A Practical Low-Cost Ultra-Wideband Research Platform

Bernd Baumann, BSc.

***Abstract*—A large variety of wireless communication technologies have been developed to address different kinds of challenges such as energy-efficiency, cost-efficiency and high data throughput. One of these technologies is ultra-wideband (UWB). The commercialization of the low-cost IEEE 802.15.4-compliant DecaWave UWB transceiver DW1000 led to the development of several UWB-based systems focusing on indoor localization. However, these UWB systems are not suitable for researchers who need hardware- and software access to perform experiments and measurements. Therefore, we introduce an UWB platform capable of exploiting all aspects of the DW1000 transceiver. The modular hardware design and the support by open-source software allows individual measurements and even replacement of single components of the platform. Various UWB-based Internet of Things applications can be studied and new protocols and solutions can be developed with this platform.**

***Index Terms*—UWB, Decawave, 802.15.4, IoT, STM32, Nucleo64, Contiki, RTLS, Positioning**

I. Introduction

In Internet of Things (IoT) applications like home automation or smart logistics it is often necessary for devices to be aware of their location, especially if the devices are in motion. Location-aware wireless sensor networks have been successfully deployed in various environments [1]–[3]. These examples have in common that the positions of all nodes within the network did not change during their lifetime, so there was no need to perform localization. If the sensor nodes are in motion, they have to localize themselves continuously in real-time.

The physical layer of a wireless sensor network plays an important role in terms of network performance. The most used wireless technologies (WiFi, Bluetooth/BTLE and Zigbee) share the same 2.4 GHz ISM band and are inherently narrow-band. Therefore, they are highly susceptible to multipath fading and cross-technology interference. One approach to overcome these limitations is to use UWB transceivers such as the DW1000 [4].

The DW1000 supports a mechanism to perform time-of-flight based ranging measurements. Based on this feature, positioning algorithms can be implemented. Hence, the choice to go for a DW1000-based platform solves both, the positioning problem and communication in multipath-rich environments. To run experiments and collect measurement data, we need a suitable platform that fulfills the following requirements.

A. Requirements

IoT OS support. The platform should support a common IoT OS. The open source operating system Contiki is preferred [5], as it allows us to re-use implemented communication protocols such as IPv6, RPL and CoAP.

Low-energy. The platform should be energy efficient enough to be powered by a small battery for a reasonable long time. This is a common requirement of devices employed to build IoT applications.

Low-cost. Instead of a few powerful and expensive computing devices, IoT applications require a high number of cheap devices. This requires a single device to be low-cost.

Open source. The hardware design and the firmware has to be shared with the research community. Researchers should profit from this work and accelerate their own development of UWB-based applications. Free and full access to all design sources also simplifies replication of our measurements and test results.

Individual power measurements. Since energy efficiency is a requirement, it is also of interest to measure the energy consumption of individual platform components. The requirement is to analyze each individual platform component separately.

Multi-purpose solution. Existing platforms use specialized designs (e.g., tag or anchor) depending on the role within the network [6]–[9]. Some are fixed anchors with a mains-powered power supply. Others are designed to be battery powered tags that can move around. This design is inflexible in dynamic and mobile networks and fault-prone. Furthermore, it prevents a dynamic role assignment of individual nodes too. Therefore, the platform should be a multi-purpose solution.

Exchangeable antenna. The antenna of the transceiver must be exchangeable. Supporting one single antenna is a considerable constraint for wireless sensors. By providing an SMA connector it is possible to select the most suitable antenna type for each application individually. Some require directional antennas, while other applications rely on an omni-directional antenna [10]. In a real-time location system (RTLS) two types of nodes exist: anchors (with a static, known position) and (tags with a mobile, unknown position). While an omni-directional antenna is a good choice for mobile nodes, anchors are often fixed on walls or in corners. An anchor's performance can be improved by using a directional antenna.

Expandability. It should be possible to extend the platform with additional sensors. In positioning applications, an inertial measurement unit (IMU) and a barometer are beneficial to achieve higher accuracy. The design should allow adding (and removing) hardware components on demand.

Interface for data logging. IoT links are known to be

IoT	Research	Positioning
IoT OS support	Open source	IMU (9 DoF), barometer, ...
Low-energy	Individual power measurements	Standalone logging option
Low-cost	Multi-purpose solution	Multi-purpose solution
	SMA antenna connector	SMA antenna connector

TABLE I: Requirements for the UWB evaluation platform.

lossy. Therefore, a standalone logging option is required to overcome loss of sampled sensor data. Furthermore, to re-run and analyze experiments offline, it is required to store measurement data. This logging option can be an IEEE 802.11 transceiver as a backbone communication channel or a local flash memory.

These requirements can be grouped into three fields of interest: IoT, research and positioning (columns of Table I). For IoT applications the requirements IoT OS support, low-energy and low-cost must be met. For research purposes the possibility for individual power measurements, open source designs and exchangeable antennas are required. An exchangeable antenna, expandability and a data logging interface are in the interest of positioning systems. To support further research in the field of location-aware IoT applications, we need a platform that fulfills all the listed requirements.

B. Analysis of Existing DW1000-based Platforms

There exist numerous platforms integrating the Decawave UWB transceiver in their design. The applications of these platforms are industrial [6], [8], scientific [11], for hobbyists [9] and embedded smartphone systems [12, SpoonPhone]. We evaluated the eligibility of all platforms in terms of our requirements summarized in Table I.

All analyzed platforms focus on localization applications. The UWB transceiver was hardly used as a communication interface to exchange data, except by Sewio [8]. The platforms in [7], [8], [13], [14] use the DWM1000 module[1]. The platforms in [6], [9], [11], [15], [16] use their own RF design. A complete list of all investigated platforms can be found in Appendix D. The following three platforms seemed most suitable for research and for experimenting. Table II shows which requirements were met by these platforms.

	EVK1000	OpenRTLS	Ciholas
IoT OS support	–	–	–
Low-energy	–	✓	–
Low-cost	–	–	–
Open-source	–	–	✓
Individual power measurements	✓	–	–
Multi-purpose solution	✓	–	✓
Exchangeable antenna	✓	–	✓
Expandability/IMU	–	✓	✓
Interface for data logging	–	✓	–

TABLE II: Requirements coverage of the platforms EVK1000, OpenRTLS and Ciholas.

[1]The DWM1000 is a module that embeds the DW1000 and includes a small, omni-directional chip-antenna.

EVK1000 [15]. This is Decawave's own reference design for the DW1000. It allows to measure the current of the DW1000 individually. The platform is a multi-purpose solution, requiring no specialized second board design for positioning. It has an SMA connector allowing to exchange the UWB antenna. Its STM32F105 micro-controller is not supported by an IoT OS and the current drain of the micro-controller (47mA@72MHz in active mode and 1.2mA@125kHz in idle mode) is relatively high for an IoT device. The cost of 473.11EUR[2] for 2 boards makes it too expensive for large-scale networks. It has no interface for data logging and it does not provide connectors to add sensors like an IMU to the existing design.

OpenRTLS [6]. This platform focuses on real-time positioning in industrial and medical environments. The tags run on batteries and their micro-controller has a low power consumption of $24mA$ in active mode and $2\mu A$ in idle mode. They include an IMU and a barometric sensor in their design as well, but no data logging interface. UWB antennas can be exchanged on anchors, but not on tags. Anchor nodes require a mains-powered power supply and embed a second wireless interface (2.4 GHz, 802.15.4 compatible) as well as an Ethernet connector. Both designs are not supported in an IoT OS. While the tag (uNemo) costs only 43EUR, the anchors are more expensive (375EUR). This platform is unsuitable for research, because neither its hardware design nor its firmware is open source. Power measurements of single components are not supported and multiple hardware designs are used instead of a multi-purpose platform solution.

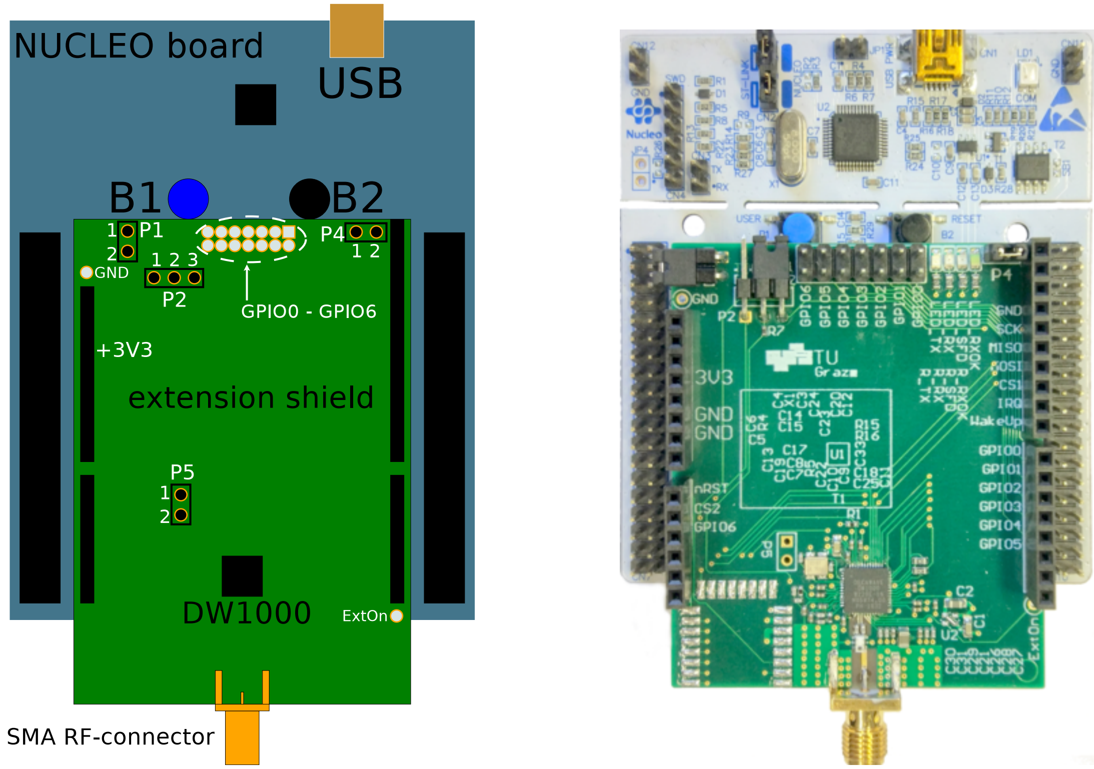
Ciholas [16]. This platform provides all hardware design files of its DWUSB board, but no firmware sources. The board has an SMA connector for the UWB antenna, an integrated IMU and a pressure sensor. Although the application focuses on RTLS, there is no separate design for anchors. The DWUSB is a multi-purpose solution. The costs per board of 180EUR are too high to build large sensor networks. Its 120MHz ARM Cortex-M4 MCU has an idle mode current drain of $6.9mA$ and $38.0mA$ in active mode. For IoT devices this energy consumption is not efficient enough. Furthermore, there is no IoT OS support for this platform and the latter has no interfaces for data logging. Individual power measurements of this board's components are not supported by its design.

Since none of the analyzed platforms fulfilled all our requirements, we decided to design a new one. The resulting modular platform design allows faster future research on ultra-wideband. The platform design is described in Section II. After discussing the limitations in Section III, we show the results of the UWB board's calibration in Section IV.

II. Design

We present the design of a new DW1000-based ultra-wideband platform. Its hardware design allows the DW1000 to be used not only for localization but also as a communication interface.

[2]Distributor www.digikey.com



(a) Block diagram of the UWB extension shield stacked on top of the NUCLEO-L152RE board.

(b) Picture of the UWB extension shield stacked on top of the NUCLEO-L152RE board.

Fig. 1: The UWB extension shield stacked on top of the NUCLEO-L152RE board.

Section II-A lists all requirements from Section I-A and describes how they are met in our design. The UWB platform consists of the following main components:

- NUCLEO-L152RE development board
  This board from ST Microelectronics combines a ST-Link programmer with an ARM Cortex-M3 STM32L152RE MCU development board. The NUCLEO-L152RE board embeds the MCU, programming and debugging interfaces, a USB connector, the voltage conversion and power supply circuits. For user inputs, the board provides a push-button (B1 in Figure 1a). Its state is readable through a GPIO pin on the MCU. To reset the MCU, there is a second push-button (B2 in Figure 1a).
- UWB extension shield
  This is our own design of an extension board compatible with the Arduino header format. It provides access to the DW1000 transceiver and includes an RF design to a 50Ω SMA connector.
- X-NUCLEO-IDW04A1 [17] *(optional)*
  It extends the platform with the 802.11 b/g/n transceiver SPWF04SA and an micro-SD card slot.
- X-NUCLEO-IKS01A1 [18] *(optional)*
  This motion MEMS extension shield includes a 3D accelerometer, a 3D gyroscope, a 3D magnetometer, a pressure sensor (barometer), a humidity sensor and a temperature sensor.

Figure 1 shows a block diagram (Figure 1a) and a picture (Figure 1b) of the UWB platform without the optional boards. The NUCLEO-L152RE board can be extended through connectors that are compatible to the Arduino format [19], as well as through the less flexible but more powerful MORPHO connectors. Additional hardware for programming the MCU is not necessary since an ST-Link programmer is already included. Another advantage is the flexibility to switch to different, compatible boards of the same product family (NUCLEO64), if necessary. If the application demands more computational power, more memory, or less energy consumption, the board can be exchanged with one that better fits the application requirements. Other extension shields can be stacked on top of the UWB board to complement the existing platform. This choice of hardware components sufficiently fulfilled our requirements from Table I. The following detailed description explains our design decisions in the context of our requirements.

A. *Fulfillment of the Requirements*

IoT OS Support. The Contiki operating system supports a variety of different processors and IoT platforms. The choice to use the STM32-based Nucleo board NUCLEO-L152RE brings multiple benefits, one of them being the support for both the STM32L152RE ARM-based Cortex-M3 CPU and the NUCLEO-L152RE platform. The driver for the DW1000 transceiver is not included in the official Contiki repository. However, an implementation exists [20].

Low-energy. Of all hardware components in wireless sensor nodes the radio transceiver has the highest energy consumption. The chosen UWB transceiver DW1000 is no exception: in some operational modes, the average current consumption exceeds $150mA$. For small battery powered devices this is a problem. To reduce the DW1000's overall power consumption, an external DC-DC converter is used as proposed in [4, section 7.2]. A crystal oscillator was used instead of a temperature-compensated crystal oscillator (TCXO) because of its higher energy efficiency.

The Cortex-M3 CPU of the NUCLEO-L152RE is low-power and supports sleep as well as deep-sleep states [21]. The CPU has a current draw of $7.55mA@32MHz$ in active mode and $21.5\mu A$ in low-power sleep mode, which makes it suitable for battery-powered IoT devices.

The DW1000 supports driving LEDs on four GPIO pins to indicate certain states of operation (transmitting, receiving, preamble received and start-of-frame delimiter received). This is useful for debugging, but it consumes more energy. Therefore the LEDs are included in the design but can be switched off by removing jumper P4. If there is no need for the LEDs on the platform, they can be left out on the board without affecting the functionality of the latter.

The choice of using an external UWB antenna has an impact on energy efficiency as well as, because a higher antenna gain can increase the quality of a wireless communication link with the same energy budget.

Low-cost. The NUCLEO-L152RE platform is a low-cost development platform that requires no additional programmer. The ST-Link software used to flash the micro-controller is open source and can be downloaded from [22]. The firmware and the Contiki OS are open source too. Components like IMUs, IEEE 802.11 transceiver or other sensors can be added to the platform on demand and can be left out to reduce the overall cost of the platform. The PCB manufacturing cost of 17.52EUR[3] accounts for the largest portion of the overall platform cost. The components per board sum up to 29.90EUR[4] (see [23] and Appendix C for details). A platform without the optional WiFi and MEMS boards would cost 47.42EUR. Including the WiFi board (18.79EUR) and the MEMS board (12.51EUR) the platform costs 78.72EUR.

Open source. The software ST-Link is released under the BSD license. The Contiki OS source code is released under the Contiki open source license. The hardware design of the UWB extension shield was created using the Altium Designer software [24]. All design files and Gerber outputs are open-source as well. The UWB extension shield is designed to fit on the Arduino header, a common development platform that is also open source hardware.

Individual power measurements. The UWB extension shield was designed to allow power measurements. To measure the current consumption of the UWB extension shield, one can connect a current probe to jumper P1 (see Table III and Figure 1a). In default mode, the jumper P1 must be shortened to connect the +3.3V supply pin from the Arduino header and the VCC plane from the UWB shield.

To measure the supply voltage of the extension shield, the voltage probe must be connected to the +3.3V supply pin and to the GND pin. To connect different types of probes, the GND test point next to the P1 jumper can be used.

Multi-purpose solution. The extension shield can be used in combination with different other MCU-boards. Nevertheless, it is possible to build sensor networks with wireless sensor nodes that all consist of the same hardware components. One is not forced to use specialized designs for different roles (like anchors or tags) in the network. If the application demands different hardware components in different nodes, the same UWB extension shield can be used in every sensor node. The SMA antenna connector simplifies the multi-purpose solution.

Exchangeable antenna The SMA-RF connector allows experimenting with all antennas that have a female SMA connector. The user can choose an antenna design depending on the application. By using an RF switch one can use multiple antennas in the same design [10].

Expandability. The decision to use the DW1000 on an extension shield was made to expand the hardware later on. The platform's micro-controller can be changed without altering the UWB extension shield by stacking the extension shield on different MCU boards. The chosen MCU board family (NUCLEO) offers two separate header formats for stacking - ST's own MORPHO header format and the widely used Arduino compatible header format. The advantage of the Arduino header connectors is that it is supported by a large community. The UWB extension shield is designed to allow stacking of other extension shields. Stacking two UWB extension shields is considered in the design and described in Section II-B. The P2 header (see Table III and Figure 1a) was added to support two different SPI chip-select lines (SPI-CS). This way, the same SPI bus can be used by another shield that is stacked on top of the UWB extension shield. The header pins of the serial UART and I2C buses were left unused in the UWB extension shield, so that other extension shields can make use of them. In particular the WiFi shield [17] and the sensor shield [18] by ST are supported in combination with the UWB extension shield.

Interface for data logging. An interface for data logging can be stacked on top of the UWB extension shield. This can be a reliable second wireless interface like IEEE 802.11, or an offline mass storage to collect all test data for offline analysis. The WiFi shield X-NUCLEO-IDW04A1 [17] supports both by providing access to an on-board micro SD card.

B. Pin Usage and Jumpers

Table IV lists the names of the Arduino header connector pins. The communication buses SPI, I2C and UART are bold. The UART and I2C communication ports were left unused on the UWB extension shield so that the WiFi and MEMS boards can be used simultaneously. Three boards are listed in the table:

[3] We paid 175.20EUR for 10 pieces. The price heavily depends on the ordered quantity. The cost of ordering a single PCB is 126.42EUR.

[4] Vendor www.mouser.com.

Jumper	Pin	Net Name	Description
P1	P1-1	+3V3 plane	positive voltage supply of UWB shield
	P1-2	+3V3 pin	positive voltage output of NUCLEO-L152RE header
P2	P2-1	SPI-CS2	alternative (2nd) SPI chip select
	P2-2	SPI-CS	SPI chip select of DW1000
	P2-3	SPI-CS1	default SPI chip select
P4	P4-1	LEDs cathodes	connected to the cathodes of all 4 LEDs
	P4-2	GND	connected to the ground plane of the UWB shield
P5	P5-1	+3V3 plane	connected to the positive voltage supply plane
	P5-2	DWM1000 VSS	connected to the supply pins of the optional DWM1000
GPIO0-GPIO6	GPIO0-1	DW1000 pin 38	connected to GPIO0 of DW1000 (LED_RXOK)
	GPIO0-2	Pin 8 of CN9	connected to pin number 8 of header CN9
	GPIO1-1	DW1000 pin 37	connected to GPIO1 of DW1000 (SFDLED)
	GPIO1-2	Pin 7 of CN9	connected to pin number 7 of header CN9
	GPIO2-1	DW1000 pin 36	connected to GPIO2 of DW1000 (RXLED)
	GPIO2-2	Pin 6 of CN9	connected to pin number 6 of header CN9
	GPIO3-1	DW1000 pin 35	connected to GPIO3 of DW1000 (TXLED)
	GPIO3-2	Pin 5 of CN9	connected to pin number 5 of header CN9
	GPIO4-1	DW1000 pin 34	connected to GPIO4 of DW1000 (EXTPA)
	GPIO4-2	Pin 4 of CN9	connected to pin number 4 of header CN9
	GPIO5-1	DW1000 pin 33	connected to GPIO5 of DW1000 (SPIPHA)
	GPIO5-2	Pin 3 of CN9	connected to pin number 3 of header CN9
	GPIO6-1	DW1000 pin 30	connected to GPIO6 of DW1000 (SPIPOL)
	GPIO6-2	Pin 4 of CN9	connected to pin number 3 of header CN8

TABLE III: List of Jumpers and their purpose.

1) NUCLEO-L152RE/STM32L152RE
   The labels of this board's connectors are CN6, CN8, CN5 and CN9. In Figure 2a, CN6 is connected to the UWB shield with the upper left, CN8 with the lower left connector, CN5 with the upper right and CN9 with the lower right. The pin names of the STM32L152RE are taken from the datasheet [21].
2) X-NUCLEO-IDW04A1
   The WiFi shield has multiple GPIO pins, but GPIO9, GPIO13, GPIO14 and GPIO15 cannot be used because they are reserved for the UWB shield's SPI pins. The X-NUCLEO-IDW04A1 is controlled using a UART bus interface.
3) UWB Shield
   The DW1000 can only be controlled using the SPI bus. Therefore, using the SPI pins cannot be avoided on the UWB shield. The GPIOx pins of the UWB shield are optional. They are only connected to the CN9 header pins if the jumpers GPIO0 - GPIO6 are closed (see Table III)

The X-NUCLEO-IKS01A1 pins are not listed in Table IV. It is controlled using the I2C bus of CN5 and powered through CN6. Table III lists all the jumpers of the UWB shield.

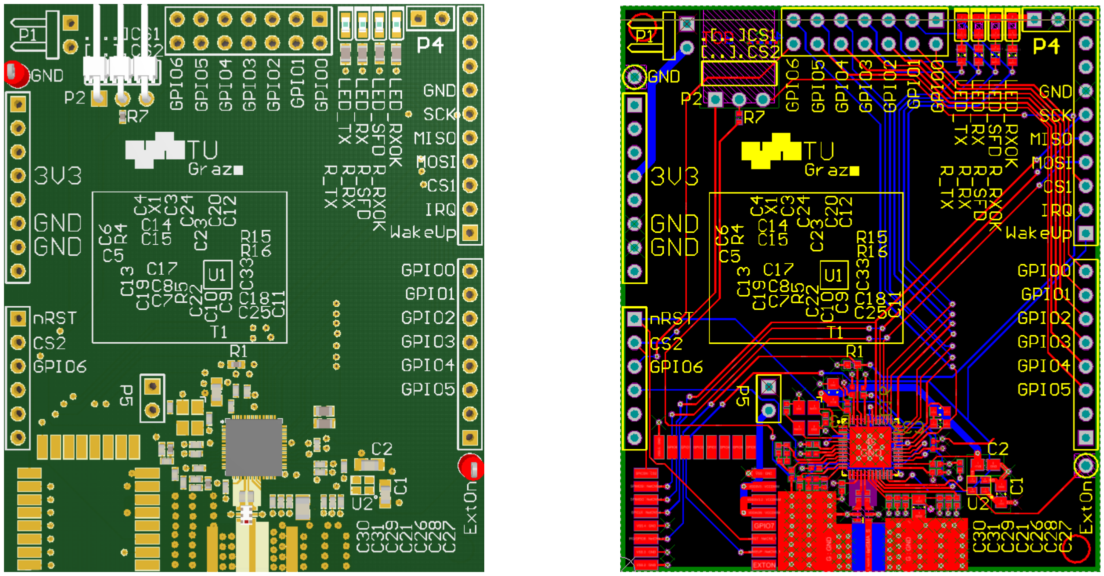
P1. This jumper was added to the design to simplify current measurements. The supply current of the UWB shield can be measured by connecting an ampere-meter to the pins P1-1 and P1-2. P1 connects the voltage supply pin CN6-4 (see Table IV) to the power plane (see Table V) and must be closed for normal operation.

P2. The center pin P2-2 is connected to the DW1000's SPI chip select pin. By closing P2-2 and P2-3 (on the right), the CN5-3 pin (see Table IV) is used as the SPI chip select. This is the default use-case. If P2-1 and P2-2 is closed instead, a second UWB shield (or any other shield that uses pin CN5-3 as SPI chip select) can be stacked onto the same NUCLEO-L152RE board. Then the UWB shield can still communicate by using CN8-2 as SPI chip select.

Pin	STM32L152RE	WiFi Shield	UWB Shield
	CN6 (Power)		
1	NC		
2	3V3		NC
3	Reset Button		NC
4	3V3		
5	5V		NC
6	GND		
7	GND		
8	VIN		NC
	CN8 (Analog)		
1	PA0/ADC_IN0	NC	nRST
2	PA1/ADC_IN1	NC	**CS2**
3	PA4/ADC_IN4	NC	*GPIO6*
4	PB0/ADC_IN8	GPIO6	SYNC/GPIO7
5	PC1/**SDA**	GPIO1	NC
6	PC0/**SCL**	NC	
	CN5 (Digital)		
10	PB8/**SCL**	GPIO5	NC
9	PB9/**SDA**	GPIO4	NC
8	AVDD		NC
7	GND		
6	PA5/**SCK**	GPIO15	**SCK**
5	PA6/**MISO**	GPIO13	**MISO**
4	PA7/**MOSI**	GPIO14	**MOSI**
3	PB6/**CS**	NC	**CS1**
2	PC7	GPIO9	**IRQ**
1	PA9	GPIO2	WAKEUP
	CN9 (Digital)		
8	PA8	WiFi RST	*GPIO0*
7	PB10	PB10	*GPIO1*
6	PB4	PB4	*GPIO2*
5	PB5	PB5	*GPIO3*
4	PB3	NC	*GPIO4*
3	PA10	GPIO11	*GPIO5*
2	PA2/**UART_TX**	GPIO11/**UART_RX**	NC
1	PA3/**UART_RX**	**UART_TX**	NC

TABLE IV: Pin usage of header connectors.

P4. There are 4 LEDs that can be soldered on the UWB shield to indicate the communication state. If GPIO0 - GPIO3 have



(a) Top view of UWB extension shield.

(b) PCB Designer view of UWB extension shield.

Fig. 2: UWB extension shield design.

to be used, the LEDs must be disconnected by opening P4. This also helps saving energy.

P5. In case the optional DWM1000 module is soldered on the UWB shield, P5 must be closed. P5 connects the DWM1000's voltage supply to the power plane. Both the DW1000 and the DWM1000 can be used at the same time. For simultaneous use, the DW1000 must be connected to CS1 by closing P2-2 and P2-3. The DWM1000's SPI-CS is connected to CS2 (CN8-2). If there is no DWM1000 soldered on the UWB shield, the state of P5 is irrelevant.

GPIO0 - GPIO6. There are 8 GPIOs on the DW1000 that can be controlled by the firmware. GPIO7 is connected to CN8-4. GPIO0 to GPIO6 are accessible to the user in two ways. One way is to attach measurement probes to the test points on the edge of the board. The other way would be to solder a pin header onto these test points or shorten them using solder bridges (in Figure 1b the pin header is soldered onto the board). Then the GPIOs are accessible to the NUCLEO-L152RE board via its CN9 header pins as well. Closing these jumpers is optional. By closing the GPIOx jumpers, the DW1000's GPIO0 - GPIO5 pins are connected to the CN9-8 - CN9-3 pins and GPIO6 is connected to CN8-3 (see Table IV).

C. Production Specifics of the UWB Extension Shield

Figure 2 shows our design of the UWB extension shield with Figure 2a showing the top view. In Figure 2b the copper elements of the top layer are colored in red and the copper elements of the bottom layer are colored in blue.

Decawave provides its own design guide for integrating the DW1000 in a hardware design [25]. This hardware design guide and Decawave's own reference design for the DW1000 (the EVK1000 evaluation platform [15]) were kept in mind for the UWB extension shield design.

As discussed earlier, the manufacturing of the PCB is the main cost factor. The following parameters affect the PCB cost and should hence be considered:

- Size (area);
- Number of Layers;
- Quantity;
- Surface finish;
- Track width;
- Via diameter;
- Material of dielectric;
- Color of solder stop;
- (optional) Solder stencil[5].

Although the cost should be minimized, we decided to make the following trade-offs to ensure a high quality design.

Track width. A smaller track width leads to higher production costs. The minimum track width of the PCB design is limited by the pad width of the DW1000, which is $200\mu m$.

4-Layer layout. The choice of a 4-layer layout instead of a cheaper 2-layer layout had two reasons: (i) to decrease the width of the RF transmission lines (see Section II-G) and (ii) to simplify the uniform power supply while placing the components close to each other. The placement of the decoupling capacitors is crucial to the performance of the DW1000. A 4-layer layout allows to place them very close to the DW1000's pins. The second and third layer in the 4-layer layout are assigned to be a VCC plane (positive voltage power supply) and a GND plane (negative voltage power supply), respectively. That way, these two nets are accessible through vias anywhere on the PCB. Section II-E gives detailed description of the layer stackup.

Board size. A larger area increases the costs of the extension

[5]When soldering in a reflow-oven, a stencil is helpful to deposit the soldering paste onto the SMT pads.

shield. The minimum width of the PCB results from using the Arduino header connector, which is $50.26mm$. To have a safe distance between the SMA connector, the RF transmission lines and the underlying NUCLEO-L152RE board, the length of the board is $60mm$. That is $10.74mm$ longer than the minimum length forced by the Arduino header.

Electroless Nickel Immersion Gold (ENIG). This type of surface plating is more expensive compared to other techniques. One of its advantages is the constant surface planarity. The surface thickness must be considered when calculating the dimensions of the RF transmission lines. A deviation of the surface thickness would lead to a incorrect width of the transmission lines.

Manufacturer. We chose the manufacturer Multi-CB [26] to produce our PCBs. Though the layer stackup recommended by Decawave [25, Figure 5] was not offered, an individual layer stackup was possible. Detailed information about thickness and material of each layer was available and was considered in the calculation of the transmission line width. The layer stackup is described in detail in Section II-E.

Components placement. Possible positions for the antenna port were the top or the bottom edge of the PCB. The bottom edge was chosen to increase the UWB antenna's distance to all other electronic components. Placing the DW1000 in the middle of the board would allow good access to all 48 of its pins. To keep the length of the transmission lines as short as possible (and therefore the chance of impedance deviations), the DW1000 was instead, placed closer to the bottom edge (see Section II-G for details). One design aspect that was not obvious before we soldered the UWB shields was the correct alignment of the stencil. The pads of R7 (close to P2) were very helpful to align the stencil on the PCB.

D. Custom SMA Footprint Library

The SMA connector must not be through-hole mounted, but should be edge-mounted. A through-hole mounted connector would add an unwanted stub to the transmission line. In our design the footprint of the SMA connector is customized to avoid impedance discontinuities. The pad width of the center pin that carries the signal is matched to have the same width as the transmission line. Furthermore, the pad length of the signal pin was increased until the edge of the PCB so that there is no gap between the transmission line and the PCB edge.

E. PCB Layer Stackup

Table V shows the layer names, the chosen FR4 types and their thickness. In total, the PCB has a thickness of $1\,526\mu m$. A symmetric layer stackup was demanded by the manufacturer. Especially the dielectric between the component side, where the transmission lines are located and the underlying GND plane is of interest. Using two layers of Prepreg 7628 with a combined thickness of $360\mu m$ came closest to the recommendation in [25]. The manufacturer Multi-CB also provided information about the Prepreg that will be used for production. In our case, this was the DE104 by Isola Group [27]. According to the DE104 datasheet, the **permittivity** at $5GHz$ is $\epsilon_r = 4.32$. These two values - the thickness and the permittivity - will affect the dimension of the transmission lines (see Section II-G).

F. Placing Decoupling Capacitors

The decoupling networks must fulfill two purposes. First, they must filter noise and other high frequency signals. Second, they act as energy buffers. To effectively filter high frequency signals, the capacitance should be small and the component must be placed close to the signal source.

The current path must be considered when placing the capacitors. A decoupling is only possible if the pads of the capacitor are positioned between the power source and the DW1000 pad. Figure 3 shows capacitor C22 to illustrate the correct placement of decoupling capacitors.

The power amplifier pins VDDPA1 and VDDPA2 of the DW1000 are decoupled by a network of 7 capacitors to avoid noise propagation to other parts of the board. See [25, chapter 6.3] for a detailed explanation on how to place them.

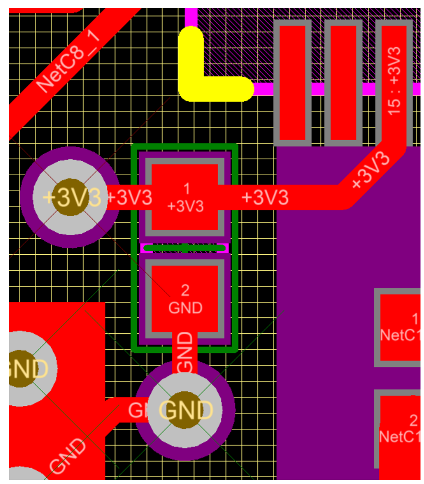


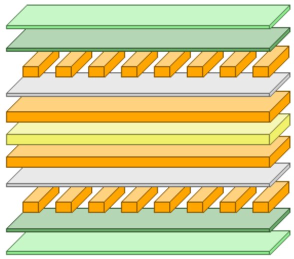
Fig. 3: Correct decoupling capacitor placement.

G. Transmission Line Design

Designing the transmission lines can be challenging, because even small miscalculations could cause problems. We included a backup solution in our PCB design in case of unforeseen problems, which is described in Section II-H.

Figure 4 shows the transmission lines on the UWB shield. The differential transmission lines connecting the DW1000's RF_P and RF_N pins to the capacitors C9 and C10 must have 100Ω. The second differential pair connects the capacitors to the UWB Balun and must have 50Ω. Finally a 50Ω single-ended transmission line connects the Balun to the SMA signal pin. There is a rule of thumb in RF design saying that a connection in a network must be treated as a transmission line if it is longer than $\lambda/10$. Assuming the highest frequency on these transmission lines is 10GHz, this would lead to a length of $3mm$ to be considered.

$$\frac{\lambda}{10} = \frac{1}{10} \cdot \frac{speed\ of\ light}{frequency} = \frac{1}{10} \cdot \frac{299\,792\,458\,\frac{m}{s}}{10 \cdot 10^9\,\frac{1}{s}} = 3mm \quad (1)$$



Name	Material	Thickness [μm]
Top Overlay	Ink	
Top Solder	Solder-Stop	10
Component Side	Copper	35
Dielectric	2 x Prepreg 7628	180+180
Ground Plane	Copper	18
Dielectric	FR-4 Core	700
Power Plane	Copper	18
Dielectric	2 x Prepreg 7628	180+180
Solder Side	Copper	35
Bottom Solder	Solder-Stop	10
Bottom Overlay	Ink	

TABLE V: Thickness and Material of the 4-Layer PCB Stackup [24].

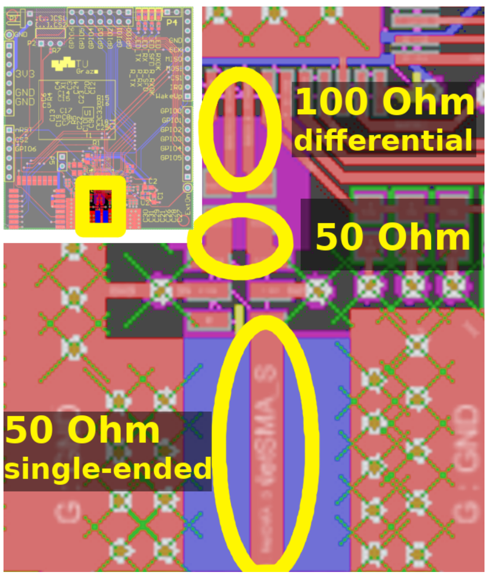


Fig. 4: RF transmission lines.

Net	Length in mm
100Ω differential	1.18
50Ω differential	0.35
50Ω single-ended	**5.4**

TABLE VI: Lengths of (potential) transmission lines.

For the same reason, the components on the UWB shield have the housing size of **0402**[6]. A bigger housing would require to treat that component like a transmission line. According to (1), the transmission lines can be treated as normal connections as long as they are shorter than 3mm. This was taken into account when designing the PCB. In the final design, the transmission lines seen in Figure 4 had the lengths listed in Table VI. Line impedance matching by calculating the correct width was neglected for connections shorter than 3mm. So only the connection from the Balun to the SMA connector was treated as a transmission line. The model to calculate the correct width of the 50Ω single-ended transmission line is illustrated in Figure 5. T is the thickness of the copper on the component side. As mentioned before in Section II-C, there are fabrication methods that guarantee an almost constant thickness. H is the thickness of the dielectric. In this case, it is a 360μm thick FR-4 (see Table V) having an $\epsilon_r = 4.32$

[6]0402 is a standard package size for SMT components. The dimensions are 1.0mm x 0.5mm.

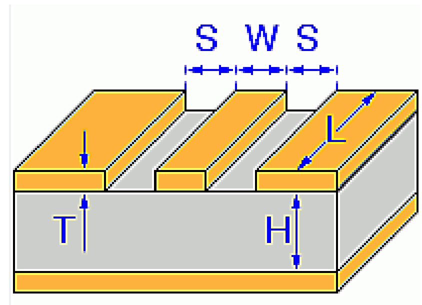


Fig. 5: Scheme for transmission lines with co-planar wave guide and ground plane.

at 5 GHz [27]. Another rule of thumb says that the space between the shielding GND planes and the transmission line should be twice the transmission line's width. There are several vias added to the GND planes on both sides of the 50Ω transmission line to improve the shielding of the transmission line.

The area around the transmission lines must be free from solder stop and printed text, because these layers are not taken into account when calculating the trace width of the transmission line.

Transmission line calculator tools. The Altium Designer software includes some helpful tools to design transmission lines [24]. By providing the layer stackup (Table V) and defining the desired line impedance, Altium automatically calculates the line width. There are several other tools available to calculate the width of a transmission line. Their results are listed in Table VII. Even though the results are different, their deviation is still within the manufacturing tolerances. We took the value calculated by Altium Designer for our design.

Tool	Result for W in μm
Altium 17 [24]	**700**
Multi-CB Online Tool [26]	636
PCB Calculator [28]	710
NI AWR TX Line [29]	672
Saturn PCB [30]	740

TABLE VII: Results for W calculated by different tools.

H. DWM1000 Footprint

The UWB extension shield design includes a backup solution by soldering a DWM1000 module [31]. The footprint was added to the PCB layout in case the RF transmission line design of the DW1000 is faulty or shows insufficient performance. That way the PCBs would still be of use. Another advantage is that the DWM1000 and the DW1000 can be used in the same design. This can be useful for comparing the performance of the DWM1000's omni-directional UWB antenna with individual designs. Listening on different RF channels at the same time is also possible.

I. Crystal

The DW1000 uses a 38.4 MHz quartz crystal to synthesize the frequencies for RF TX, RF RX and all digital blocks. Since ranging and localization applications rely on precise time measurements, the accuracy of the crystal is crucial for their performance. Ambient temperature changes have a strong impact on the clock drift of the quartz crystal. This can be omitted by using a temperature-compensated crystal instead of a simple one. In Section II-A it was mentioned that we are not using a TCXO in favor of a lower energy consumption. A low frequency tolerance and frequency drift of $\pm 10ppm$ are recommended in the DW1000 datasheet alongside with three crystals by different manufacturers [4]. Because none of these three recommended crystals were available for purchase at design time, we decided to use Epson's TSX-3225 in our design [32]. The TSX-3225 is a 10ppm crystal oscillator with the same specifications as the RSX10 used by the EVK1000 [15].

The DW1000 offers a crystal calibration mechanism to compensate for the initial frequency offset. The procedure to trim the crystal is described in detail in Appendix A-B1.

In the PCB layout, the crystal must be placed as close as possible to the XTAL1 and XTAL2 pins. To avoid too much noise interference by the DC-DC converter, the DC-DC converter was placed on the opposite side of the DW1000, as recommended in the hardware design reference guide [25].

III. Limitations

A. Optional DWM1000

If the optional DWM1000 module is soldered on the board in addition to the DW1000 transceiver, the SPI communication can be unreliable if the DWM1000 is not powered. The voltage supply of the DWM1000 module can be assured by closing jumper P5. It connects the voltage supply pins of the DWM1000 to the 3V3 net of the extension shield. While the DWM1000 is supplied, its MISO, MOSI and SCK pins have a high impedance of 50 to 90 $k\Omega$. If the module is not supplied, the pins are floating and can unintentionally pull down the voltage and, therefore, disturb the SPI communication to the DW1000. Figure 6 shows the clock signal if the DWM1000 is present and P5 is open, while Figure 7 shows the same signal with P5 closed. In both figures, measured using a digital storage oscilloscope, the upper and lower threshold for detecting if the input signal state is HIGH or LOW are sketched. In Figure 6 the SPI clock signal never exceeds the upper threshold. Because the SPI clock state as detected by the DW1000 stays at LOW, the communication to the DW1000 is impossible.

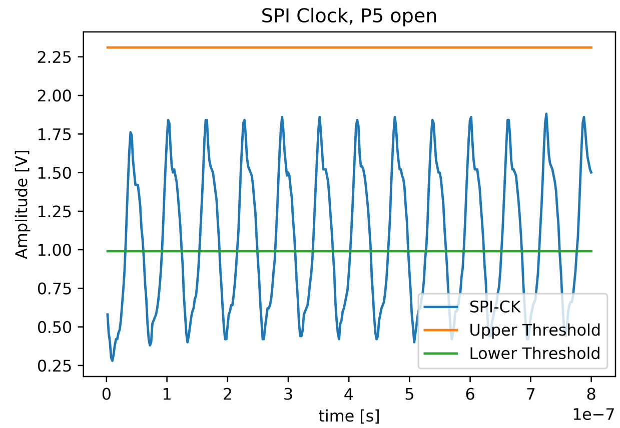


Fig. 6: Oscilloscope measurement of the distorted SPI clock signal.

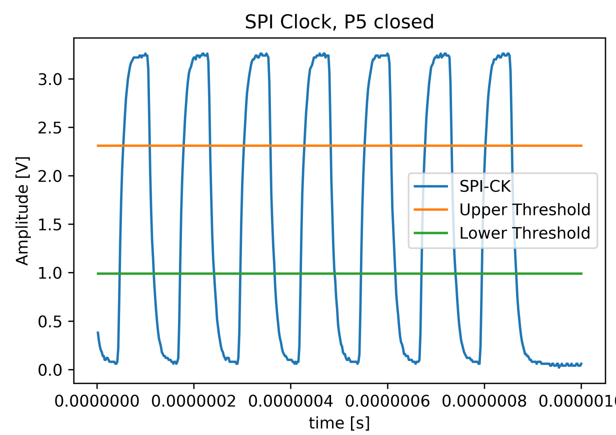


Fig. 7: Oscilloscope measurement of the valid SPI clock signal.

B. Optional DC/DC Converter

The component **U2** on the board is the DC-DC step down converter. By supplying certain pins of the DW1000 with 1.8V instead of 3.3V, the transceiver consumes less energy. However, the footprint of this component is mirrored unintentionally in the design and hence the DC-DC converter cannot be used. The pads U2-2 and U2-4 must be shortened with a piece of wire to supply all DW1000 pins as shown in Figure 8. In a future design, the footprint should be mirrored to use the DC-DC converter for saving energy.

C. Marking

On the top layer of the board, the label for the SYNC pin beneath GPIO6 is missing. The next design should include that label. All texts should be written in the same font size as well (the CN6 pin labels have a larger font size than the others).

D. TCXO Footprint

In the next design, it should be possible to choose between a TCXO and a normal quartz crystal. The TCXO usually has

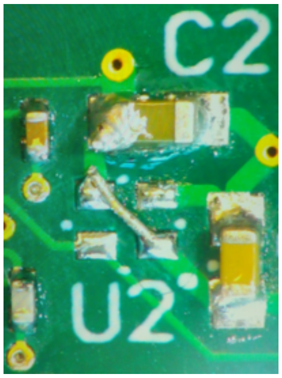


Fig. 8: Work-around for faulty DC-DC footprint.

6 connectors, because it must be connected to a low noise power source. If a TCXO is used, an additional low-dropout regulator (LDO) must be added.

IV. Evaluation and Calibration of the UWB Platform

After soldering the UWB extension shields in a re-flow oven, the production tests for DW1000 based circuits were performed [33]. A description how to execute these tests on our UWB platform can be found in Appendix A.

For voltage- and current measurements the Fluke 87-V Digital Multimeter was used. Transmit power and crystal frequency were calibrated using the Rohde-Schwarz FSQ26 Spectrum Analyzer.

A. Operational Tests

Current after 10s	**Voltage**
[mA]	[mV]
18.88	3270

TABLE VIII: Idle current measurement.

The current consumption of the board in idle mode was within the expected range (see Table VIII). The SPI test executed successfully. So did the GPIO strobe test, the RSTn test, the EXTON test and the WAKEUP test. The results of the maximum current consumption are listed in Table IX. The

Maximum current consumption	**Voltage**
[mA]	[mV]
220	3268

TABLE IX: Maximum current measurement.

operational tests showed a maximum current consumption that was much higher than expected (see Table IX). The reason for that is not clear yet. A design flaw in the UWB shield is not suspected, because this can be measured using the DWM1000 too.

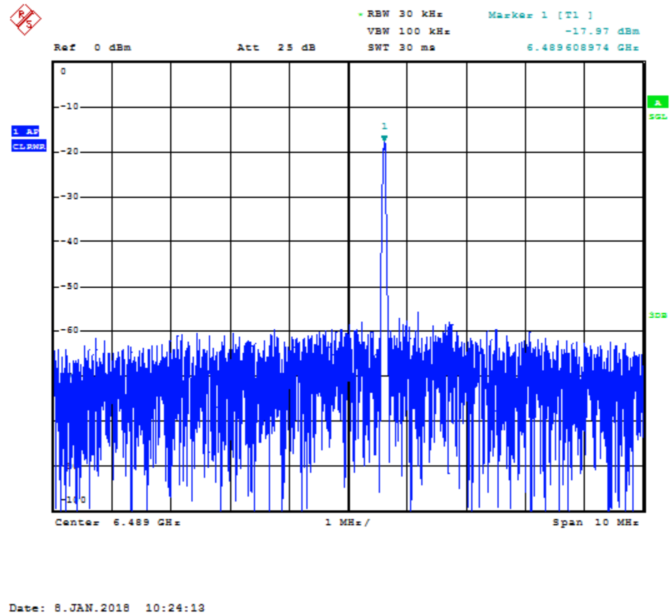


Fig. 9: Result of crystal calibration on channel 5. The marker 1 shows that the peak is at a frequency of 6.489608974 GHz.

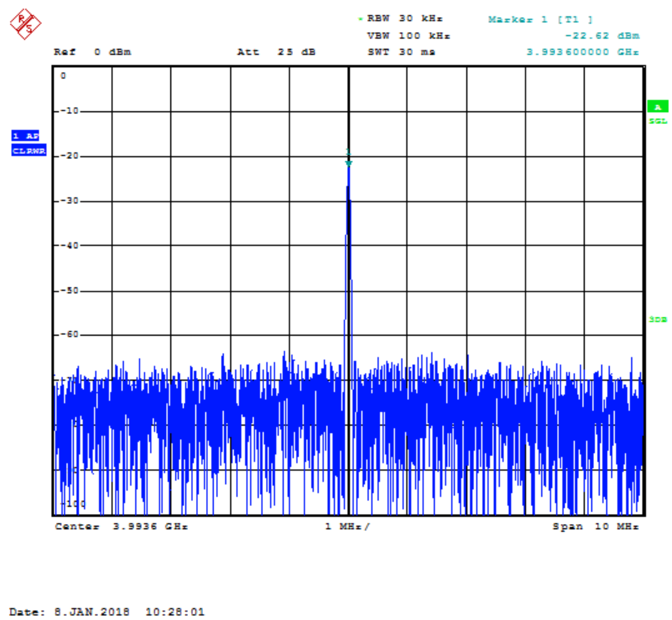


Fig. 10: Result of crystal calibration on channel 4.

B. Transmitter Calibration

Crystal calibration. The remaining frequency offset was the smallest at the crystal trim value `0x16`. On channel 5, the remaining offset was 0.008974MHz and on channel 4 the spectrum analyzer showed a remaining offset of 0.0MHz. Captures of the measurements for channel 5 are shown in Figure 9 and for channel 4 in Figure 10. In every subsequent test this crystal trim value was configured.

Transmit power calibration. The attenuation of the SMA cable was about 1dB across all tested frequencies. We con-

Test number	Channel	Frequency	Bandwidth	PRF	Code	Coarse Gain (reference)	Fine Gain (reference)	PGDELAY (reference)
		[MHz]	[MHz]	[MHz]		dB	dB	
1	1	3494.4	499.2	16	1	9 (+0)	10.5 (+0)	`0xC9` (+0)
2	1	3494.4	499.2	64	9	9 (+0)	3.5 (+0)	`0xD0` (+7)
3	2	3993.6	499.2	16	3	9 (+0)	9.5 (-1)	`0xD4` (+15)
4	2	3993.6	499.2	64	9	9 (+0)	3.5 (+0)	`0xD4` (+15)
5	3	4492.8	499.2	16	5	9 (+0)	9.0 (+1.5)	`0xD1` (+12)
6	3	4492.8	499.2	64	9	6 (+0)	6.0 (-0.5)	`0xD1` (+12)
7	4	3993.6	1331.2	16	7	**15 (+3)**	**12.0 (+3.5)**	`0x95` (+0)
8	4	3993.6	1331.2	64	9	6 (+0)	11.5 (-2.5)	`0x95` (+0)
9	5	6489.6	499.2	16	3	12 (+0)	3.0 (+1)	**`0xD2` (+18)**
10	5	6489.6	499.2	64	9	6 (+0)	1.5 (-1)	**`0xD2` (+18)**
11	7	6489.6	1081.6	16	7	6 (+0)	13.0 (+4)	`0x93` (+0)
12	7	6489.6	1081.6	64	17	0 (+0)	12.0 (+3.5)	`0x93` (+0)

TABLE X: Measurements of Transmit Power. +3 means we had to increase the reference value by 3, -1 means we had to decrease the reference value by 1. +0 indicates where the reference value was left unchanged.

figured the spectrum analyzer offset to take that attenuation into account, so we did not need to subtract it later as described in Appendix A-B2. Along with the transmit power, we adjusted the bandwidth as well. The bandwidth can be adjusted by changing the UWB pulse width. The column for the pulse generator delay (PGDELAY) lists the calibrated values for register `2A:0B`. By increasing the pulse generator delay, the pulse width (and the transmit power) increases and the bandwidth decreases. If this value is decreased, the bandwidth increases (and the transmit power decreases). The measurement results are listed in Table X. The numbers in brackets indicate the difference to the reference values from the DW1000 user manual [34]. Figure 12 and Figure 13 show the spectrum for all channels and pulse repetition frequencies (PRF). Outside the band, the transmit power has to be $10dB$ below the maximum limit of $-41.3\ dBm/MHz$.

Antenna delay calibration. After the crystal oscillator and the transmit power and bandwidth, the antenna delay can be calibrated. The DW1000 supports a special transmit mode for time-of-flight measurements. In this mode, the timestamps of received and transmitted packets are saved to a register. The timestamp is internally corrected by taking the antenna delay into account, when reading the timestamp from the register. The antenna delay is the time duration between a frame being processed digitally and the corresponding RF signal passing the antenna. For reception and transmission the antenna delay is different. The ranging accuracy can be increased from 30cm to 4.5cm by calibrating the antenna delay [35]. A description of the test procedure can be found in Section A-C. Figure 11 shows the histogram of 10000 ranging measurements.

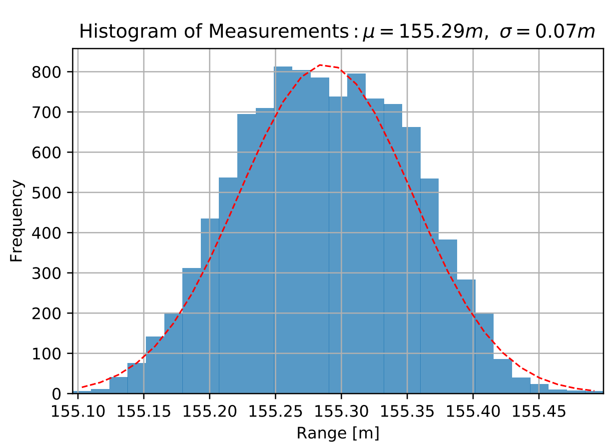


Fig. 11: Histogram of ranging measurements.

The test boards were connected using a 100cm long coaxial cable[7]. This gives a combined antenna delay of:

$$Delay_{TX,RX} = \frac{155.29m - 1.0m}{299702547\frac{m}{s}} = 514.81ns \tag{2}$$

The antenna delay for transmitted frames is then given by (3) and by (4) for received frames (see Appendix A-C). Processing received frames takes slightly longer than transmitting frames. The combined delay is apportioned 56% and 44% for the receiver delay and the transmitter delay respectively.

$$Delay_{RX} = 514.81ns \cdot 0.56 = 288.29ns \tag{3}$$

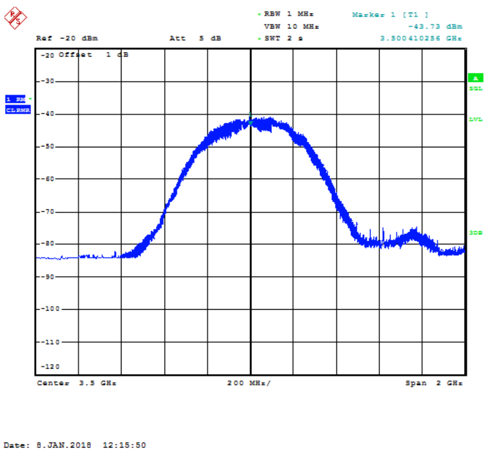
$$Delay_{TX} = 514.81ns \cdot 0.44 = 226.52ns \tag{4}$$

V. Conclusion

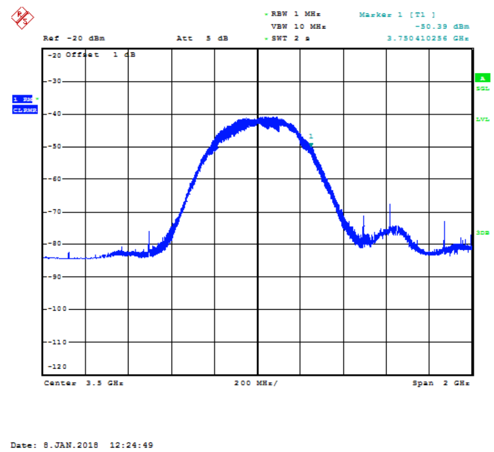
We designed a platform that fulfills all our requirements. Even if these requirements change over time, the flexible stacking design allows the adaption and replacement of single components. The platform is compatible with the Contiki operating system from a software perspective and with other Arduino boards from a hardware point of view. These features and the fact that it is a low-cost open source platform make it an interesting choice, especially for research projects.

Furthermore, we showed that its RF performance and ranging accuracy is comparable to Decawave's own reference design. For some channels the default transmit power and bandwidth settings were sufficient. On higher frequency channels minor adjustments had to be made to improve the performance.

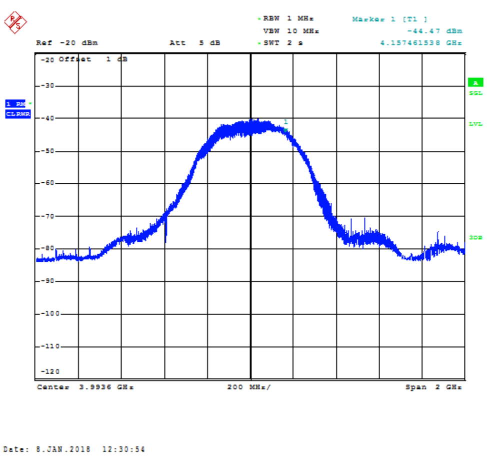
[7]Harbour Industries M17/152-00001 MIL-DTL-17, tested for a maximum frequency of 12.4 GHz.



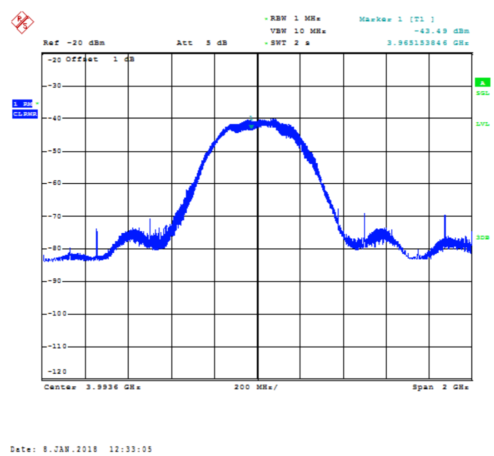
(a) Channel 1 at 16MHz PRF



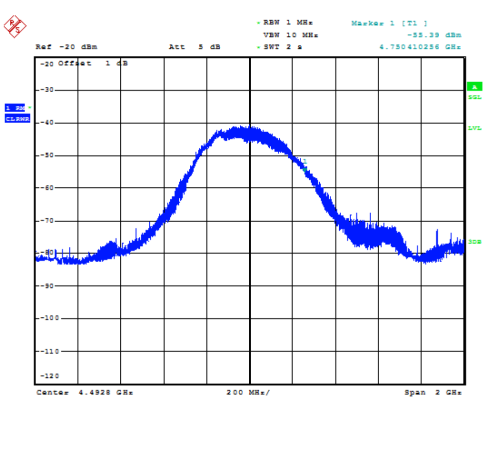
(b) Channel 1 at 64MHz PRF



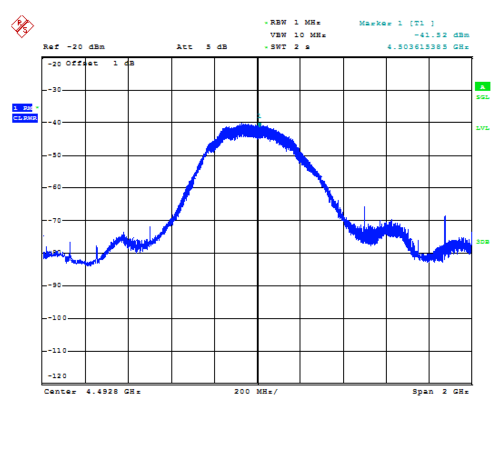
(c) Channel 2 at 16MHz PRF



(d) Channel 2 at 64MHz PRF

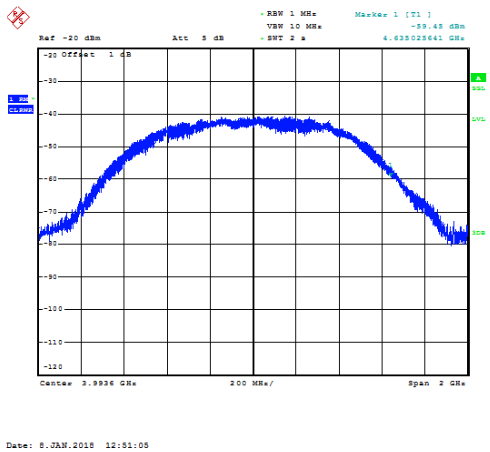


(e) Channel 3 at 16MHz PRF

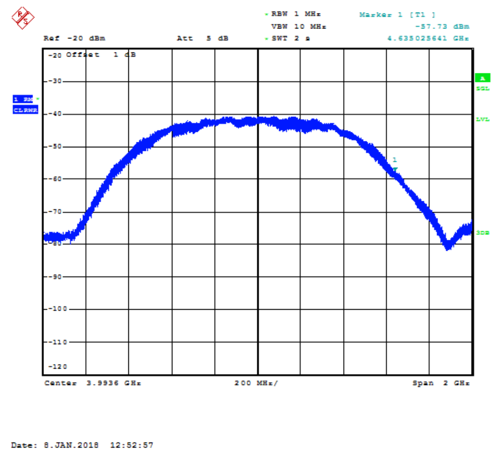


(f) Channel 3 at 64MHz PRF

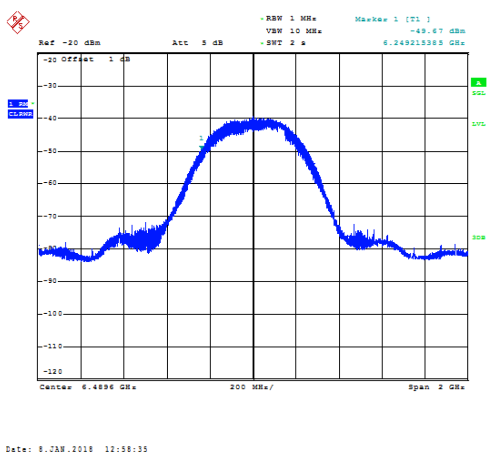
Fig. 12: Transmit Power and Bandwidth on Channels 1, 2, and 3 for 16 and 64MHz PRF, respectively.



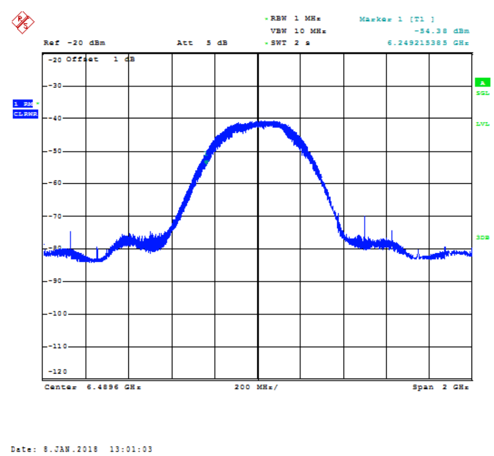
(a) Channel 4 at 16MHz PRF



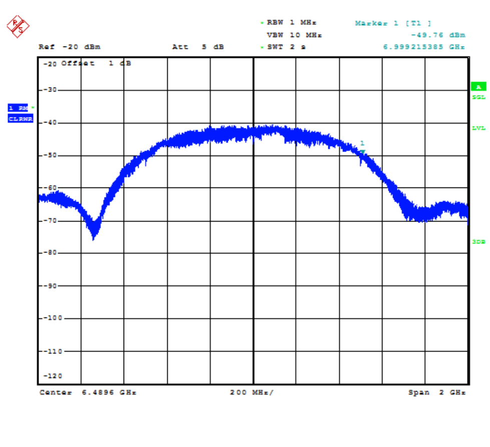
(b) Channel 4 at 64MHz PRF



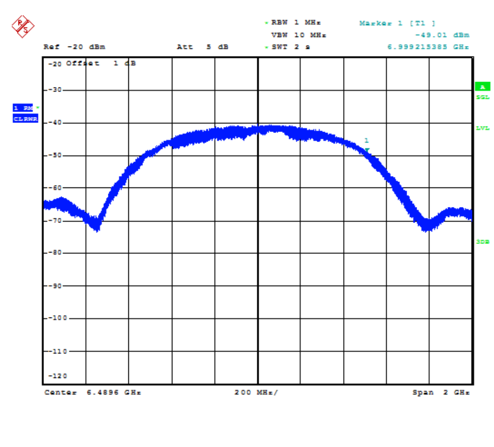
(c) Channel 5 at 16MHz PRF



(d) Channel 5 at 64MHz PRF



(e) Channel 7 at 16MHz PRF



(f) Channel 7 at 64MHz PRF

Fig. 13: Transmit Power and Bandwidth on Channels 4, 5, and 7 for 16 and 64MHz PRF, respectively.

REFERENCES

[1] W. Hu *et al.*, "The design and evaluation of a hybrid sensor network for cane-toad monitoring," April 2005.

[2] G. Werner-Allen *et al.*, "Monitoring volcanic eruptions with a wireless sensor network," Jan 2005.

[3] M. Ceriotti *et al.*, "Is there light at the ends of the tunnel? wireless sensor networks for adaptive lighting in road tunnels," April 2011.

[4] D. Ltd, *DW1000 Datasheet Version 2.09*, Decawave, 2015.

[5] Contiki: The open source os for the internet of things. The Contiki Community. [Online]. Available: https://www.contiki-os.org

[6] Openrtls. The Contiki Community. [Online]. Available: https://openrtls.com

[7] U. Company. (2017) Sequitur. [Online]. Available: http://www.unisetcompany.com/

[8] s.r.o. Sewio Networks. (2017) Sewio. [Online]. Available: http://www.sewio.net

[9] P. BVBA. (2015) Pozyx accurate positioning. Pozyx. [Online]. Available: https://www.poyzx.io

[10] B. Großwindhager *et al.*, "Poster: Switchable directional antenna system for uwb-based internet of things applications," in *Proceedings of the 2017 International Conference on Embedded Wireless Systems and Networks*, 2017, pp. 210–211.

[11] (2017) Lab11 polypoint. [Online]. Available: https://lab11.eecs.umich.edu/projects/polypoint/

[12] A. R. J. Ruiz and F. S. Granja, "Comparing ubisense, bespoon, and decawave uwb location systems: Indoor performance analysis," *IEEE Transactions on Instrumentation and Measurement*, vol. 66, no. 8, pp. 2106–2117, Aug 2017.

[13] Localino: Open source indoor localization system. [Online]. Available: https://hackaday.io/project/18296-localino-open-source-indoor-localization-system

[14] G. Nordahl. Lps mini. [Online]. Available: https://hackaday.io/project/7183/instructions

[15] D. Ltd, *EVK1000 User Manual Version 1.12*, Decawave, 2016.

[16] CIHOLAS. Dwusb. [Online]. Available: https://www.ciholas.com/

[17] ST, *IDW04A1 Wi-Fi expansion board based on SPWF04SA module for STM32 Nucleo*, ST Microelectronics.

[18] ——, *IKS01A1 Motion MEMS and environmental sensor expansion board for STM32 Nucleo*, ST Microelectronics.

[19] Arduino. (2017) Arduino website. Arduino. [Online]. Available: https://www.arduino.cc/

[20] M. Stocker, "Uwb dw1000 radio driver for the contiki os," May 2017.

[21] S. Microelectronics, *STM32L152 Datasheet DocId17659 Rev 12*, ST Microelectronics, 2016.

[22] Texane. (2017) St-link repository. [Online]. Available: https://github.com/texane/stlink

[23] B. Baumann. (2017) Uwb extension shield production tests. Graz University of Technology. [Online]. Available: https://github.com/BerndBaumann/

[24] A. LLC. (2017) Altium website. [Online]. Available: http://www.altium.com

[25] D. Ltd, *APH001: DW1000 Hardware Design Guide, Version 1.0*, Decawave, 2016.

[26] Multi-CB. (2017) Multi-circuit-board. [Online]. Available: https://www.multi-circuit-boards.eu

[27] I. Group. (2017) Isola-group de104 datasheet. [Online]. Available: http://www.isola-group.com/products/all-printed-circuit-materials/de104/

[28] KiCad. (2017) Kicad pcb calculator. [Online]. Available: http://kicad-pcb.org/

[29] N. Instruments. (2017) Tx-line: Transmission line calculator. [Online]. Available: http://www.awrcorp.com/products/additional-products/tx-line-transmission-line-calculator

[30] I. Saturn PCB Design. (2017) Saturn pcb design toolkit version 7.04. [Online]. Available: http://www.saturnpcb.com/pcb_toolkit.htm

[31] D. Ltd, *DWM1000 Datasheet Version 1.6*, Decawave, 2016.

[32] S. E. Corporation, *TSX-3225 Datasheet*, Seiko Epson Corporation, 2017.

[33] D. Ltd, *APS012: Production Tests for DW1000-Based Products, Version 1.3*, Decawave, 2015.

[34] ——, *DW1000 User Manual Version 2.10*, Decawave, 2016.

[35] ——, *APS014: DW1000 Antenna Delay Calibration Version 1.01*, Decawave, 2014.

APPENDIX A
PRODUCTION TEST MEASUREMENTS

Decawave provides a set of recommended production tests for products that contain the DW1000 transceiver [33]. They are divided in three groups of tests:

1) operational tests, where no RF functionality is tested;
2) RF measurements that require a Spectrum Analyzer;
3) RF measurements that require a compatible reference device.

All tests use the NUCLEO-L152RE board with the extension shield stacked onto its Arduino connectors. The source code and the firmware binaries for each test can be downloaded from [23]. To monitor the output of the firmware during each test, a serial terminal is needed. The communication setting is: *baudrate 115200 8N1*, no flow control. If the DW**M**1000 is soldered on the UWB shield too, jumper P5 must be closed during all tests.

To ensure that the extension shield works correctly and within legal boundaries, the following tests must be performed.

PHY Setting	**Value**
Channel	5
Pulse repetition frequency	16 MHz
Preamble symbol repetitions	128
Data rate	6.8 Mbps
Payload size (including MAC header)	12 Bytes

TABLE XI: Default configuration of the DW1000 radio on power-up.

A. Operational Tests

Operational tests do not use the RF section of the device under test (DUT). They test the DW1000s GPIO lines and SPI bus communication. During all operational tests the supply voltage and the supply current should be monitored. A voltmeter and an ampere meter are required for these tests.

1) Measure Idle Mode Current: This test measures the current consumption of the extension shield during the DW1000 IDLE state. After being connected to a power source, the DW1000 is in the WAKEUP state. The transceiver then initializes itself with the Decawave factory default settings (see Table XI) and waits until the crystal is stable and the RSTn is HIGH [34, Section 2.3]. After successful initialization, the DW1000 remains in IDLE state[8]. The jumper P1 on the extension shield was added in the design to measure its current consumption. It connects the 3V3 pin and the VCC plane.

a) Instructions:

1) Flash the test program named `01_idlemode.bin` to the NUCLEO-L152RE board.
2) Reset the device by pushing the button **B2** on the NUCLEO-L152RE board.
3) Remove all jumpers from the DUT (P1, P2, P4, P5 must be open).
4) Disconnect the NUCLEO-L152RE board from its USB power source.

[8] If current consumption is measured over time, it is possible to trigger on the rising edge of the RSTn as this firmware is performing a hardware-reset after power-up.

5) Connect the ampere meter to the pins of P1.
6) Connect the voltmeter to the 3V3 and the GND pins of the DUT.
7) While monitoring voltage and current, power the board again by plugging in the USB cable.
8) Measure the current and the voltage a few seconds after the power source was connected.

b) Remarks: The current consumption depends on the supply voltage and all other components within the test circuit. The limits according to [33, section 3.2] are min. 10mA and max. 25mA.

2) SPI test: This test can either be pass or fail. The device ID of every transceiver is programmed to address `0x000`. Reading this address must return the 4 byte device ID.

a) Instructions:

1) Flash the test program named `02_readdevid.bin` to the NUCLEO-L152RE board.
2) Jumper P1 must be closed, jumper P4 can be closed or open.
3) Jumper P2 must be in position **CS1**.
4) Reset the board by pressing the button **B2** on the NUCLEO-L152RE board.
5) If the green LED **LD2** on the NUCLEO-L152RE board is on, reading the device ID was successful. If the green LED is flashing, the test failed.

After this test, disconnect the NUCLEO-L152RE board from its power source for a few seconds. This is necessary to reset the configuration of the STM32's GPIO6 pin on port B, which is the same GPIO used for SPI-CS.

3) Maximum Current Consumption: In this test, the maximum current consumption in the listening mode and with Decawave default settings (see Table XI) is measured. The current consumption should be in the range 10mA to 25 mA.

a) Instructions:

1) Flash the test program named `03_max_current.bin` to the NUCLEO-L152RE board.
2) Jumper P4 must be open and jumper P2 must be in position CS1.
3) Optional: If a DWM1000 module is soldered on the board, then jumper P5 must be closed too.
4) Jumper P1 must be used to measure the current consumption. Connect the ampere-meter to P1.
5) Connect the voltmeter to the 3V3 pin and GND pin.
6) Reset the device by pushing the button **B2** on the NUCLEO-L152RE board.
7) Measure the current consumption and the voltage.

4) GPIO strobe test: The GPIO lines that are connected in the schematic must be tested. Therefore, the firmware sends the SPI commands to set (or reset) each of the GPIO pins. The firmware then reads the state of the GPIO pin automatically and verifies if it works properly (the jumpers GPIOx must be closed). Additionally, the GPIO level can be measured with a voltage probe on the extension shield.

The state LOW is defined in [4] as max. 0.3*VDDIO and the state HIGH as min. 0.7*VDDIO.

a) Instructions:

1) Flash the test program named `04_gpio.bin` to the NUCLEO-L152RE board.
2) Jumper P1 must be in position CS1, jumper P1 must be closed. If the LED_RXOK, LED_SFD, LED_RX, LED_TX are soldered on the board, jumper P4 must be closed. If a DWM1000 module is soldered on the board, jumper P5 must be closed too.
3) Reset the device by pushing the button **B2** on the NUCLEO-L152RE board.
4) Measure the voltage on every GPIO pin. It must be higher than 0.7*VDDIO.

5) RSTn test: After the board is powered, the line RSTn must go high. This happens when the DW1000 changes from its WAKEUP to INIT state. During the power-on phase, the DW1000 drives the RSTn line LOW. After 5 ms, the RSTn line must be HIGH.

a) Instructions:

1) Flash the test program named `05_reset.bin` to the NUCLEO-L152RE board.
2) Jumper P1 must be closed and jumper P2 must be in position CS1.
3) Reset the device by pushing the button **B2** on the NUCLEO-L152RE board.

6) WAKEUP test: To save energy, the DW1000 supports a sleep mode. After a command was sent to the DW1000 via SPI, the device goes to sleep and must wake up again after driving the WAKEUP line HIGH.

1) Flash the test program named `06_wakeup.bin` to the NUCLEO-L152RE board.
2) Reset the device by pushing the button **B2** on the NUCLEO-L152RE board.
3) Jumper P1 must be closed and jumper P2 must be in position CS1.
4) Connect the ampere meter to the pins of P1.
5) Connect the voltmeter to the 3V3 and the GND pins of the DUT.
6) Push the button **B1** to trigger a WakeUp.
7) Measure the current and the voltage after pushing the button. If the DW1000 enters IDLE state successfully the current consumption should be the same as in A-A1 (idle mode current).

a) Remark: The DW1000 enters the IDLE state after waking up again. Therefore, the current consumption must be the same as in A-A1.

7) EXTON test: This test is equivalent to A-A5 (RSTn test). Instead of the RSTn line, it is checked if the EXTON line goes HIGH after power is applied.

a) Instructions:

1) Flash the test program named `07_exton.bin` to the NUCLEO-L152RE board.

2) Jumper P1 must be closed and jumper P2 must be in position CS1. If a DWM1000 module is soldered on the board, JP5 must be closed too.
3) Reset the device by pushing the button **B2** on the NUCLEO-L152RE board.

B. Transmitter Calibration

The second group of tests measure and then calibrate the DUTs transmitter. The extension shield has a SMA RF connector to use various antennas. During calibration it is recommended in [33] to use a coax cable, if possible. Before measuring the transmit power of the DUT, the path loss P_{LOSS} of the SMA cable must be determined for every tested frequency in Table XIII. This systematic measurement error must then be added to the transmit power P_{SA} measured by the spectrum analyzer. The real transmit power of the DUT P_{DUT} is defined by (5). These measurements require a spectrum analyzer.

$$P_{DUT} = P_{SA} + P_{LOSS} \tag{5}$$

1) Crystal Trim: The DW1000 supports tuning the crystal oscillator by changing the value in register FS_XTAL `0x2B:0E` [34, Section 7.2.44.5]. The initial value of the trim register is `0x00` and the maximum value is `0x1F`. By pushing a button on the NUCLEO-L152RE board, the value of this register can be increased by 1. After reaching the maximum value the register will become `0x00` again. This needs to be done for one frequency only, because all carrier center frequencies are derived from the same oscillator frequency.

a) Instructions:

1) Flash the test program named `08_xtaltrim.bin` to the NUCLEO-L152RE board.
2) Jumper P1 must be closed and jumper P2 must be in position CS1.
3) Connect the DUT to the spectrum analyzer using a coax cable.
4) Connect the Voltmeter to the pins 3V3 and GND of the extension shield.
5) Configure the spectrum analyzer center frequency at 6489.6 MHz, the channel 5 center frequency.
6) Reset the device by pushing the button **B2** on the NUCLEO-L152RE board. The board will send a continuous wave (CW) signal on the channel 5 center frequency.
7) If the output signal is not equal to the channel 5 center frequency, increase the trim value by pushing button **B1** on the NUCLEO-L152RE board.
8) Count how often the button has been pushed and write down the count when the output signal was closest to the channel 5 center frequency. This value must be used to trim the crystal.

2) Transmit Power and Bandwidth Calibration: It is important to calibrate the transmit power and bandwidth for each channel that will be used during operation later-on. Otherwise it is possible that the extension shield violates EU or FCC spectrum regulations. If the transmit power is too low the performance of the extension shield is not optimal. To find the highest transmit power gain that does not violate regulations [33, Appendix A], the user can set a coarse gain in 3dB steps and a fine gain in 0.5dB steps. Furthermore, the bandwidth limits (-51.3dB outside the channel boundaries) must not be exceeded while the bandwidth should be as high as possible. To change the channel bandwidth, the PGDELAY is adjusted. Increasing the PGDELAY value results in wider transmit pulses, hence decreasing the channel bandwidth. Calibrating the transmission gain and the PGDELAY is done via a serial terminal interface. The DW1000 supports two modes for choosing the transmit power. A manual transmit power mode, where the gain is set to a fixed value for all transmission bit rates and frame lengths and a smart transmit power mode. The "Smart Tx Power" mode applies only for the highest bit rate of 6.8 Mbps and allows to boost the transmit power if the transmitted frame is shorter than one millisecond. This test procedure calibrates the manual transmit power mode only.

Resolution Bandwidth	1 MHz
Video Bandwidth	1 MHz
Span	2 GHz
Sweep time	2 seconds
Detector	rms
Average time per point	1 ms

TABLE XII: Spectrum analyzer settings for measuring transmit power

a) Instructions:

1) Flash the test program named `09_txbandwidth.bin` to the NUCLEO-L152RE board.
2) Jumper P1 must be closed and jumper P2 must be in position CS1.
3) Connect the DUT to the spectrum analyzer using a coax cable.
4) Connect the DUT to a computer using a USB cable. Open a serial terminal on this computer to display the NUCLEO-L152RE boards serial output. The settings for the serial connection are `115200 8N1, no flow control`.
5) Set the Spectrum Analyzer settings according to Table XII recommended in [34, Section 8.2].
6) Reset the device by pushing the button **B2** on the NUCLEO-L152RE board.
7) The DUT is now transmitting continuously at the highest power setting. The power spectrum must be visible on the spectrum analyzer.
8) Follow the instructions in the serial terminal to configure the TXPOWER and PGDELAY registers.
9) Once the optimal combination of PGDELAY and TXPOWER is found, write down the values in Table XIII.
10) Repeat steps 1 to 8 for every test number in Table XIII.

C. Antenna Delay Calibration

For this measurement a second DW1000 device is necessary. The antenna delay describes the time that the signal needs

Test number	Channel	Frequency	Bandwidth	PRF	Code	Coarse Gain	Fine Gain	PGDELAY
		[MHz]	[MHz]	[MHz]		dB	dB	
1	1	3494.4	499.2	16	1			
2	1	3494.4	499.2	64	9			
3	2	3993.6	499.2	16	3			
4	2	3993.6	499.2	64	9			
5	3	4492.8	499.2	16	5			
6	3	4492.8	499.2	64	9			
7	4	3993.6	1331.2	16	7			
8	4	3993.6	1331.2	64	9			
9	5	6489.6	499.2	16	3			
10	5	6489.6	499.2	64	9			
11	7	6489.6	1081.6	16	7			
12	7	6489.6	1081.6	64	17			

TABLE XIII: Measurements of transmit power and bandwidth.

to travel from the TX/RX ports of the DW1000 until it reaches the antenna. Because the DW1000 is an impulse radio (IR), it can measure distances very accurately through time-of-flight (ToF) measurement. This is done by the two-way ranging (TWR) application which measures the distance between two DW1000 transceivers. The antenna delay adds a constant error to this measurement that must be compensated for accurate distance measurements. The propagation velocity of the transmitted signals depends on the medium. This must be taken into account when calculating the distance from the measured time-of-flight. This test requires a computer with a serial terminal installed to monitor the DUTs output a reference test board and a coaxial cable.

a) Instructions:

1) Flash the test program named `11_twr.bin` to **two** NUCLEO-L152RE boards.
2) On both boards jumper P1 must be closed and jumper P2 must be in position CS1.
3) Connect both extension shields RF connectors with an coax cable.
4) At least one test board must be connected to a computer. Open a serial terminal on this computer to display the NUCLEO-L152RE boards serial output. The settings for the serial connection are `115200 8N1, no flow control`.
5) Reset both devices by pushing the button **B2** on the NUCLEO-L152RE boards. The serial terminal must show the output of the TWR application.
6) In the terminal the distance in meter is printed continuously. Take 30 measurements and calculate the mean D_{mean} and the standard deviation.
7) Measure the length of the coax cable. The antenna delay in meter can be calculated according to (6).

b) Formulas: Formula for calculating T_{delay}

$$Delay_{RX,TX} = \frac{\left(D_{mean,measured} - D_{cable}\right)}{speed\ of\ light} \quad (6)$$

The antenna delay for transmitted frames is then given by (7) and by (8) for received frames. Receiving frames takes slightly longer than transmitting frames in the DW1000. Therefore the combined delay is not apportioned equally, but 56% and 44% for the receiver delay and the transmitter delay respectively.

$$Delay_{RX} = Delay_{RX,TX} \cdot 0.56 \quad (7)$$

$$Delay_{TX} = Delay_{RX,TX} \cdot 0.44 \quad (8)$$

The programmed values of the delays must be a multiple of Decawave-time-units (dwtu). The are calculated by (9).

$$Delay_{dwtu} = \frac{Delay_{seconds}}{1\,/\,499.2 \cdot 10^6\,/\,128} \quad (9)$$

APPENDIX B
BILL OF MATERIALS

Comment	Pattern	Quantity	Components
0.10uF	CAP 0402/1005	15	C11, C12, C13, C14, C15, C16, C17, C18, C19, C20, C23, C25, C26, C27, C29
0732511150	MOLEX SD-73251-115	1	SMA
1.2pF	CAP 0402/1005	1	C6
100	RES 0402/1005	1	R7
10000pF	CAP 0402/1005	1	C22
100k	RES 0402/1005	1	R40
10pF	CAP 0402/1005	1	C30
11K 1%	RESC0603(1608)L	1	R3
12pF	CAP 0402/1005	2	C9, C10
16k	RES 0402/1005	1	R4
18pF	CAP 0402/1005	1	C8
270	RES 0402/1005	1	R5
27pF	CAPC0402(1005)60L	1	C5
330pF	CAP 0402/1005	2	C28, C31
38.4MHz	EPSON TSX-3225	1	Y1
4.7uF	CAPC0603(1608)100M	2	C1, C24
47uF	CAP 0805/2012	1	C21
8.5pF	CAPC0402(1005)60L	2	C3, C4
81-LXDC2HL18A-052	PCBComponent1	1	U2
820pF	CAP 0402/1005	1	C7
analog	HDR1X6	1	CN8
CurrentProbe	HDR1X2H	1	P1
digital	HDR1X10	1	CN5
digital	HDR1X8	1	CN9
DW1000 IC	QFN-48	1	U1
HHM1595A1	HHM1595A1	1	T1
LMK107BJ106MALTD	CAPC1608X100X35ML20T25	1	C2
POWER	HDR1X8	1	CN6

TABLE XIV: Bill of Materials.

APPENDIX C
BILL OF MATERIALS

Comment	Pattern	Quantity	Components
0.10uF	CAP 0402/1005	15	C11, C12, C13, C14, C15, C16, C17, C18, C19, C20, C23, C25, C26, C27, C29
0732511150	MOLEX SD-73251-115	1	SMA
1.2pF	CAP 0402/1005	1	C6
100	RES 0402/1005	1	R7
10000pF	CAP 0402/1005	1	C22
100k	RES 0402/1005	1	R40
10pF	CAP 0402/1005	1	C30
11K 1%	RESC0603(1608)L	1	R3
12pF	CAP 0402/1005	2	C9, C10
16k	RES 0402/1005	1	R4
18pF	CAP 0402/1005	1	C8
270	RES 0402/1005	1	R5
27pF	CAPC0402(1005)60L	1	C5
330pF	CAP 0402/1005	2	C28, C31
38.4MHz	EPSON TSX-3225	1	Y1
4.7uF	CAPC0603(1608)100M	2	C1, C24
47uF	CAP 0805/2012	1	C21
8.5pF	CAPC0402(1005)60L	2	C3, C4
81-LXDC2HL18A-052	PCBComponent1	1	U2
820pF	CAP 0402/1005	1	C7
analog	HDR1X6	1	CN8
CurrentProbe	HDR1X2H	1	P1
digital	HDR1X10	1	CN5
digital	HDR1X8	1	CN9
DW1000 IC	QFN-48	1	U1
HHM1595A1	HHM1595A1	1	T1
LMK107BJ106MALTD	CAPC1608X100X35ML20T25	1	C2
POWER	HDR1X8	1	CN6

TABLE XV: Bill of Materials.

APPENDIX D
INVESTIGATED PLATFORMS

	OpenRTLS	Pozyx	DecaWave	LPS/LPSmini	LPS2	sewio	Localino	Radino
Host MCU	**ATSAM4S2B**	**STM32F4-01RCT6**	**STM32-F105RC**	**ATMEGA328-MUR**	**nRF52**	**EFM32G**	**ATMega328P**	**STM32L151CC**
Architecture	ARM Cortex-M4	ARM Cortex-M4 32bit	ARM Cortex-M3	8bit	Cortex-M4F 32bit	Cortex-M3 32bit	8bit	Cortex-M3 32bit
Max. clock	120 Hz	84 MHz	72 MHz	12MHz		32MHz	12MHz	32MHz
Flash/RAM size	128kB/64kB	256kB/?	256kB/64kB	32kB/?	512kB/64kB	16-128kB/8-16kB		256kB/32kB
V supply	1.6-3.6V	1.7-3.6V	3.3	1.8V	1.7-3.6V	3V	1.8V	
Energy idle	1µA@1.8V & RTC	42µA-65µA @ 25 deg	1.2mA@125kHz, 3.3V, 25deg	750nA	1.9µA@3V with RTC on	900nA, wake on RTC	?	1.15µA, wake on RTC
Energy active	180µA/MHz	128µA with peripherals OFF	47mA@72MHz +peripherals	200µA@1MHz, 25deg	5.4mA with BLE RX/TX	180µA/MHz	?	185µA/MHz
Peripherals	**mUSB Port/JTAG**	**mUSB(UART)**	**mUSB/(UART)**	**mUSB**	**mUSB**	**?**	**Mini USB**	
IMU	LSM9DS0	yes	No	MPU-9250	yes	no	no	no
Temperature	HTS221	no	no	no	yes	no	no	no
Acceleration	LIS3DSH	no	no	MPU-9250		MMA8543	no	no
Pressure	LPS25H	yes	no	MS5611-01BA03	yes	yes	no	no
Power	Qi, LiPo	5V Jack or USB	USB	LiPo, LP3982ILD	LiPo	LiPo	USB	
Additional Bus IFs	Ethernet, PoE	I2C, IRQ	SPI, UART, JTAG, **SMA**	No	no	No	I2C	I2C, SPI, 2xUART, **SMA**
Other wireless IFs	anchor has 802.15.4 at 2.4 GHz	no	no	no	BLE, optional Wifi	no	Optional 802.11 via ESP8266	no
Firmware Access	**No. Licenses only provides API and Examples**	**no. API and Examples are public**	**? only if TREK1000 kit License available**	**no. just open source Arduino libs for DW1000**	**no**	**Maybe. HW layout available for License**	**After purchase using open source arduino libs**	**Yes**
Hardware Design Access	**no**	**no**	**Not for the board, but Altium Library for the DW1000**	**No**	**No**	**On request**	**On request**	**Yes, but only Package and Pinout. No RF Schematics**
Cost	7000EUR (Starter Kit) 3000EUR (SW + HW Design)	135EUR per tag (or Anchor) 1000EUR for devel. Kit	545USD for 2 Boards 925USD for TREK1000	160EUR out-of-stock	Not available	2850EUR for TdoA kit 500EUR for sniffer	63EUR for anchor	34.9EUR + ~15EUR for carrier board
Website	https://wiki.openrtls.com	https://www.pozyx.io	www.decawave.com	www.loligo.se	Not available	www.sewio.net	Www.localino.net	http://wiki.in-circuit.de/index.php5?title=radino32_DW1000

Fig. 14: Investigated platforms.

	Sequitur	Nanotron	Lab11	Ciholas	Spoonphone	insight sip
Host MCU	**Unknown**	**unknown**	**nRF51822**	**ATSAM4S4AA**	**unknown**	**nRF52832**
Architecture	unknown	unknown	ARM Cortex-M0 32bit	32-bit ARM Cortex-M4	unknown	ARM Cortex-M4 32bit
Max. clock	unknown	unknown		120 MHz	unknown	64 MHz
Flash/RAM size	unknown	unknown	256kB/32kB	256kB/64kB	unknown	512kB/64kB
V supply	unknown	3 - 5.5 V	1.8 - 3.6 V	1.6 - 3.6 V	unknown	1.7 - 3.6 V
Energy idle	unknown	4.5 uA - 600 uA	2.6 uA	1 uA	unknown	1.9 uA
Energy active	unknown	120 mA	10 mA	unknown	unknown	5.4 mA
Peripherals	**unknown**	**UART**	**SPI, I2C, UART**	**USB**	**unknown**	**GPIO, SWD**
IMU	yes	no	no	MPU-9250	unknown	no
Temperature	no	no	Yes, OC	no	unknown	OC
Acceleration	yes	no	no	MPU-9250	unknown	no
Pressure	yes	no	no	LPS25H	unknown	no
Power	Unknown	unknown	MAX8887EZK33	USB	unknown	no
Additional Bus IFs		50 Ohm for ext. antenna	JLink	no	unknown	SPI
Other wireless IFs	BLE	no	BLE	no	unknown	NFC, BLE
Firmware Access	**After purchase**	**no**	**yes**	**Only binaries**	**no**	**no**
Hardware Design Access	**no. (they use DWM1000)**	**no**	**yes**	**yes**	**no**	**no**
Cost	1100 EUR for Kit 105 EUR per tag	On request	unknown	180 USD per tag	unknown	unknown
Website	http://www.unisetcompany.com/products/sequitur	Nanotron.com	Lab11.eecs.umich.edu	www.ciholas.com/dwusb	http://bespoon.com/bespoon-products/	Www.insightsip.com

Fig. 15: Investigated platforms continued.