maximum number of 15 retransmissions and a 10 min warm-up to stabilize the routing topology. The network topology has a radius of 2–3 hops and an avg. neighborhood size of 3.83. We changed the inter-message interval (IMI) from 2 to 15 s to assess the performance with various traffic intensities. Figure 3 presents the resulting multi-hop packet delivery ratio (PDR) with two different data rates of those supported by the DW1000.

For IMI $\geq$ 10 s, Contiki Collect achieved PDR = 100%. With IMI = 5 s, we registered packet loss of 0.7–1.8% as the denser traffic started to create channel congestion. This effect was amplified with IMI > 2 s causing the network to drop 10–30% of the packets. We observe that the slower 110 kbps data rate yields better performance as it enjoys increased communication range, resulting in a 2-hop network instead of the 3 hops required for the faster 6.8 Mbps setting. These results confirm the feasibility of running unmodified Contiki Collect directly atop UWB radios with similar performance to that of conventional narrowband radios.

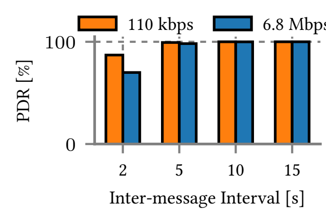


Figure 3. Collect PDR.

5 Conclusions

We presented a Contiki port for the DecaWave EVB1000 platform. The experimental results with our port confirm decimeter-level ranging accuracy and the feasibility of running unmodified Contiki Collect reliably atop UWB radios.

6 References

[1] A. Dunkels et al. Contiki - a Lightweight and Flexible Operating System for Tiny Networked Sensors. In *Proc. of Emnets*, 2004.

[2] DecaWave. DecaWave ScenSor EVB1000 Evaluation Board, 2013.

[3] M. Charlier et al. Support for IEEE 802.15. 4 ultra wideband communications in the Contiki operating system. In *Proc. of SCVT*, 2016.

[4] N. Janicijevic, P. Corbalán, T. Istomin, G. P. Picco, and E. Varriale. Demo: Small PLaNS Towards Mars: Exploiting Ultra-wideband for Self-localizing Rover Navigation. In *Proc. of EWSN*, 2018.

[5] M. Z. Win and R. A. Scholtz. Impulse radio: How it works. *IEEE Communications letters*, 2(2):36–38, 1998.

[6] IEEE Std 802.15.4-2011. *IEEE Standard for Local and metropolitan area networks–Part 15.4: Low-Rate Wireless Personal Area Networks (LR-WPANs)*, 2011.