

International Telecommunication Union

**ITU-T**

TELECOMMUNICATION  
STANDARDIZATION SECTOR  
OF ITU

**Q.4062**

(09/2020)

SERIES Q: SWITCHING AND SIGNALLING, AND  
ASSOCIATED MEASUREMENTS AND TESTS

Testing specifications – Testing specifications for  
IMT-2020 and IoT

---

## Framework for IoT testing

Recommendation ITU-T Q.4062



ITU-T Q-SERIES RECOMMENDATIONS  
**SWITCHING AND SIGNALLING, AND ASSOCIATED MEASUREMENTS AND TESTS**

SIGNALLING IN THE INTERNATIONAL MANUAL SERVICE	Q.1–Q.3
INTERNATIONAL AUTOMATIC AND SEMI-AUTOMATIC WORKING	Q.4–Q.59
FUNCTIONS AND INFORMATION FLOWS FOR SERVICES IN THE ISDN	Q.60–Q.99
CLAUSES APPLICABLE TO ITU-T STANDARD SYSTEMS	Q.100–Q.119
SPECIFICATIONS OF SIGNALLING SYSTEMS No. 4, 5, 6, R1 AND R2	Q.120–Q.499
DIGITAL EXCHANGES	Q.500–Q.599
INTERWORKING OF SIGNALLING SYSTEMS	Q.600–Q.699
SPECIFICATIONS OF SIGNALLING SYSTEM No. 7	Q.700–Q.799
Q3 INTERFACE	Q.800–Q.849
DIGITAL SUBSCRIBER SIGNALLING SYSTEM No. 1	Q.850–Q.999
PUBLIC LAND MOBILE NETWORK	Q.1000–Q.1099
INTERWORKING WITH SATELLITE MOBILE SYSTEMS	Q.1100–Q.1199
INTELLIGENT NETWORK	Q.1200–Q.1699
SIGNALLING REQUIREMENTS AND PROTOCOLS FOR IMT-2000	Q.1700–Q.1799
SPECIFICATIONS OF SIGNALLING RELATED TO BEARER INDEPENDENT CALL CONTROL (BICC)	Q.1900–Q.1999
BROADBAND ISDN	Q.2000–Q.2999
SIGNALLING REQUIREMENTS AND PROTOCOLS FOR THE NGN	Q.3000–Q.3709
SIGNALLING REQUIREMENTS AND PROTOCOLS FOR SDN	Q.3710–Q.3899
TESTING SPECIFICATIONS	Q.3900–Q.4099
Testing specifications for next generation networks	Q.3900–Q.3999
Testing specifications for SIP-IMS	Q.4000–Q.4039
Testing specifications for Cloud computing	Q.4040–Q.4059
<b>Testing specifications for IMT-2020 and IoT</b>	<b>Q.4060–Q.4099</b>
PROTOCOLS AND SIGNALLING FOR PEER-TO-PEER COMMUNICATIONS	Q.4100–Q.4139
SIGNALLING REQUIREMENTS AND PROTOCOLS FOR IMT-2020	Q.5000–Q.5049
COMBATING COUNTERFEITING AND STOLEN ICT DEVICES	Q.5050–Q.5069

*For further details, please refer to the list of ITU-T Recommendations.*

# Recommendation ITU-T Q.4062

## Framework for IoT testing

### Summary

The Internet of things (IoT) is one of the global infrastructures for the information society, delivering advanced services by interconnecting things based on, existing and evolving, interoperable information and communication technologies. Such a global infrastructure can be achieved by use of multiple access technologies for different types of communication networks such as body area network (BAN), personal area network (PAN), local area network (LAN), wireless local area network (WLAN), low power wireless access network (LPWAN), field area network (FAN), metropolitan area network (MAN), wide area network (WAN) and cellular networks. Conformance and interoperability tests not only for domains with single access technology but also for the integrated domains with multiple access technologies are required. The main goal of this Recommendation is to specify the testing framework for IoT to accommodate the tests for such integrated domains with multiple access technologies. Conformance and interoperability tests for domains served by single unified access technology have been taken into account by relevant SDOs and therefore are out of scope of this Recommendation.

Recommendation ITU-T Q.4062 describes the types of the tests for the domains with multiple access technologies and specifies the test procedures and the considerations corresponding to the testing types.

### History

Edition	Recommendation	Approval	Study Group	Unique ID*
1.0	ITU-T Q.4062	2020-09-29	11	<a href="http://handle.itu.int/11.1002/1000/14387">11.1002/1000/14387</a>

### Keywords

IoT devices, IoT technologies, test procedure, test specification.

---

\* To access the Recommendation, type the URL <http://handle.itu.int/> in the address field of your web browser, followed by the Recommendation's unique ID. For example, <http://handle.itu.int/11.1002/1000/11830-en>.

## FOREWORD

The International Telecommunication Union (ITU) is the United Nations specialized agency in the field of telecommunications, information and communication technologies (ICTs). The ITU Telecommunication Standardization Sector (ITU-T) is a permanent organ of ITU. ITU-T is responsible for studying technical, operating and tariff questions and issuing Recommendations on them with a view to standardizing telecommunications on a worldwide basis.

The World Telecommunication Standardization Assembly (WTSA), which meets every four years, establishes the topics for study by the ITU-T study groups which, in turn, produce Recommendations on these topics.

The approval of ITU-T Recommendations is covered by the procedure laid down in WTSA Resolution 1.

In some areas of information technology which fall within ITU-T's purview, the necessary standards are prepared on a collaborative basis with ISO and IEC.

## NOTE

In this Recommendation, the expression "Administration" is used for conciseness to indicate both a telecommunication administration and a recognized operating agency.

Compliance with this Recommendation is voluntary. However, the Recommendation may contain certain mandatory provisions (to ensure, e.g., interoperability or applicability) and compliance with the Recommendation is achieved when all of these mandatory provisions are met. The words "shall" or some other obligatory language such as "must" and the negative equivalents are used to express requirements. The use of such words does not suggest that compliance with the Recommendation is required of any party.

## INTELLECTUAL PROPERTY RIGHTS

ITU draws attention to the possibility that the practice or implementation of this Recommendation may involve the use of a claimed Intellectual Property Right. ITU takes no position concerning the evidence, validity or applicability of claimed Intellectual Property Rights, whether asserted by ITU members or others outside of the Recommendation development process.

As of the date of approval of this Recommendation, ITU had not received notice of intellectual property, protected by patents, which may be required to implement this Recommendation. However, implementers are cautioned that this may not represent the latest information and are therefore strongly urged to consult the TSB patent database at <http://www.itu.int/ITU-T/ipr/>.

© ITU 2021

All rights reserved. No part of this publication may be reproduced, by any means whatsoever, without the prior written permission of ITU.

## Table of Contents

	<b>Page</b>
1 Scope .....	1
2 References.....	1
3 Definitions .....	1
3.1 Terms defined elsewhere.....	1
3.2 Terms defined in this Recommendation.....	2
4 Abbreviations and acronyms .....	2
5 Conventions .....	4
6 Testing types.....	4
7 Test procedure .....	4
7.1 Conformance test.....	4
7.2 Connectivity test.....	5
7.3 Compatibility test .....	7
7.4 Response time test for network type classification .....	9
8 Consideration for test procedure.....	11
8.1 Testing network .....	11
8.2 Identification systems .....	12
8.3 Test timing.....	13
8.4 Grouping of target devices .....	13
8.5 Remote testing of IoT-devices.....	15
Annex A – Testing specifications .....	20
Appendix I – Estimation method on network types from RTT samples .....	42
Appendix II – Examples of IoT device detection and classification .....	45
Appendix III – Detail test procedure for LoRa connectivity .....	49
III.1 LoRa devices test.....	49
III.2 LoRa receivers test .....	49
III.3 LoRa transmitter test .....	50
III.4 Additional testing .....	51
Bibliography.....	52



# Recommendation ITU-T Q.4062

## Framework for IoT testing

### 1 Scope

The Internet of things (IoT) is achieved by use of multiple access technologies for different types of communication networks such as body area network (BAN), personal area network (PAN), local area network (LAN), wireless local area network (WLAN), low power wireless access network (LPWAN), field area network FAN, metropolitan area network (MAN), wide area network WAN and cellular networks. Therefore, conformance and interoperability tests not only for domains with single access technology but also for the integrated domains with multiple access technologies are required. The main goal of this Recommendation is to specify the testing framework for IoT to accommodate the tests for such integrated domains with multiple access technologies. Conformance and interoperability tests for domains served by single unified access technology have been taken into account by relevant standards developing organizations (SDOs) and thus are out of scope in this Recommendation.

### 2 References

The following ITU-T Recommendations and other references contain provisions which, through reference in this text, constitute provisions of this Recommendation. At the time of publication, the editions indicated were valid. All Recommendations and other references are subject to revision; users of this Recommendation are therefore encouraged to investigate the possibility of applying the most recent edition of the Recommendations and other references listed below. A list of the currently valid ITU-T Recommendations is regularly published. The reference to a document within this Recommendation does not give it, as a stand-alone document, the status of a Recommendation.

- [ITU-T Q.3952] Recommendation ITU-T Q.3952 (2018), *The architecture and facilities of a model network for Internet of things testing.*
- [ITU-T Y.2060] Recommendation ITU-T Y.4000/Y.2060 (2012), *Overview of the Internet of things.*
- [ITU-T Y.2069] Recommendation ITU-T Y.4050/Y.2069 (2012), *Terms and definitions for the Internet of things.*
- [ITU-T Y.4500.15] Recommendation ITU-T Y.4500.15/Q.3955 (2018), *oneM2M – Testing framework.*
- [ITU-T X.800] Recommendation ITU-T X.800 (1991), *Security architecture for Open Systems Interconnection for CCITT applications.*

### 3 Definitions

#### 3.1 Terms defined elsewhere

This Recommendation uses the following terms defined elsewhere:

**3.1.1 device** [ITU-T Y.2060]: With regard to the Internet of things, this is a piece of equipment with the mandatory capabilities of communication and the optional capabilities of sensing, actuation, data capture, data storage and data processing.

**3.1.2 Internet of things (IoT)** [ITU-T Y.2060]: A global infrastructure for the information society, enabling advanced services by interconnecting (physical and virtual) things based on existing and evolving interoperable information and communication technologies.

**3.1.3 conformance testing** [ITU-T Y.4500.15]: Process for testing that an implementation is compliant with a protocol standard, which is realized by test systems simulating the protocol with test scripts executed against the implementation under test.

**3.1.4 denial of service (DOS)** [ITU-T X.800]: The prevention of authorized access to resources or the delaying of time-critical operations.

## **3.2 Terms defined in this Recommendation**

None.

## **4 Abbreviations and acronyms**

This Recommendation uses the following abbreviations and acronyms:

6LoWPAN	IPv6 over Low power Wireless Personal Area Networks
ACK	Acknowledgment
ADC	Analogue to Digital Converters
AP	Access Point
ARB	Arbitrary Waveform Generators
BAN	Body Area Network
BER	Bit Error Rate
BLE	Bluetooth Low Energy
CDF	Cumulative Distribution Function
CIR	Committed Information Rate
CoAP	Constrained Application Protocol
CRC	Cyclic Redundancy Check
DDoS	Distributed Denial of Service
DTE	Data Terminal Equipment
DUT	Device Under Test
EDGE	Enhanced Data Rates for Global System for Mobile Communications (GSM) Evolution
EIR	Excess Information Rate
FAN	Field Area Network
FCC	Federal Communications Commission
FIN	Finish
GPRS	General Packet Radio Service
GW	Gateway
HTTP	Hypertext Transfer Protocol
HTTPS	Hypertext Transfer Protocol Secure
IANA	Internet Assigned Numbers Authority
ICMP	Internet Control Message Protocol
ID	Identifier



IoT	Internet of Things
IP	Internet Protocol
IS	Identification System
ISM	Industrial Scientific and Medical
JSON	JavaScript Object Notation
LE	Low Energy
LPWAN	Low Power Wireless Access Network
LTE	Long Term Evolution
LTE-A	LTE-Advanced
M2M	Machine-to-Machine
MAC	Medium Access Control
MAN	Metropolitan Area Network
MQTT	Message Queue Telemetry Transport
NB-IoT	Narrow Band Internet of Things
OBW	Occupied Bandwidth
OS	Operating System
PAN	Personal Area Network
PER	Packet Error Rate
PON	Passive Optical Network
QE	Qualified Equipment
QoE	Quality of Experience
QoS	Quality of Services
RBW	Resolution Bandwidth
REFhi	Maximum Radiated Power for highest channel frequency
REFlo	Maximum Radiated Power for lowest channel frequency
RF	Radio Frequency
RSSI	Received Signal Strength Indicator
RST	Reset
RTMP	Real Time Messaging Protocol
RTSP	Real Time Streaming Protocol
RTT	Round Trip Time
Rx	Receiver
SDC	Smart Device Communications
SDO	Standards Developing Organization
SFD	Start of Frame Delimiter
SLA	Service Level Agreement
SQL	Structured Query Language

SSID	Service Set Identifier
SYN	Synchronization
TCP	Transmission Control Protocol
TIA	Telecommunications Industry Association
UART	Universal Asynchronous Receiver/Transmitter
UDP	User Datagram Protocol
VBW	Video Bandwidth
WAN	Wide Area Network
WLAN	Wireless Local Area Network
WEP	Wired Equivalent Privacy
WPA	Wi-Fi Protected Access

## 5 Conventions

None.

## 6 Testing types

In order to test performance and to check characteristics of IoT devices, there are four types of testing. To check the characteristics of the IoT devices, the technical feasibility of interacting with network units, and the consistency in the operation of various types of networks, the network and all its components should be tested comprehensively. This procedure implies the following set of measures:

- **Conformance test** is a type of testing for certifying the technical characteristics declared by technical standards.
- **Connectivity test** is a type of testing to determine whether IoT hardware and network elements interact correctly within their own architecture, and how this network interacts with the infrastructures of various networks.
- **Compatibility test** is a type of testing to verify the compatibility of IoT hardware and IoT software from different suppliers during its operation on the network.
- **Response time test for network type classification** is defined as a type of testing in which the type of the target network is estimated from the measured response time of the network. If the target network is not congested, in general, a packet injected into the target network will arrive at its destination IoT device with a certain level of transmission delay.

## 7 Test procedure

### 7.1 Conformance test

Conformance requirements for IoT devices and systems which consist of different IoT technologies should be determined from requirements for the relevant standards. Conformance tests for IoT framework are similar to tests regarding to the relevant test specification standards. Due to the fact that the characteristics for different IoT devices differ, the methodology for conducting conformance testing also varies.

Conformance testing is not in the scope of this Recommendation.

## **7.2 Connectivity test**

### **7.2.1 Connectivity test**

The IoT framework should have the interfaces which enable a device to communicate with other devices over the network. The interfaces are based on specific IoT communication technologies or protocols. IoT framework connectivity tests are based on testing procedures which are described in standards or developer manual's requirements for the interface.

For example, if an interface for testing finished its work, the framework's user should check whether or not testing finished successfully by comparing every decelerated operation by framework developers (in special standard or manual) with the result of this operation obtained by the test. Comparison can be both quantitative and qualitative.

The IoT framework should have unified and multi-platform support with a configuration manager and can include several modules with expansion abilities, such as the IoT test tool, communication technology definition module, real-time statistics logger, real radio frequency (RF) channel modelling module, capture module, graphical user interface, instrument and test set remote controller module, input data modeler, security checker, and demo configurator wizard.

The communication technology definition module describes RF signals used in testing and uses a set of waveforms and can control external instruments to form real signals usable across all test platforms interacting with IoT devices.

The IoT test tool module based on a web server that works in application and network layers forms a compatibility test between different types and versions of hardware and firmware, module vendors, software, networks. The IoT test tool module can also act as a web-based system for testing interoperability between different IoT vendors, testing of applications on different operating systems (OS) and devices (desktop/mobile).

The capture module can measure the performance of IoT devices at the protocol layer and test end-to-end application with encoded data from sensors via network layer and measure packet error trends.

The instrument and test set remote controller module can communicate with a wide variety set of external instruments for additional testing. RF physical layer forming, triggering and measurement, data collecting, power consumption measurement.

IoT devices have strict requirements for power consumption in order to maintain expected battery lifetimes for multiple years without a connection to a power supply. In order to verify the power consumption and expected battery lifetime, voltage and current have to be measured accurately for different operating states of the device under test.

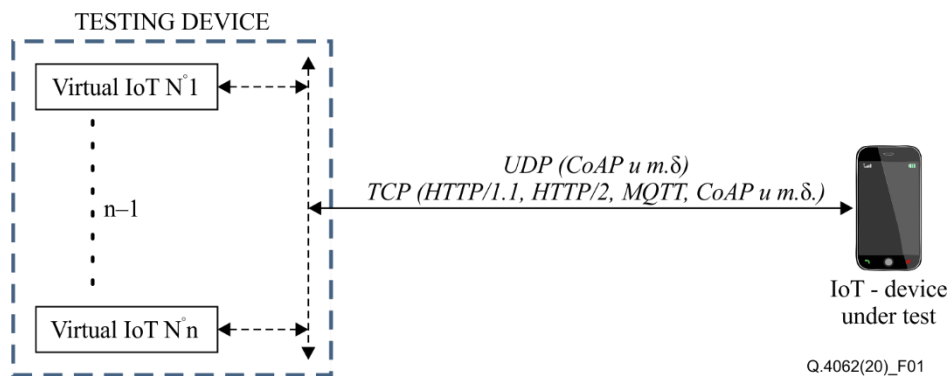
The detailed testing procedures are shown in Annex A of this Recommendation.

### **7.2.2 Stress test**

Stress test of IoT-devices aims to check the interworking between IoT devices and network infrastructure. The verification of the routers' stable performance to the IoT traffic is one of the most important objectives of the IoT stress test. The characteristics of traffic and parameters of IoT are different from existing traffic models.

Checking the performance of the application and the system is also important. The results of testing can help planning development and implementation of IoT systems.

The network capacity, latency, traffic value, packets loss, number of devices (that may be enormous) and so on should be considered during the stress test. Furthermore, the device and environment parameters such as energy consumption and temperature should also be considered. The testing model of IoT devices stress testing is shown in Figure 1.



**Figure 1 – The IoT-devices stress testing model**

Virtual IoT in testing devices is a simulated function of real IoT devices when it is required to simulate a high load from many IoT devices. Virtual devices have all the attributes of physical IoT devices and contain an identifier (for example, a MAC address) and a logical address (for example, IPv4).

For device stress testing, the testing device needs to generate typical traffic of IoT devices that are characterised by:

- packet size. The data field is in most cases smaller than the packet header;
- number of packets due to the large number of IoT things connected to the network;
- other laws of time distribution. In addition to the properties of self-similarity, which was observed in communication networks of the next generation, Internet of things can generate anti-persistent traffic.

For stress testing of the IoT devices, the following functions are to be considered:

- 1) Possibility of parallel traffic generation from virtual IoT to create a set of testing devices with unique parameters (unique MAC, IP, identification number, etc.).
- 2) Each virtual IoT running on a device needs to have one of the main types of IoT devices, for example:
  - sensor (an electronic device that measures the physical state or chemical composition and delivers an electronic signal corresponding to the observed characteristic);
  - actuator (a device that initiates a physical action after being excited by an input signal);
  - multimedia device (a device that sends multimedia information, i.e., digital information in which many types of information content and information processing are used to inform or entertain users, for example, text, images, audio, video, three-dimensional panoramic images and digital maps).
- 3) Support for the main scenarios of IoT operation. Each IoT has a unique algorithm of work laid down in it by the developer, but there are major types of IoT scenarios as follows:
  - constant data sending. Each device periodically sends data to a remote cloud server. In this case, the device can have two data types of sensor and of multimedia device. They generate self-similar traffic. For this scenario, the most commonly used transport layer protocols are connectionless, such as user datagram protocol (UDP), or less commonly used protocols that support the establishment of a connection such as transmission control protocol (TCP);
  - sending data on request. In this case, the device can have two data types of sensor, and of multimedia device. It can generate both self-similar traffic and antipersistent traffic, depending on the algorithm running on the remote control server. For this scenario, connection-oriented transport layer protocols (for example, TCP) are most often used,

in some cases, for example, in cases where both the IoT and the cloud server have an address in the global network, it is possible to use connectionless protocols (for example, UDP);

- impact on the surrounding world from a device by a control signal. In this case, the device can be the actuator type. It can generate both self-similar traffic and antipersistent traffic, depending on the algorithm running on the remote control server. For this scenario, the most commonly used transport layer protocols are connection-oriented (for example, TCP), in some cases, it is possible to use connectionless protocols (for example, UDP).
- 4) The ability to generate various types of traffic from each IoT according to a certain distribution law.
  - 5) Communication with a remote test server, required for testing scenarios that require the use of connectivity transport protocols. This server is necessary to measure the latency level at high levels of equipment load with IoT traffic.

In addition to support the capabilities of the above stress testing model, the device needs to support various major types of physical data transfer interfaces such as passive optical network (PON) interface, Ethernet interface, and WLAN interface.

### 7.3 Compatibility test

Compatibility test of an item measures how satisfactorily the item works in parallel with other independent elements in a common environment (co-existent) and, as necessary, interacts with other systems or system elements.

The consideration points of the compatibility testing of various IoT technologies are inter-protocol data conversion and identification of the protocol itself using certain (semantic) attributes to ensure the compatibility of devices and platforms of the IoT. Compatibility issues can be addressed by considering specific compatibility levels for devices, applications, systems, networks, and other elements of IoT systems.

Figure 2 to Figure 4 show architectures for compatibility testing of IoT devices, IoT gateways and IoT cloud, which are elements of IoT eco-system.

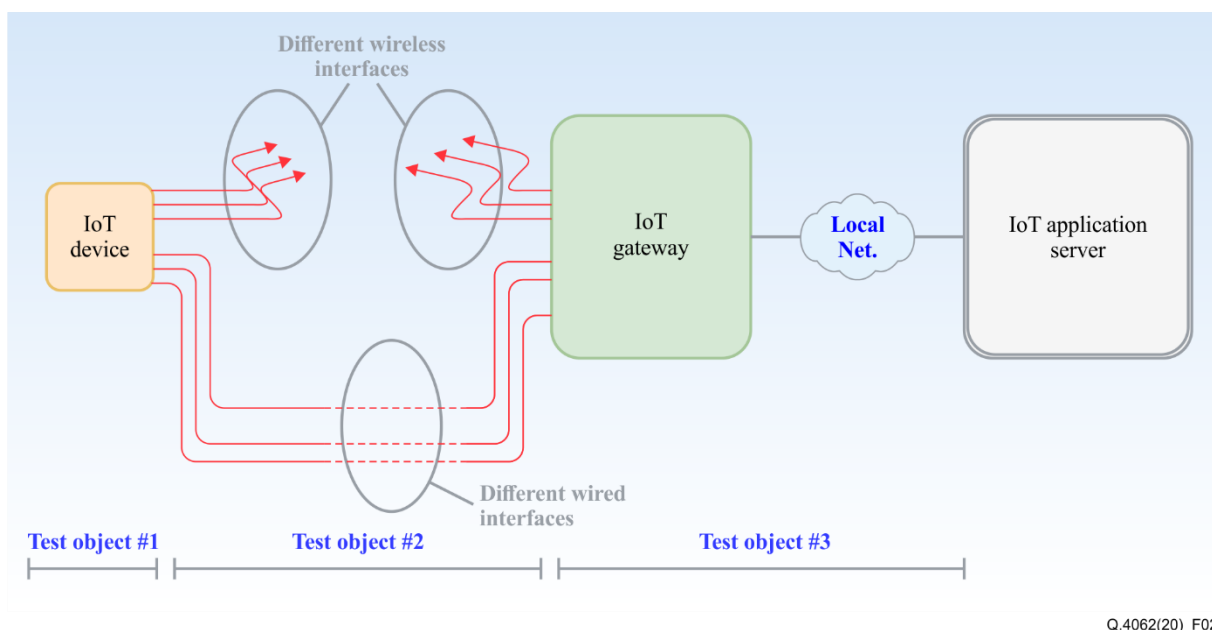
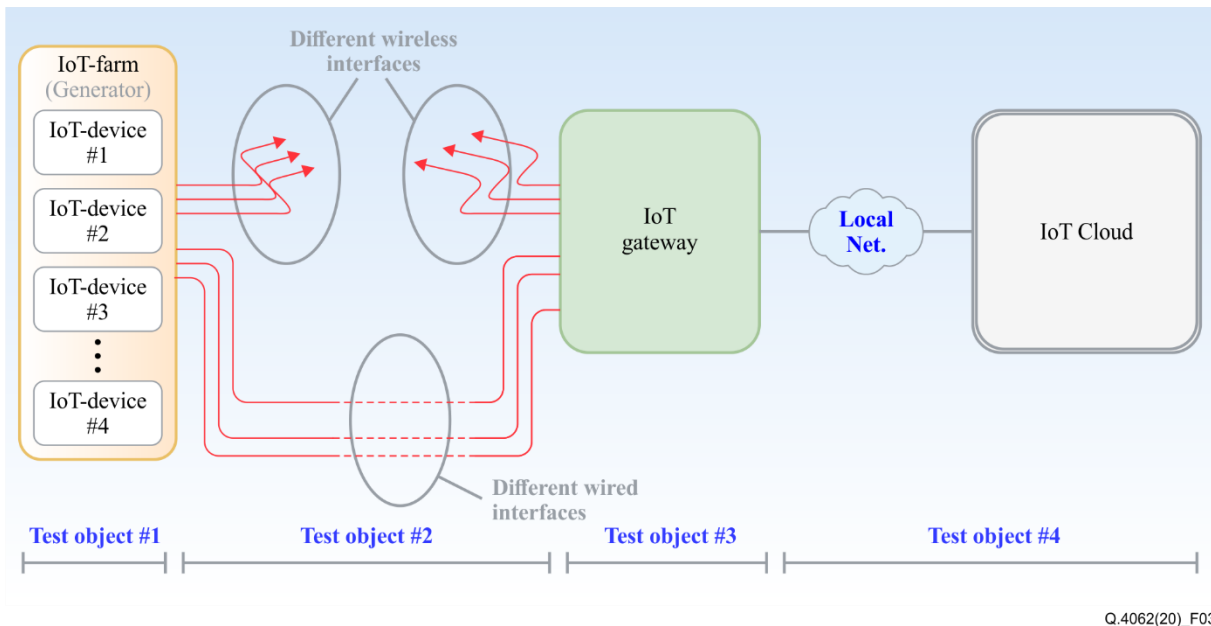


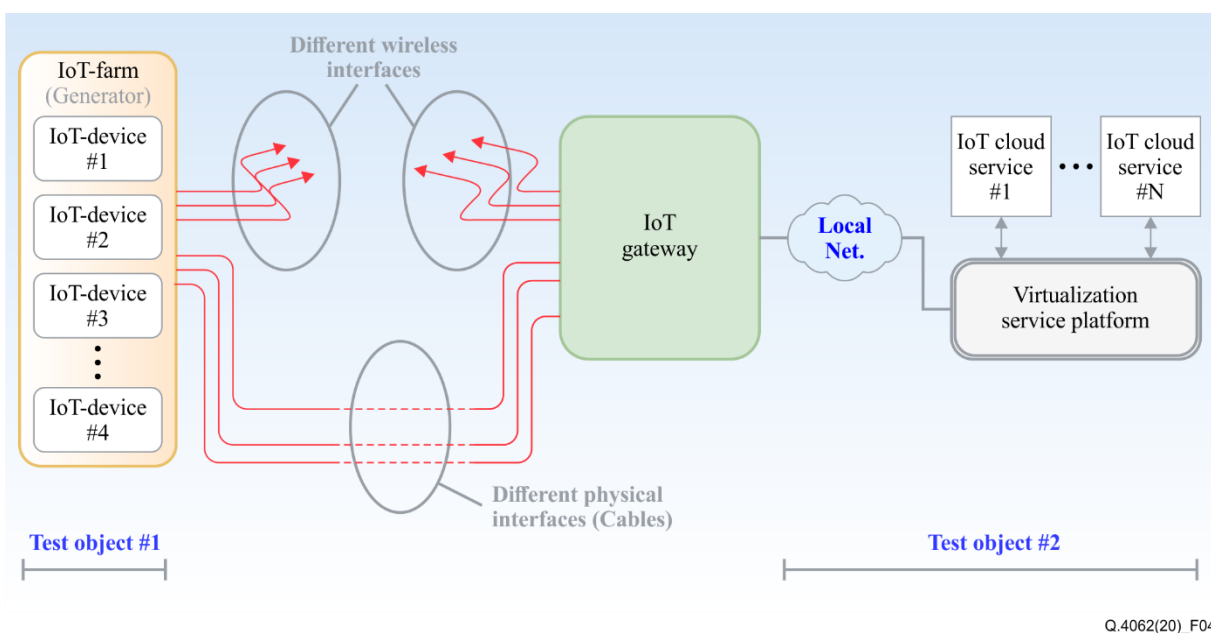
Figure 2 – Compatibility IoT-device testing

In case of IoT device testing, the following test objects, as shown in Figure 2, are tested: IoT-device, different wired and wireless interfaces and the network part, which consist of IoT-gateway and local network to server. The IoT-device testing includes testing of IoT interfaces between sensors and actuators, processes of data sending via different network interfaces (Figure 2), and compatibility with IoT Application Server through the network.



**Figure 3 – Compatibility IoT-gateway testing**

In case of IoT gateway testing, the following test objects, as shown in Figure 3, are tested: IoT-farm, which consists of the number IoT-devices, wired and wireless channels between IoT-farm and IoT gateway, IoT-gateway, network with IoT application server like the "black box". The IoT-gateway testing includes compatibility testing with IoT-device from IoT-farm via defined IoT-gateway's interfaces (wired and wireless), high-load testing of IoT-gateway from IoT-farm, and also compatibility testing between IoT-gateway and IoT application server (via local network and server's API).



**Figure 4 – Compatibility IoT-Cloud service testing**

In the case of IoT cloud service testing, the following test objects, as shown in Figure 4, are tested: IoT-farm, local network with virtualization service platform, which have a different IoT application servers. The IoT cloud service testing includes the compatibility testing between IoT-device and different IoT application servers (API and defined functionalities), high-load testing the IoT application server with help of IoT-farm.

### Compatibility levels

For the compatibility testing, the following compatibility levels should be considered:

- **Fully compatible:** In IoT systems, information resources or other IoT entities need to be capable of exchanging information and performing their required functions in a shared environment without any need of modifying their input and/or output interfaces, protocols, software and hardware or converting devices (adapters, gateways, etc.).
- **Compatible:** In IoT systems, information resources or other IoT entities need to be capable of exchanging information and performing their required functions in a shared environment by adapting their input and/or output interfaces, used protocols, software and hardware to each other or to the environment, or converting devices (adapters, gateways, etc.).
- **Partially compatible:** In IoT systems, information resources or other IoT entities need to be capable of exchanging information to some extent and performing a constrained set of their required functions in a shared environment, a help of additional tools unifying or converting their input and/or output interfaces, used protocols, procedures implemented by their software and hardware. The partial compatibility may be reached by negotiating the acceptable constraints on functionality among the parties.
- **Incompatible:** In IoT systems, information resources or other IoT entities are not capable of exchanging information and performing even a constrained set of their required functions in a shared environment due to the significant differences in technology and functional or non-functional requirements.

### 7.4 Response time test for network type classification

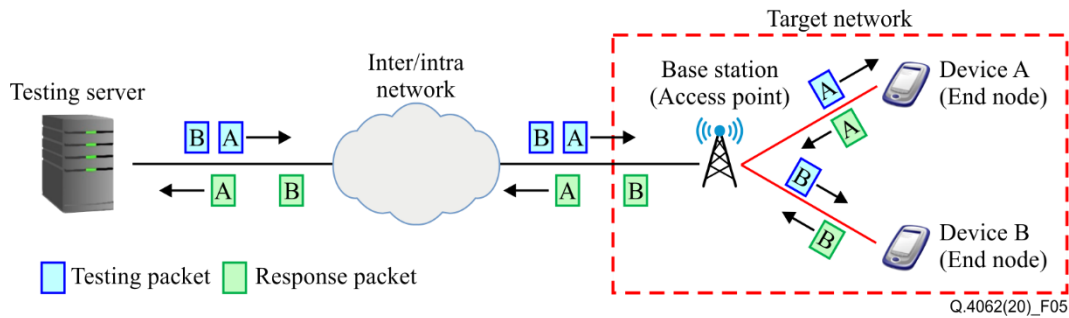
Response time test for network type classification measures the response time of the target network and estimates its network type. The response time is highly dependent on the transmission delay in the target network. The range of the transmission delay in non-congested situations is highly dependent on the network type of the target network, and the network type can be classified by comparing the transmission delay in a non-congested situation of the testing packet injected into the target network and the range of the transmission delay that is obtained in advance. Appendix I shows in detail the method of the network type classification.

In this method, the round trip time (RTT) between the testing server and IoT device(s) in the target network is used as the measure of response time. The testing server sends testing packet(s) to IoT device(s) in the target network. The IoT device sends back response packet(s) to the testing server once it receives the testing packet(s). The testing server measures, collects the round trip times (RTTs) of the packets, and then classifies the RTT values to estimate the type of the target network.

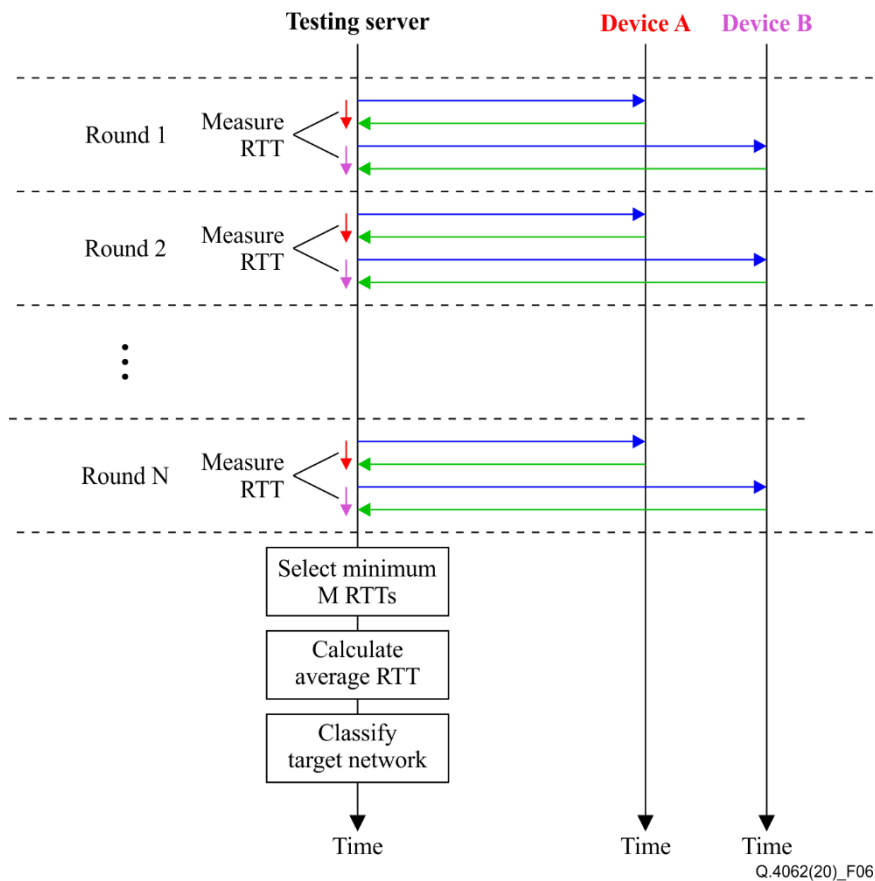
An example of the testing network is shown in Figure 5, and the procedure of the network type classification test is illustrated in Figure 6. The procedure of this test method is as follows:

- 1) Send  $P$  testing packets (as shown by blue boxes in Figure 5) to  $Q$  devices in the target network.
- 2) Receive the response(s) (as shown by green boxes in Figure 5) from the device(s) in the target network, and measure their RTT as the elapsed time from sending testing packet(s) to receiving response packets.

- 3) Repeat steps 1) and 2)  $N$  times with changing of the testing period to send the testing packets. Here,  $N$  denotes the number of rounds. In each round, the testing period to send the testing packets is chosen at a random (as shown in Figure 7 (a)) or by a round-robin manner (as shown in Figure 7 (b)) to find non-congested situation(s).
- 4) Select the minimum  $M$  RTT values, and take the average of the selected RTT values. Here,  $M$  is not greater than  $PQN$ .
- 5) Classify the type of target network from the averaged RTT value.

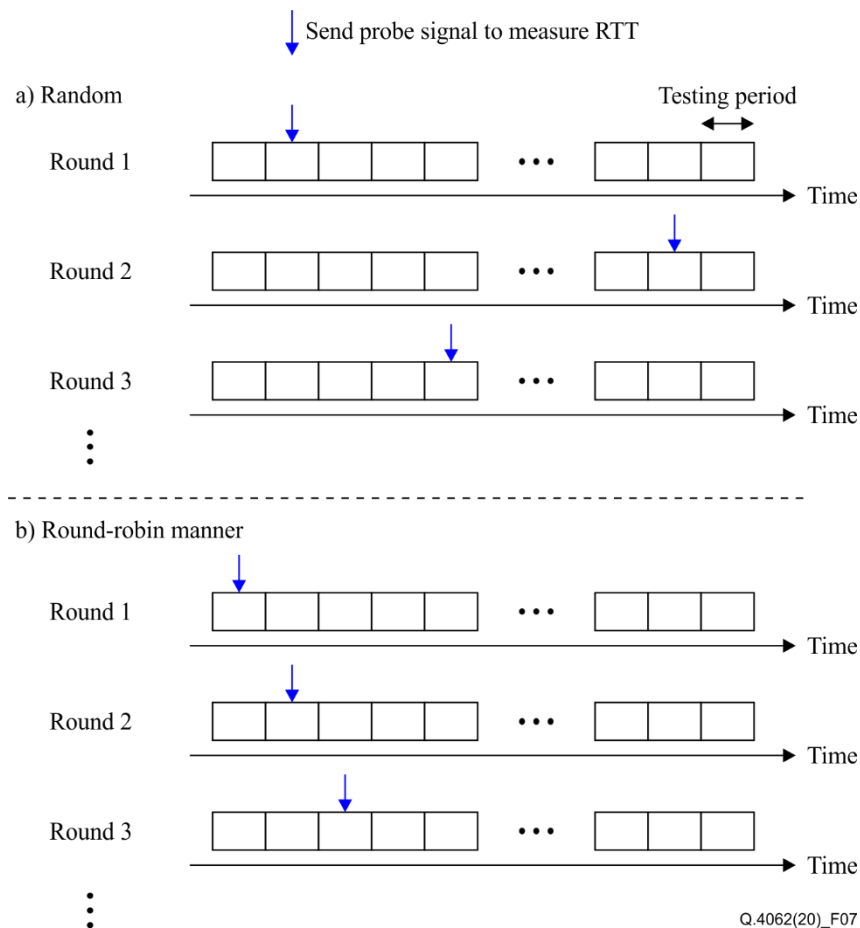


**Figure 5 – An example of testing network**



**Figure 6 – Procedure of the network type classification test**





**Figure 7 – An example of timing to send testing packet**

## 8 Consideration for test procedure

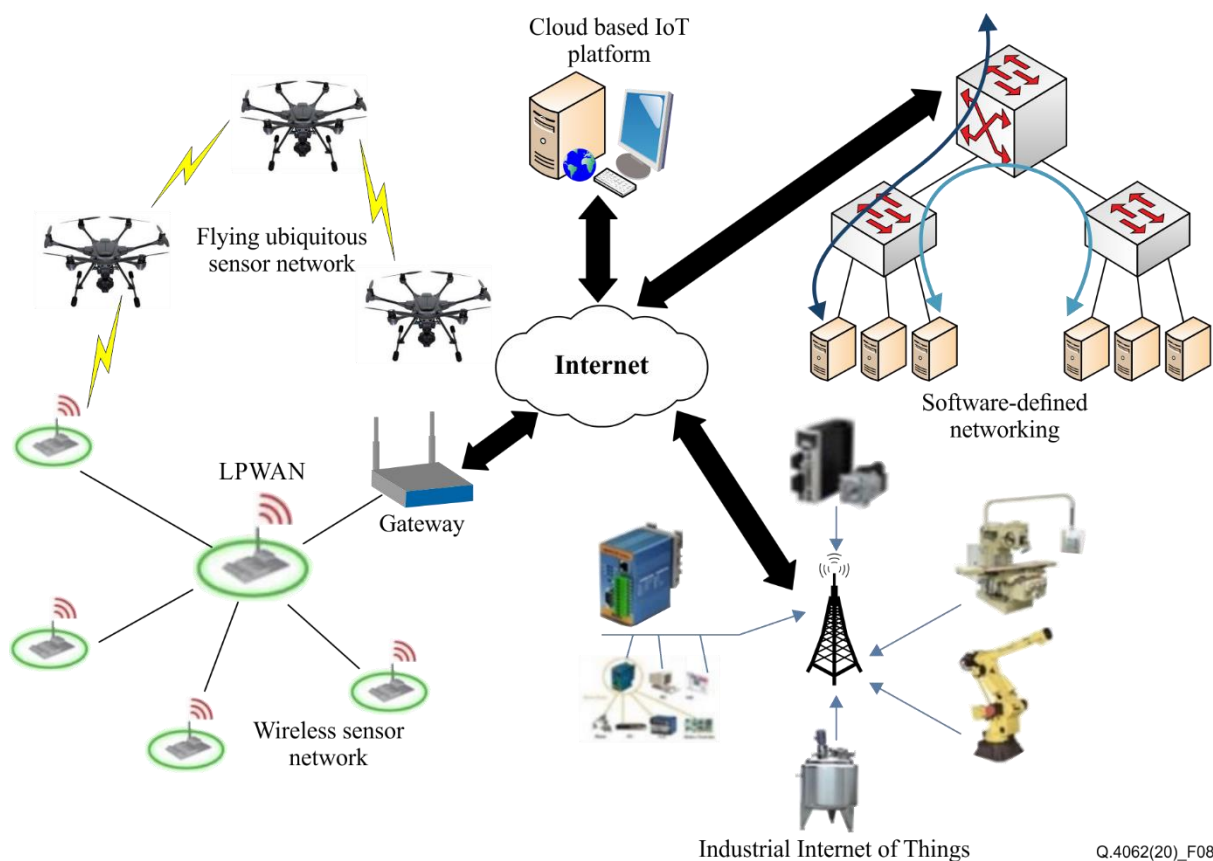
### 8.1 Testing network

#### 8.1.1 Existing network for testing

Testing of IoT devices and applications should be, in principle, conducted under the environment where they are used. With the existing network, any kinds of testing for IoT devices and applications can be performed and their results reflect the performance in the real situation where the IoT devices and applications are really used.

#### 8.1.2 Model networks for testing

If it is impossible to perform the full range of tests on the existing network due to either a technical or economic issue, an alternative approach is necessary to conduct testing. [ITU-T Q.3952] presents a model network shown in Figure 8 on which various IoT integration scenarios are implemented. [ITU-T Q.3952] defines the architecture of model networks to be used for IoT testing and presents individual segments of model network. The model network reproduces the architecture of both the designed network and the existing network, as well as their various combinations. Application of additional structural units such as traffic generators, delays and interference allows studying the network in any conceptual scenario.



**Figure 8 – Structure of model network for testing IoT devices and applications**

Figure adapted from [ITU-T Q.3952]

There are two types of model networks; dedicated networks and distributed networks. A dedicated network is a part of a public communication network that is not connected with other model networks in any way. It is used for general testing and compatibility and interoperability testing. It is formed by at least two nodes interconnected by data link layers. One of the nodes is the equipment being tested; the second is the model network itself. A distributed model network can be thought of as several model networks connected via public telecommunications networks or other types of networks. It is used for compatibility and interaction testing, as well as for checking quality of service (QoS) parameters, requirements for interaction with other technical requirements, as well as requirements for compliance with network and information security measures.

Model networks can be implemented in both physical and virtual forms. Their virtualization allows moving away from the real "hardware" to a huge range of emulators. This modification allows reducing both material costs of network equipment and network deployment cost because number of the network's structural elements are represented by a single software platform.

Segmentation is another tool for improving the model network's structure. It refers to the division of the network into independent segments. Each of these segments triggers the operation of a public telecommunications network segment designed with the use of simulated technologies and reflects its properties. Scenarios for the operation of individual segments can be prescribed in advance, which provides flexible configuration of the system.

## 8.2 Identification systems

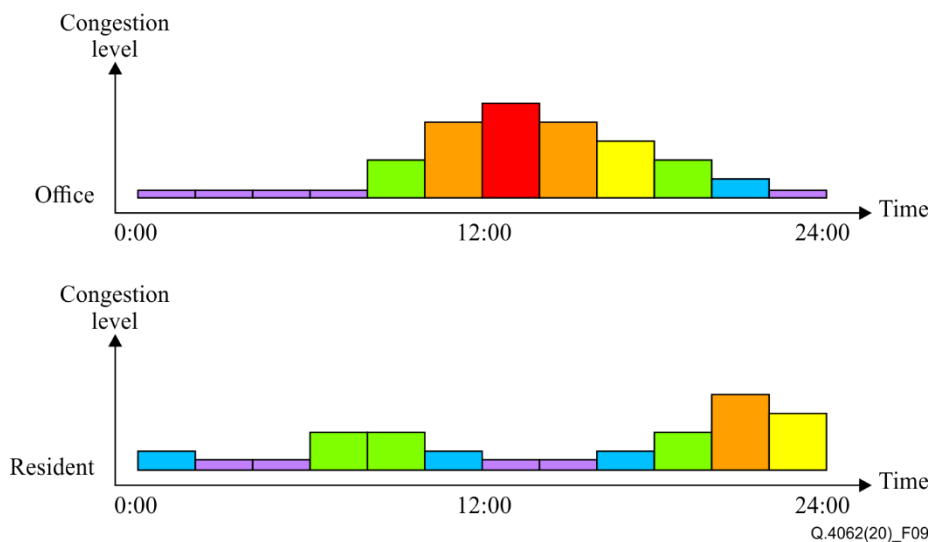
An identification system (IS) should be used for authentication in the early phase of IoT testing. Methods of testing IoT devices are different according to the types of identification system. Every IS should meet the following requirements:

- IS should unambiguously identify an IoT device (virtual or physical) by unique IoT identifier;
- IS should support special security system to communicate with IoT device (example: symmetric/asymmetric encryption, etc.);
- IS should meet requirements of data transmission services for every data type.

The selected IS should also meet the developer or standard requirements.

### 8.3 Test timing

Quality of services for wireless networks, such as cellular networks, wireless LAN (or radio LAN) and LPWAN, differ greatly according to network types, operation environment and time of day. The congestion level of wireless networks fluctuates within a day, and the pattern of congestion level depends on areas such as office and residential areas as shown in Figure 9. In order to efficiently conduct test of all devices including IoT devices connected to wireless access networks, it is required to consider test timing for each network type. In the case of Figure 9, it is better to select test timing for late at night or very early morning in office areas, since congestion levels are relatively low and it might be able to avoid severe network congestion due to increasing traffic volume by testing packets. In residential areas, it might be better to select test timing for around noon. An additional merit of this timing selection is that many terminals or devices can be tested at one time.



**Figure 9 – Examples of fluctuation of congestion level**

### 8.4 Grouping of target devices

Generally, in cases of IoT testing, there are an extraordinarily large number of devices to be tested, so the testing procedure may cause network congestion due to the injection of a large number of testing packets into the network. It is preferable to consider grouping target devices into multiple device groups and testing each group at different timing in order to reduce unnecessary amounts of bandwidth for the test traffic, and to avoid over-capacity for each of access technology's bandwidth. This will as a result contribute to shortening the total time for testing. There are several methods to form multiple device groups, for example, based on the identifier of devices or communication characteristics of target devices such as transmission speed (i.e., achievable data-rate) and delay performance as follows:

- Grouping based on device identifier

Each target device has its own device identifier. One of the simplest methods to form some (sub-)groups in the target network is to group based on device identifiers.

If the target network is an IP network, the target devices can be grouped by the following methods.

- Grouping based on IP address

Generally, the target network consists of several network segments with different ranges of IP addresses, so it may be easy to group target devices by the network segments.

For example, if the network is an IPv4 network consisting of IP addresses

AAA.BBB.CCC.0 ~ AAA.BBB.CCC.255,

the target devices can be grouped as

AAA.BBB.CCC.0 ~ AAA.BBB.CCC.127 and

AAA.BBB.CCC.128 ~ AAA.BBB.CCC.255.

(NOTE – AAA, BBB, and CCC are arbitrary values within a range of 0 ~ 255.)

Similarly, if the network is an IPv6 network consisting of IP addresses:

GGGG:HHHH:IIII:JJJ:0:0:0:0 ~ GGGG:HHHH:IIII:JJJ:ffff:ffff:ffff:ffff,

the target devices can be grouped as:

GGGG:HHHH:IIII:JJJ:0:0:0:0 ~ GGGG:HHHH:IIII:JJJ:7fff:ffff:ffff:ffff and

GGGG:HHHH:IIII:JJJ:8000:0:0:0 ~ GGGG:HHHH:IIII:JJJ:ffff:ffff:ffff:ffff.

(NOTE – GGGG, HHHH, IIII, and JJJ are arbitrary values within a range of 0 ~ ffff.)

- Grouping based on domain name

Generally, one or more ranges of IP addresses are assigned to a (sub-)domain name. Therefore, similar to grouping based on the IP address, the target devices can be grouped by their (sub-)domain name, as:

DOMAIN\_NAME1.com and DOMAIN\_NAME2.com,

or

SUBDOMAIN\_NAME1.MMM.com and SUBDOMAIN\_NAME2.MMM.com.

If the target network is a non-IP network, a MAC address and/or network-dependent device identifier can be used. Examples of the network-dependent device identifier are as follows:

- In LoRa network: DevAddr or DevEUI
- In Wi-SUN: Destination PAN ID and/or Destination Address
- In ZigBee: DstAddress
- Grouping based on communication characteristics

A target network usually has communication characteristics such as transmission speed and RTT value. These characteristics can be used for grouping the target devices so as to avoid congestion due to testing packets by assessing the injectable amount of testing packets. For example, the following communication characteristics can be used to group the target devices.

- Grouping based on transmission speed

The amount of testing packets that can be injected into the target network or that can be sent to the target devices at a time depends on the transmission speed of the target devices. If the target devices with different transmission speeds co-exist, they can be grouped into multiple device groups so that the devices in the same group have similar transmission speed. Grouping the target devices based on transmission speed and sending testing packets at a suitable packet rate would be an efficient way to conduct valid testing. Since the transmission speed varies over time, the instantaneous transmission speed should be used for grouping target devices in order to perform effective testing. The averaged transmission speed can also be used for grouping in the case of performance evaluation testing.

- Grouping based on RTT values

If all or some devices have extraordinarily large RTT values in some device groups, these devices may face network congestion. This is because each device has less chances to transmit the testing/response packets, and retransmission due to collision frequently occurs in a congested network. In such cases, in order to test the devices at a different timing from the original one, they can be set apart and put into new or other RTT-based device group(s).

## **8.5 Remote testing of IoT-devices**

### **8.5.1 IoT device detection and classification**

IoT device detection is one of the most important preliminary stages. Scanning open network ports might be used for detecting IoT devices.

To detect IoT devices without access to the data link via which the target IoT's traffic passes, existing methods of port scanning that scan devices connected to the Internet to find open network communication ports can be used. Since TCP and UDP transport layer protocols are most often used for networking in the Internet of things, the scan will also be performed to detect TCP and UDP ports.

There are several ways to perform TCP port scanning. The fastest and most easily implemented method uses the network functions of the operating system where the virtual scanning device is located. The virtual scanning device analyses the presence of open network ports on the network node being scanned by sequential port iteration. In the event that the analysed device has an open port, the operating system performs a three-step connection i.e. three-way handshake using synchronization (SYN), SYN-ACK, and acknowledgement (ACK) establishment procedure, after which the port is confirmed to be available for connection; the connection is then closed. To speed up the sequential port enumeration process, the parallel scanning method can be used, in which two or more ports of the analysed network node are scanned simultaneously from different ports of the virtual scanner.

UDP port scanning is a more time-consuming and resource-consuming process, as there is no concept of a communication session for the UDP protocol, and there is no guarantee that data will reach the port of the analysed device. Despite this, scanning is still possible, because most communication nodes respond to a packet incoming on a closed UDP port by sending an Internet control message protocol (ICMP) message stating that this network port is unavailable, and the absence of this message indicates that the UDP port is open. Since there is no guarantee of data delivery when transmitting data using the UDP protocol, it is necessary to send multiple requests to each port of the scanned device when scanning UDP ports by this method.

Most IoT use certain data transfer protocols, which, as mentioned above, occupy certain network ports. Examples include constrained application protocol (CoAP), message queue telemetry transport (MQTT), hypertext transfer protocol (HTTP), HTTP/2, real time messaging protocol (RTMP), real time streaming protocol (RTSP), etc. The non-profit organization "Internet Assigned Numbers Authority" (hereinafter "IANA") determines the fixed port data numbers which are used for certain purposes and establishes a correspondence between them, and the application-layer protocols applied. By comparing the list of detected network ports on the scanned device and the IANA database, the application data transfer protocol used on this network port can be determined.

Other ports not listed in the IANA database that use the TCP protocol for data transfer typically send a response to any request. The response received from the device allows determining of which application-layer protocol the port is using.

To determine the type of IoT devices, the data field of the response received for the request is analysed. For example, responses to HTTP requests may contain information about the device, such as the name of the operating system or manufacturer.

An example is the following HTTP header:

```
HTTP/1.1 200 OK;  
Date: Wed, 26 Jan 2016 11:06:42 GMT;  
Server: Linux/2.x UPnP/1.0 Avtech/1.0;  
Connection: close;  
Last-Modified: Wed, 08 Jan 2014 09:36:39 GMT;  
Content-Type: text/html.
```

The "Server" header contains the following useful information:

- Linux/2.x – the core of the operating system used by the device.
- UPnP – technology, which includes a set of network protocols, designed to connect devices in a home or corporate environment.
- Avtech – video camera manufacturer.

Based on the foregoing, it can be estimated that the device from which the response was received is a video camera.

Determining the type of device is based on the protocols that it uses, as well as on the basis of responses that were received from open ports. For example, there is a device with open port 15757, which uses the HTTP protocol, sending a response with the following headers:

```
HTTP/1.1 401 N/A  
Router Webservers  
close  
Basic realm = "TP-LINK Wireless N Router WR841N"  
5.text/html
```

From the information on the open port and header information above, the type of device can be estimated as a wireless router.

Many devices that are used on the Internet of things (especially serial devices such as video cameras, home appliances, etc.) have a web interface. Analysing information from the web pages of such devices can help determine its type. It is worth noting that this method of determining the type of Internet of things is becoming more relevant due to the emergence of the Web of things concept, which suggests that every Internet of things device should have a web interaction interface.

However, the absence of a web page on the device or the absence of open network ports operating via HTTP, HTTPS, and HTTP/2 protocols does not mean that the device is not in an IoT. There are many other IoT interaction protocols that do not support the Web of things concept but are still connected to the Internet. In this regard, the scanning device must have response formats for each of the popular protocols for the Internet of things in its database to be able to analyse the information contained in the data field of the received packets. An example of a searching system structure is shown in Figure II.1

Analysis of all available information about the device, using certain algorithms, allows it to be determined whether (or not) it is an IoT device, its type, and also to get access to it for interaction. A more detailed description is presented in Appendix II.

### 8.5.2 Remote testing of IoT-devices

A packet analysis system is used for sniffing traffic passing through the local network and to determine devices, with the aid of unique parameters that characterize the IoT device (specialized protocols, average size of packets from the device, packet sending frequency, antipersistent or self-similar character of the traffic, the set of network ports used, information on operating servers on the device to which the packet belongs, and so on) on the basis of information derived from the information packets. In this way, thanks to the information in question, the packet analysis system can determine the IoT devices operating in the local network.

Based on the model network it is advisable to develop a technique for remote testing of the IoT devices. It is proposed to consider the IoT devices (for a known IP address) as a "black box". In the process of testing special requests are sent to the IoT device with remote server. Requests are delivered to the input of the network interface. After receiving the requests, the IoT devices send response service packets from the output of the network interface; these packets are received by the remote cloud server for further analysis. For IoT devices testing it is necessary to have an IP address or additional identifier of one of the IoT device interfaces and a unique identifier that is stored in the database of an IoT identification server.

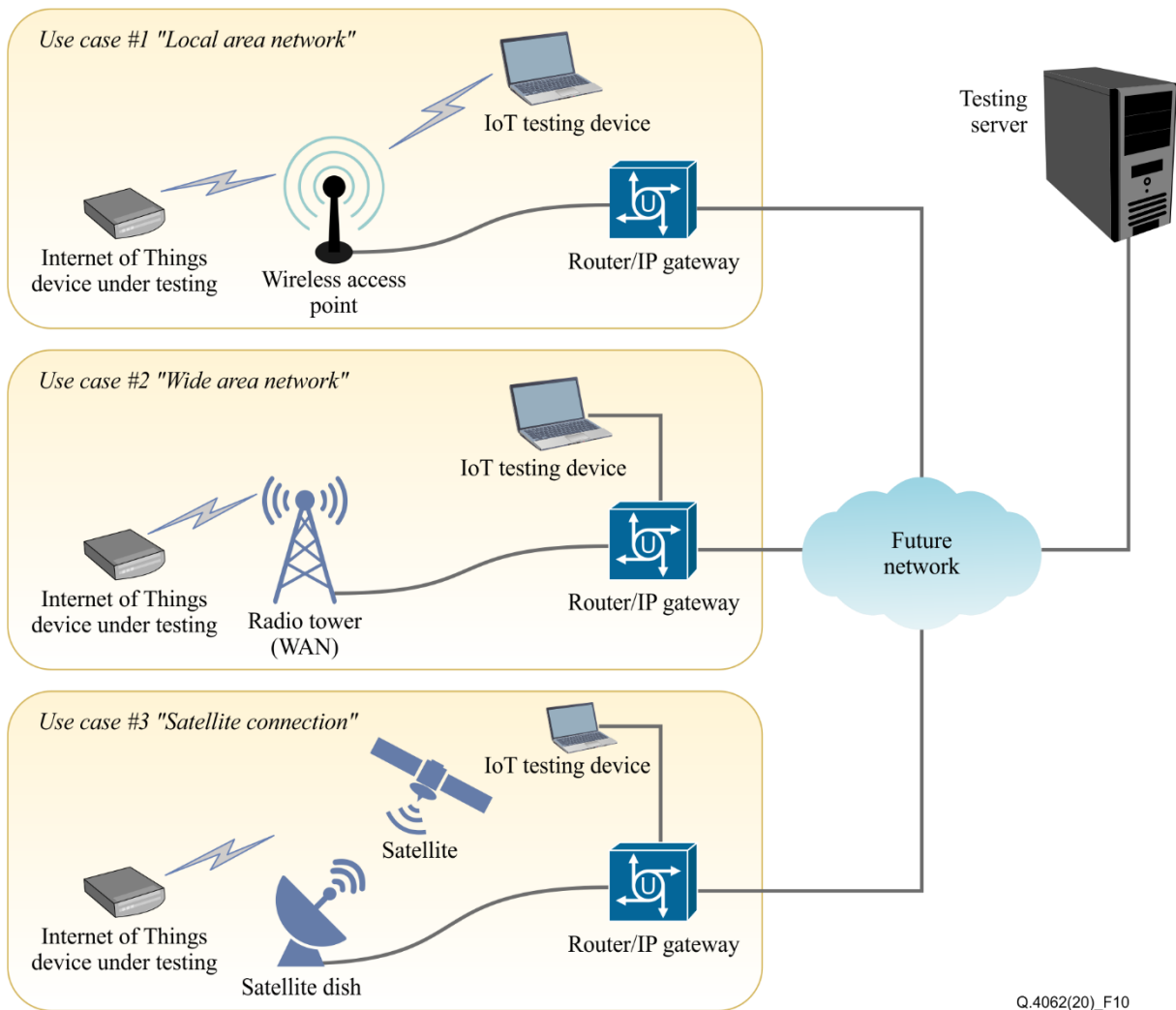
The cloud server, that receives packets from the IoT device, identifies the format of packets that are processed and sent by IoT devices. The format of packets and their structure are determined using the method of deep analysis of the traffic. Using the IoT device encryption, the key for arranging interaction with the IoT device must first be obtained.

Owing to the availability of data on the format of packets from IoT devices, it is possible determine the type of IoT device:

- actuator-type;
- sensor-type [ITU-T Y.2069];
- mixed type (both the actuator and the sensor);
- devices for transfer of multimedia information [ITU-T Y.2069].

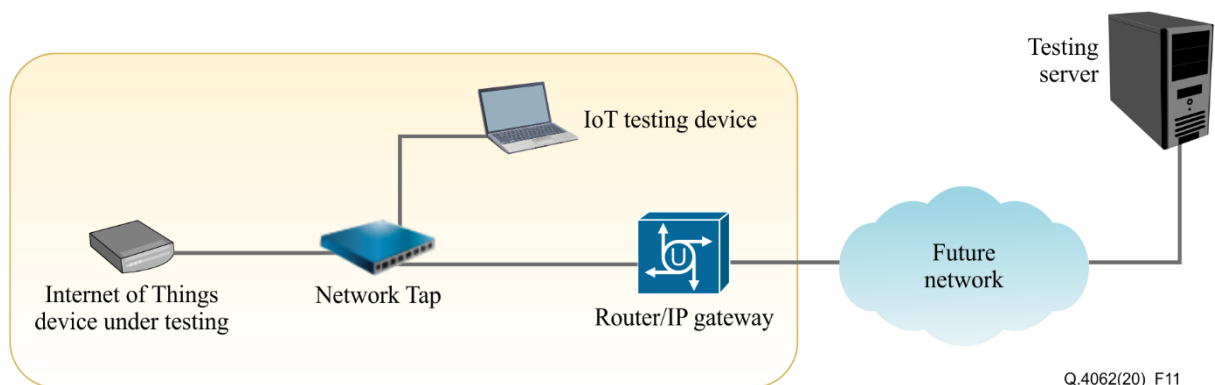
The testing server receives information on the device to be tested from the testing device which, with the aid of the packet analysis system, finds the devices in the local network.

A model of remote testing of the IoT devices is shown in Figure 10 and Figure 11. The process of discovering IoT devices is shown in Figure 12.



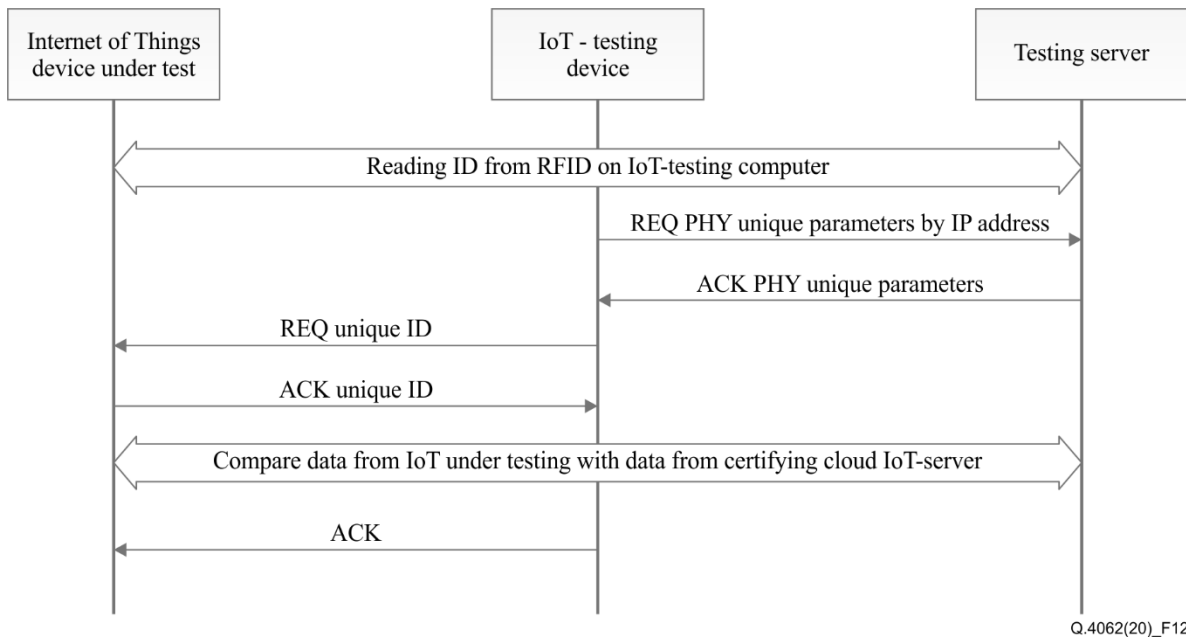
**Figure 10 – A remote testing example of the IoT-devices for wireless technologies**

The method is used for remote testing of IoT devices with the aid of a testing device located within the local IPv4/IPv6 network. If the mobile device (such as a vehicle, satellite, etc.) operates on the basis of the IPv4/IPv6 network and the testing device has access to the communication channel connecting the device being tested to the Internet, this testing method can be used.



**Figure 11 – A remote testing example of the IoT devices for wired technologies**





**Figure 12 – Process of discovering IoT devices**

The process of discovering devices is described in Figure 12.

During the process of discovering devices, the IoT testing device determines the type of device using a packet analysis system, based on traffic data from IoT device. The following settings can be used to analyse: protocol types (CoAP, MQTT, etc.), average size of packet, packet transmission rate, traffic profile, etc. After that the IoT device is initialized on the test server. The steps are as follows:

- 1) Transfer of unique identifiers of the IoT device which is under test (ID, unique identifier that is stored in the database of IoT identification server).
- 2) The next step is the IoT testing device requests unique physical parameters based on the transferred IP address of the IoT device which is under test from the testing server.
- 3) After that an exchange of service messages takes place between IoT device which is under test and the IoT testing device to confirm the ID of device.
- 4) The next step is testing, after which the packets are analysed and compared with the data from a certifying cloud IoT server. When testing is completed, the testing device sends a confirmation response to the IoT device which is under test.

Determination of the packet format and the IoT device type enables the packets to be generated, and thus checks of vulnerabilities and characteristics of the IoT device are made, for example, as follows:

- checking the availability of standard ports for interfacing with a computing device (80, 8080, 21, 22, 23, and so on.);
- checking the operation of the IoT device when it is prompted, or it sends incorrect values;
- checking the vulnerability of the IoT devices to DDoS-attacks.

## Annex A

### Testing specifications

(This annex forms an integral part of this Recommendation.)

Reference information about types of testing which include the committed information rate (CIR) test, and excess information rate (EIR), and test of traffic policing is available in [b-ITU-T Q.4060], [ITU-T Q.3952], [b-ITU-T Y.4500.13] and [ITU-T Y.4500.15]. Some of the references are based on example networks, such as Bluetooth low energy (BLE), ZigBee and Thread. Also, there are compulsory types of functional testing, objectives and conditions of their conduction, schemes of connection (configuration of stand), and the requirements for the report for each of the given tests. There are also examples of tables and formulas for calculations, which should also be presented in the report on the relevant type of testing.

Ethernet testing (standards 10Base-T, 100Base-T, 1000Base-T, 10GBase-T)

**Table A.1 – Test No. 01**

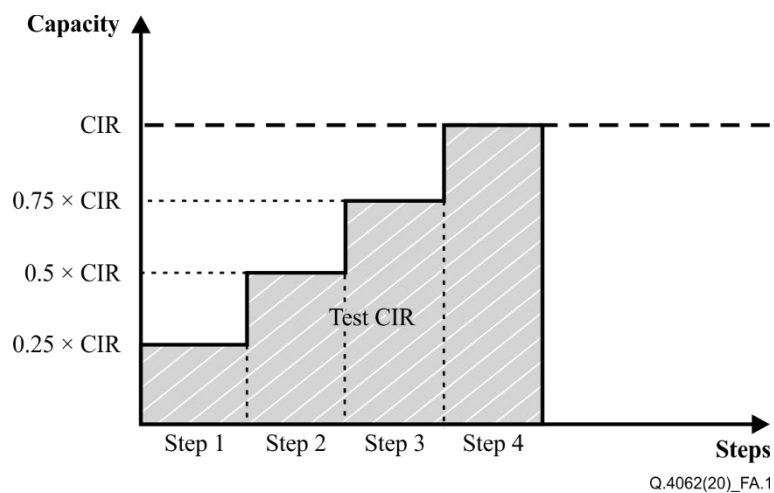
Test number	No. 01
Test name	The network configuration testing
Testing layer	Network
Type of test	Functionality
Status	Optional
Test goal	The services testing on the service level agreement (SLA) conformity
Configuration	
Testing procedure	1) Test CIR. 2) Test EIR. 3) Test traffic policing. * <i>The time for each test is not more than 60s.</i>
Expected results	SLA conformity

#### Test device

IoT device with Ethernet interface.

#### Test CIR

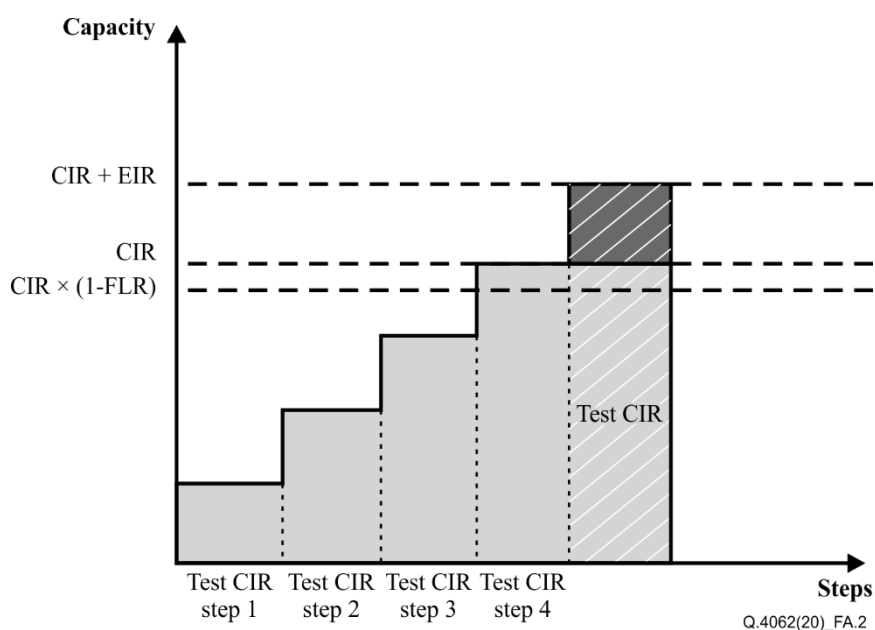
Test CIR is the test for definition of guaranteed capacity. Test can be conducted for load 25%, 50%, 75%, 100% of guaranteed capacity. The schedule of the CIR test is shown on the Figure A.1.



**Figure A.1 – The schedule of the CIR test**

### Test EIR

Test EIR is the test for verification that the capacity to each service will not more than the permissible value when the load value is the CIR+EIR. This test is conducted in the conditions from guaranteed capacity CIR up to maximum of the non-guaranteed capacity EIR (best effort conditions). The schedule of the EIR test is shown on the Figure A.2.



**Figure A.2 – The schedule of the EIR test**

### Test of traffic policing

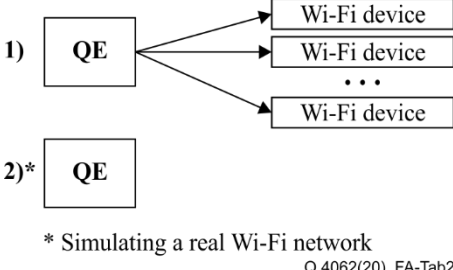
Test of traffic policing is conducted for verification that the network will limited the capacity to separate service if the real traffic of this service will be more than authorized traffic.

### Report of testing

The guaranteed and non-guaranteed capacity which was observed during testing should be marked in the report.

## IEEE 802.11 (WLAN) standard testing

**Table A.2 – Test No. 02**

Test number	No. 02
Test name	The complex test of WLAN with using precision measuring equipment which allows simulation of the real network performance simulation.
Testing layer	Network
Type of test	Functionality
Status	Mandatory
Test goal	The definition of the value of the main WLAN networks parameters (SSID, RSSI, frequency, encryption type, BER, PER, channel utilization and so on.)
Configuration	 <p>1) QE → Wi-Fi device          Wi-Fi device          ...          Wi-Fi device</p> <p>2)* QE</p> <p>* Simulating a real Wi-Fi network          Q.4062(20)_FA-Tab2</p>
Test procedure	<ol style="list-style-type: none"> <li>1) Set up measuring equipment in accordance with its technical terms and conditions.</li> <li>2) Detect the WLANs which should be testing.</li> <li>3) Define the values of the main parameters.</li> </ol> <p>* <i>The measurement equipment can allow simulate the wireless channel with different errors.</i></p>
Expected results	The set of WLAN parameter values.

### Test device

IoT devices with WLAN interfaces, WLAN AP.

### Testing procedure

The following metrics should be tested:

- mobility;
- interoperability with WLAN equipment;
- the main parameters that estimate the WLAN throughput and QoS and QoE;
- scalability.

At first, the tests should be made for main parameters without traffic. The load testing should be made on the next step when the real networks conditions will be simulated.

The precision measuring equipment, such as oscilloscope and spectrum analyser, can be used for testing.

### Report

The report about testing performed in the table which includes the following parameters:

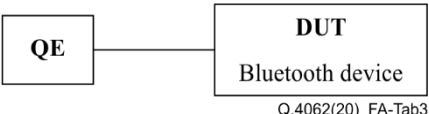
- WLAN network identifier (SSID);
- the received signal strength (RSSI);
- frequency;

- encryption type (WEP, WPA, WPA2);
- BER (permissible value  $10^{-5}$  for WLAN) [b-Heegard];
- PER (permissible value  $10^{-2}$  for WLAN) [b-Heegard];
- channel utilization (in %).

\* The table can be extended if testing scenario requests to add some special parameters.

### IEEE 802.15.1 (Bluetooth, Bluetooth LE) standards testing

**Table A.3 – Test No. 03**

Test number	No. 03
Test name	BER definition
Testing layer	Network
Type of tests	Functionality
Status	Mandatory
Tests goal	BER definition on the IEEE 802.15.1 standard base
Configuration	
Testing procedure	<ol style="list-style-type: none"> <li>1) Connect QE and Bluetooth device.</li> <li>2) Generate the pseudo-random bit sequence at the Bluetooth device.</li> <li>3) Define the value of BER.</li> </ol>
Expected results	Value of BER

#### Test device

IoT device with Bluetooth interface.

#### Report

The test results should be reported in Table A.3a.

**Table A.3a – Test results**

No.	Number of the bit which were send	BER	Check sum (CRC)	Time
1				
2				
...				
n				

## IEEE 802.15.4 (ZigBee, 6LoWPAN) standard testing

**Table A.4 – Test No. 04**

<b>Test number</b>	<b>No. 04</b>
Test name	PER definition
Testing layer	Network
Type of tests	Functionality
Status	Mandatory
Tests goal	PER definition on the base of standard IEEE 802.15.4
Configuration	<pre> graph LR     subgraph QE [QE]         ZGW[ZigBee GW]     end     subgraph DUT [DUT]         ZEP[ZigBee end point]     end     ZGW --- ZEP             </pre> <p style="text-align: right; font-size: small;">Q.4062(20)_FA-Tab5</p>
Testing procedure	<ol style="list-style-type: none"> <li>1) Connect GW and ZigBee End Point.</li> <li>2) Generate at ZigBee-device specified number of packets.</li> <li>3) Define PER.</li> </ol>
Expected results	PER values

### Test device

IoT device with ZigBee interface.

### Testing procedure

ZigBee End Point device and ZigBee gateway are used for testing. The value of RSSI is observed on the ZigBee micro controller and PER is calculated. A generic purpose sniffer software can be used for detailed packets analysis.

### Report

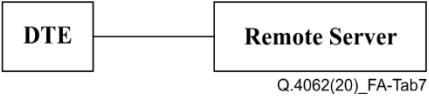
The test results should be reported in Table A.4a.

**Table A.4a – Test results**

No.	RSSI	Number of packets which were send	PER	Probability of packets delivery	Time
1					
2					
...					
n					

## Mobile technologies testing

**Table A.5 – Test No. 05**

Test number	No. 05
Test name	Mean value of data transmission rate
Testing layer	Network
Type of tests	Functionality
Status	Mandatory
Tests goal	Mean value of data transmission rate in case of web-content download
Configuration	
Testing procedure	<ol style="list-style-type: none"> <li>1) Connect by TCP/IP terminal and remote server using protocol HTTP.</li> <li>2) Download the file of specified size.</li> <li>3) Define time that was needed for full file download.</li> <li>4) Define rate of data transmission.</li> <li>5) Repeat the procedure more times.</li> </ol>
Expected results	Mean value of data transmission rate

### Test device

IoT device with modules GPRS, EDGE, 3G/4G (LTE).

### Testing procedure

The mean rate of data transmission is calculated by:

$$\text{Kbit/s} = \frac{\sum_{i=1}^n V}{n}, \text{ where}$$

$V$  – rate of data transmission using protocol HTTP from remote server to terminal,

$n$  – number of tests.

$$V = \frac{P}{t_{\text{initial}} - t_{\text{final}}}, \text{ where}$$

$P$  – test file size,

$t_{\text{initial}}$  – time of the start of transmission,

$t_{\text{final}}$  – time of the end of transmission.

### Report

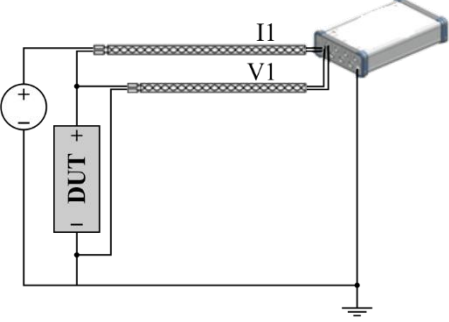
The test results should be reported in Table A.5a.

**Table A.5a – Test results**

No.	Test file size	Time of start	Time of end	Rate
1				
2				
...				
n				
<b>Mean rate</b> ____ <b>kbit/s</b>				

## Battery life testing

**Table A.6 – Test No. 06**

Test number	No. 06
Test name	Battery life
Testing layer	Physical
Type of tests	Functionality
Status	Mandatory
Tests goal	IoT Power Consumption Measurement
Configuration	 <p style="text-align: right; font-size: small;">Q.4062(20)_FA-Tab9</p>
Testing procedure	<ol style="list-style-type: none"> <li>1) Connect LoRa and gateway device.</li> <li>2) Connect power nodes from DUT to multichannel probe with oscilloscope.</li> <li>3) Define the power consumption.</li> </ol> <p>Use oscilloscope with 3 enabled traces and multichannel probe: the 1st plot for both the supply voltage plot and the current drain over time. In this plot, the current drain during packet transmission can be seen. The second plot shows the total power consumption over time. Using the area (integral) measurement function on the math channel with gating enabled allows to measure the energy consumed during one transmit frame. Measure sleep mode energy consumption using Markers. (Table A.6a)</p> <p>Battery life for NB-IoT, Bluetooth, LTE, 3GPP communication technologies can be measured automatically using communication tester and software package. (Table A.6b)</p>
Expected results	Battery life in hours

### Test device

IoT LoRa device.

### Report

The test results should be reported in Table A.6a and Table A.6b.



**Table A.6a – Test results**

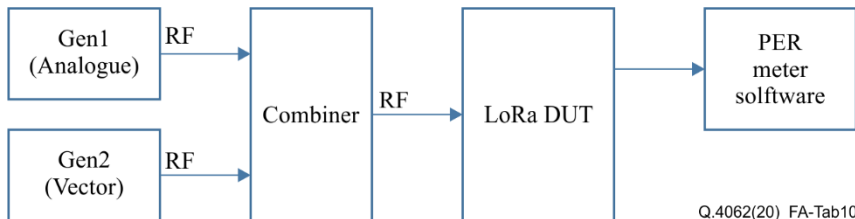
No.	Transmit frame length, ns	Supply voltage, V	Current drain, mA	Sleep mode current drain, mA	Consumed energy, W*s
1					
2					
...					
n					
Current battery life ____ h					
Capacity of battery ____ mAh					

**Table A.6b – Test results**

No.	Capacity of battery	Voltage avg	Current avg	Power avg	Battery life
1					
2					
...					
n					
Current battery life ____ h					

**Receiver testing**

**Table A.7 – Test No. 7**

Test number	No. 7
Test name	Receiver test
Testing layer	Physical
Type of tests	Functionality
Status	Mandatory
Tests goal	Rx sensitivity measurement
Configuration	 <p style="text-align: right; font-size: small;">Q.4062(20)_FA-Tab10</p>
Testing procedure	<ol style="list-style-type: none"> <li>1) Connect vector and analogue generator to power combiner.</li> <li>2) Connect LoRa testset device to power combiner.</li> <li>3) Run PER testset measurement software.</li> </ol> <p>The test set-up consists of two signal generators, the signals from which are fed to the DUT as a sum signal via a power combiner. Generator #1 generates an unmodulated, sinewave interference signal which is transmitted either with a spacing of 200 kHz relative to the wanted signal (adjacent channel blocking) or at the same frequency as the wanted signal (on-channel blocking). Generator #2 supplies the LoRa wanted signal, which is generated via LoRa ARB waveform files. The packet error rate value is measured using the LoRa test tool.</p> <p>For the Rx sensitivity test, load a set of LoRa ARB waveform files in the Vector</p>

**Table A.7 – Test No. 7**

<b>Test number</b>	<b>No. 7</b>
	Signal Generator for testing the sensitivity of the receiver. The set of files contains waveforms with various signal bandwidths and spreading factors. A RF carrier signal is modulated using these baseband ARB files, which are loaded in the vector signal generator, and fed to the receiver in the appropriate frequency range. While the signal power is being reduced, the LoRa test tool is used to read out and monitor the packet error rate (PER). The receiver sensitivity up to which no bit errors or very few bit errors occur depends on the used spreading factor and ranges from approximately –117 dBm to –137 dBm. For testing standalone modules without test tool provided by manufacturer see software Wireshark method.
Expected results	Rx sensitivity level, blocking level

**Test device**

IoT LoRa testset.

The test results should be reported in Table A.7a and Table A.7b .

**Table A.7a – Receiver test results**

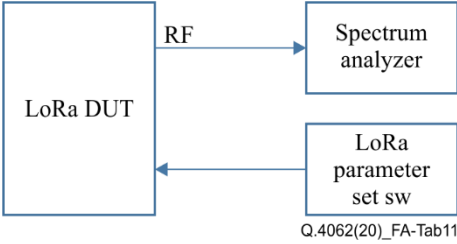
No.	Frequency, MHz	Level, dBm	Modulation, SF type	PER, %
1				
2				
...				
n				
<b>Current PER, %</b>				

**Table A.7b – Blocking test results**

No.	Frequency gen1, MHz	Level gen1, dBm	Modulation gen1, SF type	Frequency gen2, MHz	Level gen2, dBm	PER, %
1						
2						
...						
n						
<b>Current PER, %</b>						

## 6dB bandwidth testing

**Table A.8 – Test No. 8**

<b>Test number</b>	<b>No. 8</b>
Test name	6 dB Bandwidth
Testing layer	Physical
Type of tests	Functionality
Status	Mandatory
Tests goal	Transmitter test
Configuration	 <p style="text-align: right; font-size: small;">Q.4062(20)_FA-Tab11</p>
Testing procedure	<p>1) Connect LoRa testset device to spectrum analyser.                  2) Run LoRa parameter set software.</p> <p>Using the spectrum analyser with Central Frequency = current DUT frequency (902 MHz to 928 MHz), span = 1.5 MHz, RBW = 100 kHz, VBW = 300 kHz, choose trace function with Positive Peak detector and Max Hold. Set reference level such that the maximum value of the signal is below the reference level. Use marker function for calculating "n dB down" for 6 dB Value. Following condition must be fulfilled: n dB down BW <math>\geq</math> 500 kHz.</p>
Expected results	6 dB Bandwidth

### Test device

IoT LoRa testset.

### Report

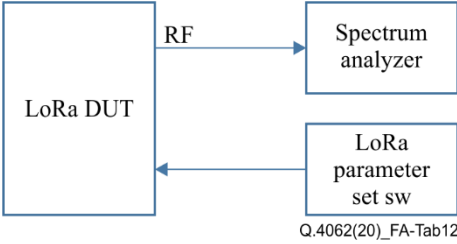
The test results should be reported in Table A.8a.

**Table A.8a – 6 dB bandwidth test results**

No	Frequency, MHz	Spreading Factor	Bandwidth, kHz	n dB down BW, kHz
1	915	SF7		
2				
...				
n				
Span = ____ MHz, RBW = ____ kHz, VBW = ____ kHz (3 x RBW)				
6 dB Bandwidth = ____ kHz				

## Occupied bandwidth testing

**Table A.9 – Test No. 9**

Test number	No. 9
Test name	Occupied bandwidth
Testing layer	Physical
Type of tests	Functionality
Status	Mandatory
Tests goal	Transmitter test
Configuration	 <p style="text-align: right; font-size: small;">Q.4062(20)_FA-Tab12</p>
Testing procedure	<p>1) Connect LoRa testset device to spectrum analyser.                  2) Run LoRa parameter set software.</p> <p>According to FCC 15.247, the output power of a transmitter in the frequency range 902 MHz to 928 MHz must not exceed 1 W or 30 dBm. The total output power and the band power respectively are determined by integrating the power over the signal bandwidth. The signal bandwidth corresponds to the occupied bandwidth (OBW). The OBW is the bandwidth in which 99% of the signal power is contained.</p> <p>Using the spectrum analyser with Central Frequency = current DUT frequency (902 MHz to 928 MHz), span = 2 MHz, RBW = 30 kHz, VBW = 100 kHz, sweep time = 2ms, choose trace function with Positive Peak detector and Max Hold. Set Reference Level such that the maximum value of the signal is at least <math>10\log(\text{OBW}/\text{RBW})</math> below the reference level. Use measurement function and marker for calculating "Occ BW, kHz" Value.</p>
Expected results	Occupied bandwidth

### Test device

IoT LoRa testset.

### Report

The test results should be reported in Table A.9a.

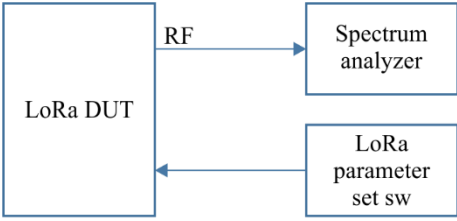
**Table A.9a – Occupied bandwidth test results**

No.	Frequency, MHz	Spreading factor	Bandwidth, kHz	Span, MHz	OBW, kHz
1	915	SF7		2	
2					
...					
n					

Span = \_\_\_\_ MHz (1.5 to 5 x OBW), Sweep = 2 ms,  
 RBW = \_\_\_\_ kHz (1% to 5% of the OBW), VBW = \_\_\_\_ kHz (3 x RBW)  
 Occ BW = \_\_\_\_ kHz

**Emission output power testing**

**Table A.10 – Test No. 10**

Test number	No. 10
Test name	Emission output power
Testing layer	Physical
Type of tests	Functionality
Status	Mandatory
Tests goal	Transmitter test
Configuration	 <p style="text-align: right; font-size: small;">Q.4062(20)_FA-Tab13</p>
Testing procedure	<p>1) Connect LoRa testset device to spectrum analyser.            2) Run LoRa parameter set software.</p> <p>Use Average mode on trace1 with RMS detector type. Choose average count at least 100. Set sweep Time to 50 ms. Set marker 1 for the transmit frequency of the DUT. Use the band power function with Span value = OBW value from previous measurements. Power spectral density. Perform single measurement on the spectrum analyser; wait until the number of averaging operations have been performed. The result of the measurement is Band Power in dBm.            The following conditions must be fulfilled: Band power ≤ 30 dBm.</p>
Expected results	Emission output power

**Test device**

IoT LoRa testset.

**Report**

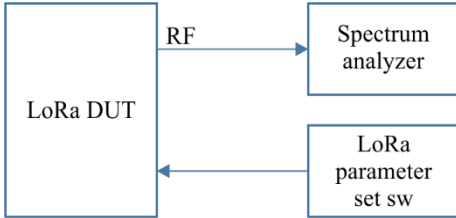
The test results should be reported in Table A.10a.

**Table A.10a – Emission output power test results**

No.	Frequency, MHz	Spreading factor	Sweep time, ms	OBW, kHz	Band power, dBm
1	915	SF12	50		
2	915	SF7	50		
...					
n					
<b>Average count = 100</b>					

**Power spectral density testing**

**Table A.11 – Test No. 11**

Test number	No. 11
Test name	Power spectral density
Testing layer	Physical
Type of tests	Functionality
Status	Mandatory
Tests goal	Transmitter test
Configuration	 <p style="text-align: right; font-size: small;">Q.4062(20)_FA-Tab14</p>
Testing procedure	<p>1) Connect LoRa testset device to spectrum analyser.                  2) Run LoRa parameter set software.</p> <p>Using the spectrum analyser with Central Frequency = current DUT frequency (902 MHz to 928 MHz), span = 1.5xOBW value from Table A.10, RBW = 3 kHz, VBW = 10 kHz, sweep time = 10 ms for SF7 or 500 ms for SF12, use Average mode on trace1 with RMS detector type. Choose average count at least 100. Set Auto Reference Level. Perform single measurement on the spectrum analyser; wait until the number of averaging operations have been performed. Use Marker peak measurement function. The result of the measurement is Power marker M1 in dBm. The following conditions must be fulfilled: Power marker M1 ≤ 8 dBm.</p>
Expected results	Power spectral density

**Test device**

IoT LoRa testset.

**Report**

The test results should be reported in Table A.11a.

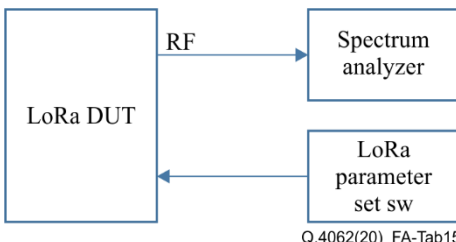
**Table A.11a – Power spectral density test results**

No.	Frequency, MHz	Spreading factor	OBW, kHz (from Table A.11)	PSD, dBm
1	915	SF7		
2	915	SF12		
...				
n				

Span = \_\_\_\_ MHz (1.5 x OBW), Sweep = 10ms for SF7, Sweep = 500ms for SF12,  
 RBW = 3 kHz, VBW = 10 kHz (3 x RBW)

**Emission in non-restricted bands testing**

**Table A.12 – Test No. 12**

Test number	No. 12
Test name	Emissions in non-restricted bands
Testing layer	Physical
Type of tests	Functionality
Status	Mandatory
Tests goal	Transmitter test
Configuration	 <p style="text-align: right; font-size: small;">Q.4062(20)_FA-Tab15</p>
Testing procedure	<p>1) Connect LoRa testset device to spectrum analyser.                  2) Run LoRa parameter set software.</p> <p>Measure maximum radiated power (REFlo) for lowest channel frequency for a 500 kHz wide LoRa signal (uplink), SF7 (903 MHz).                  Measure maximum radiated power (REFhi) for highest channel frequency for a 500 kHz wide LoRa signal (uplink), SF7 (914,2 MHz).                  Set the spectrum analyser on central frequency <math>F_{tx} = 903</math> MHz, according to lowest channel center frequency for a 500 kHz wide LoRa signal (uplink), SF7.                  Choose span at least 1.5x(n dB down BW) from Table A.9 (1.5 MHz),                  RBW = 100 kHz, VBW = (3xRBW) kHz, auto sweep. Use trace function with Positive Peak detector and Max Hold. Adjust the reference level accordingly to the maximum signal level with amplitude function. Use marker peak function for REFlo calculation.                  Use trace function with positive peak detector and Max Hold. Adjust the reference level accordingly to the maximum signal level with amplitude function. Use marker peak function for REFhi calculation. Use marker-to-peak search measurement function, define a range (upper and lower edges of the ISM band), turn on the Auto Max Peak function. The marker will indicate the highest level value M1 within the frequency range to be analysed.                  The following condition must be fulfilled: <math>REFhi - M1 \geq 30</math> dB.                  Using the marker-to-peak search function measure maximum radiated power at upper and lower edge of industrial scientific and medical (ISM) band.</p>

**Table A.12 – Test No. 12**

<b>Test number</b>	<b>No. 12</b>
	The following condition must be fulfilled: $REF_{lo-M1} \geq 30$ dB.
Expected results	Emissions in non-restricted bands

**Test device**

IoT LoRa testset.

**Report**

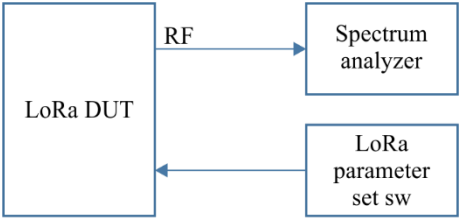
The test results should be reported in Table A.12a.

**Table A.12a – Emissions in non-restricted bands test results**

No.	Frequency, MHz	Spreading factor	REF <sub>lo</sub> , dBm	REF <sub>hi</sub> , dBm	REF <sub>hi-M1</sub> , dB	REF <sub>lo-M1</sub> , dB
1	903	SF7				
2	915	SF12				
...						
n						

**20dB bandwidth testing**

**Table A.13 – Test No. 13**

<b>Test number</b>	<b>No. 13</b>
Test name	20 dB Bandwidth
Testing layer	Physical
Type of tests	Functionality
Status	Mandatory
Tests goal	Transmitter test
Configuration	 <p style="text-align: right; font-size: small;">Q.4062(20)_FA-Tab16</p>



**Table A.13 – Test No. 13**

Test number	No. 13
Testing procedure	1) Connect LoRa testset device to spectrum analyser. 2) Run LoRa parameter set software. DUT settings = LoRa, 915 MHz, SF7, 125 kHz. Using the spectrum analyser with Central Frequency = 915 MHz. sweep = 5 ms, span = at least 2 to 3 times the 20 dB bandwidth, RBW = 1% of the 20 dB bandwidth, VBW = 3 x RBW, choose trace function with Positive Peak detector and Max Hold. Set auto reference level. Use marker function for calculating "n dB down" for 20 dB Value. If necessary, use the measured value for the 20 dB bandwidth (n dB down BW) to adjust the span and resolution bandwidth in line with the conditions named above. The following condition must be fulfilled: $n \text{ dB down BW} \leq 500 \text{ kHz}$ .
Expected results	20 dB Bandwidth

**Test device**

IoT LoRa testset.

**Report**

The test results should be reported in Table A.13a.

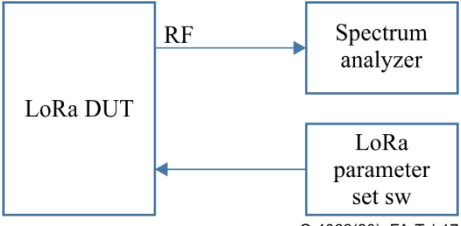
**Table A.13a – 20 dB bandwidth test results**

No.	Frequency, MHz	Spreading factor	Bandwidth, kHz	n dB down BW, kHz
1	915	SF7	125	
2	915	SF7	250	
...				
n				
Span = ____ MHz, RBW = ____ kHz, VBW = ____ kHz (3 x RBW) 20 dB Bandwidth = ____ kHz				

**Power spectral density testing****Table A.14 – Test No. 14**

Test number	No. 14
Test name	Power spectral density (hybrid mode)
Testing layer	Physical
Type of tests	Functionality
Status	Mandatory
Tests goal	Transmitter test

**Table A.14 – Test No. 14**

Test number	No. 14
Configuration	 <p style="text-align: right; font-size: small;">Q.4062(20)_FA-Tab17</p>
Testing procedure	<p>1) Connect LoRa testset device to spectrum analyser.                  2) Run LoRa parameter set software.</p> <p>Step 1                  Using the spectrum analyser with Central Frequency = current DUT frequency (902 MHz to 928 MHz), span = 600 kHz or (1.5 to 5x)OBW, RBW = 10 kHz or (1% to 5% of the OBW), VBW = 30 kHz or (3xRBW), sweep time = 10 ms for SF7 or 100 ms for SF12, use trace function with Positive Peak detector and Max Hold. Choose average count at least 100. Set Reference Level such that the maximum value of the signal is at least <math>10\log(\text{OBW}/\text{RBW})</math> below the reference level. Use measurement function OBW (Power Measurements) for SF7 125 kHz and SF12 125 kHz.</p> <p>Step 2                  Measurement of PSD. Set span = (1.5xOBW), RBW = 3 kHz, VBW = 10 kHz, sweep time = 10 ms for SF7 or 100 ms for SF12, use Average mode on trace1 with RMS detector type. Choose average count at least 100. Set Auto Reference Level. Perform single measurement on the spectrum analyser; wait until the number of averaging operations have been performed. Use Marker peak measurement function. The result of the measurement is Power marker M1 in dBm. The following conditions must be fulfilled for both SF7 and SF12: Power marker <math>M1 \leq 8</math> dBm.</p>
Expected results	Power spectral density (hybrid mode)

**Test device**

IoT LoRa testset.

**Report**

The test results should be reported in Table A.14a.

**Table A.14a – Power spectral density (hybrid mode) test results**

No.	Frequency, MHz	Spreading factor	Bandwidth, kHz	OBW, kHz	PSD, dBm
1	915	SF7	125		
2	915	SF12	125		
3					
n					
s					

## Interworking testing

**Table A.15 – Test No. 15**

<b>Test number</b>	<b>No. 15</b>
Test name	Interworking
Testing layer	Physical
Type of tests	Functionality
Status	Optional
Tests goal	Coexistence test
Configuration	
Testing procedure	The test set-up consists of one vector generator with two RF outputs or ability to generate several IoT signals on Baseband or two independent vector signal generators, the signals from which are fed to the DUT as. Generator(s) supplies the LoRa wanted signals, which is generated via several LoRa ARB waveform files. The packet error rate value is measured using the LoRa test tool.
Expected results	Coexistence immunity

### Test device

IoT LoRa testset.

### Report

The test results should be reported in Table A.15a.

**Table A.15a – Power spectral density (hybrid mode) test results**

No.	Frequency, MHz	Spreading factor	Bandwidth, kHz	IoT devices	PER
1	915	SF7	125		
2	915	SF12	125		
3					
n					
s					

## Packet collision simulation testing

**Table A.16 – Test No. 16**

<b>Test number</b>	<b>No. 16</b>
Test name	Packet collision simulation
Testing layer	Physical/data
Type of tests	Functionality
Status	Optional
Tests goal	Performance of a LoRa gateway
Configuration	<p style="text-align: right;">Q.4062(20)_FA-Tab19</p>
Testing procedure	<p>The test set-up consists of one vector generator, the signal from which are fed to the DUT. Generator supplies special LoRa signals with collision, which is generated via several LoRa ARB waveform files.</p> <p>ARB files can be converted from MATLAB.</p> <p>The packet error rate value is measured using the LoRa test tool.</p>
Expected results	PER

### Test device

IoT LoRa testset.

### Report

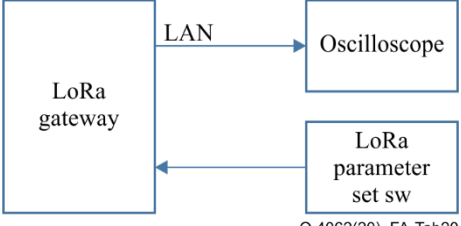
The test results should be reported in Table A.16a.

**Table A.16a – Power spectral density (hybrid mode) test results**

No.	Frequency, MHz	Spreading factor	Bandwidth, kHz	Collisions	PER
1	915	SF7	125		
2	915	SF12	125		
3					
n					
s					

## Ethernet decoding testing

**Table A.17 – Test No. 17**

<b>Test number</b>	<b>No. 17</b>
Test name	Ethernet decoding
Testing layer	Physical/data
Type of tests	Functionality
Status	Optional
Tests goal	Performance of a LoRa link
Configuration	<p style="text-align: center;">Setup</p>  <p style="text-align: right; font-size: small;">Q.4062(20)_FA-Tab20</p>
Testing procedure	The test set-up consists of one oscilloscope with triggering and decode ability for Ethernet signals.
Expected results	<ul style="list-style-type: none"> <li>Start/end of frame</li> <li>Frame</li> <li>Error frame</li> <li>Preamble/SFD/FrameCheck</li> <li>Destination address</li> <li>Source address</li> <li>Address</li> <li>Data</li> </ul>

### Test device

IoT LoRa gateway.

### Report

The test results indicated in Table A.17a should be reported:

**Table A.17a – Ethernet decoding test results**

Test reference	Tests
[b-IETF RFC 2544]	<ul style="list-style-type: none"> <li>1. Throughput;</li> <li>2. Back-to-back</li> <li>3. Frame loss</li> <li>4. Latency</li> </ul>
[b-ITU-T Y.1564]	<ul style="list-style-type: none"> <li>1. CIR</li> <li>2. EIR</li> <li>3. FLR</li> <li>4. FTD</li> <li>5. FDV</li> <li>6. AVAIL</li> </ul>

**Table A.17a – Ethernet decoding test results**

Test reference	Tests
Additional tests	1. IPDV/Jitter 2. IP connectivity (ping and traceroute commands)

**Sensors data decoding testing**

**Table A.18 – Test No. 18**

Test number	No. 18
Test name	Sensors data decoding
Testing layer	Physical/data
Type of tests	Functionality
Status	Optional
Tests goal	Performance of a sensor
Configuration	<p style="text-align: center;">Setup</p> <p style="text-align: right; font-size: small;">Q.4062(20)_FA-Tab21</p>
Testing procedure	The test set-up consists of one oscilloscope with triggering and decode ability for protocol busses like UART.
Expected results	Start and stop bits Start error, stop error, parity error Parity bit Word Word contains error

**Test device**

IoT LoRa DUT.

**Report**

The test results should be reported as indicated in Table A.18a.

**Table A.18a – Sensors data decoding test results**

OSI physical layer	<ol style="list-style-type: none"> <li>1. Determination of transmitter power</li> <li>2. Determination of the power spectral density mask (PSD)</li> <li>3. Determination of the magnitude of the error vector (EVM)</li> <li>4. Determination of the offset of the centre frequency</li> <li>5. Determination of spurious emissions</li> <li>6. RSSI</li> <li>7. LQI</li> <li>8. BER</li> </ol>
OSI link layer	<ol style="list-style-type: none"> <li>1. PER</li> <li>2. Channel utilization</li> </ol>

**Network type classification testing**

**Table A.19 – Test No. 19**

Test number	No. 19
Test name	Response time test for network type classification
Testing layer	Network
Type of tests	Functionality
Status	Optional
Tests goal	Determine the network type of the target network
Configuration	
Testing procedure	<ol style="list-style-type: none"> <li>1) Send testing packets to the IoT devices connected to the target network;</li> <li>2) Receive the responses from the devices, and measure their RTT;</li> <li>3) Repeat measuring RTTs by changing the timing to send testing packets;</li> <li>4) Select several minimum RTTs, and take their average;</li> <li>5) Classify the type of target network by the average RTT and pre-determined thresholds.</li> </ol>
Expected results	Response time and type of network

**Test network**

Network to which IoT device(s) belong.

**Report**

The response time and estimated network type of the target network should be reported.

## Appendix I

### Estimation method on network types from RTT samples

(This appendix does not form an integral part of this Recommendation.)

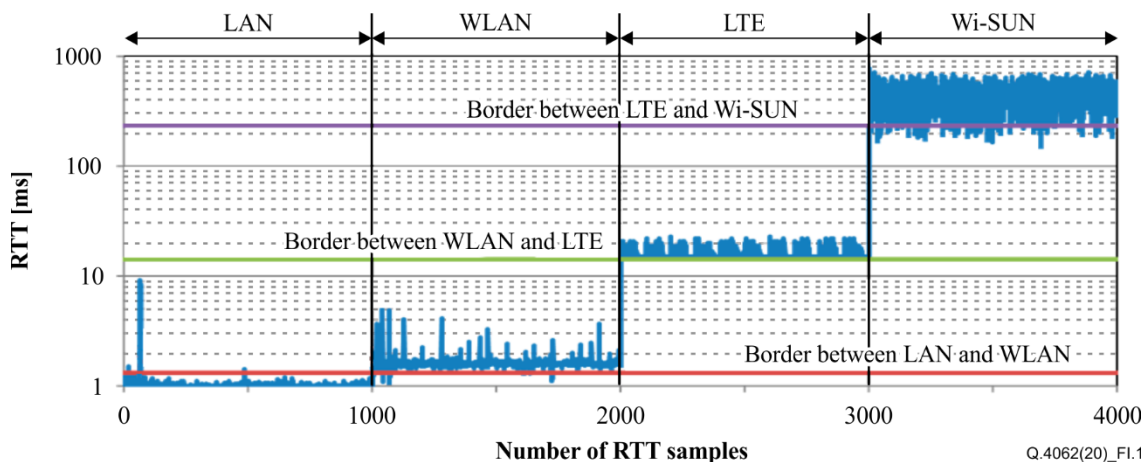
This appendix provides reference information about the network type classification test.

In non-congested situations, the RTT of the target network distributes in different ranges according to the network types. This means that the type of the target network can be classified if the borderlines between the RTT ranges of each network type are obtained. The detailed procedure of the classification is as follows.

- 1) Collect RTT values of candidate networks such as LAN (Ethernet), WLAN, 3GPP LTE and LPWAN (Wi-SUN) and so on, in non-congested situations by preliminary experiment or simulation.
- 2) Obtain the classification thresholds by applying a clustering algorithm such as k-means clustering to the RTT values collected by Step 1).
- 3) Compare the target RTT value (i.e., the average value of RTTs obtained by Step 4) in clause 7.5) and the borderlines, and detect the network type whose RTT range includes the target RTT.

Figure I.1 shows samples of RTT values to validate the above classification method. The candidate networks are LAN (Ethernet), WLAN, 3GPP LTE and LPWAN (Wi-SUN). RTT values were obtained by indoor experiment and simulations assuming that the target network is in an intra-network. As shown in Figure I.1, each network has a different range of RTTs, and the RTT thresholds are finely obtained by k-means clustering.

For the network classification test in an intra-network, the regions of the target RTTs  $t_{rtt}$  to classify the target network as LAN, WLAN, LTE, Wi-SUN, and other extremely narrow band networks, that can be obtained from Figure I.1, are summarized in Table I.1.



**Figure I.1 – Examples of RTTs in LAN, WLAN, 3GPP LTE and Wi-SUN and thresholds to classify the network types obtained by k-means algorithm in an intra-network**

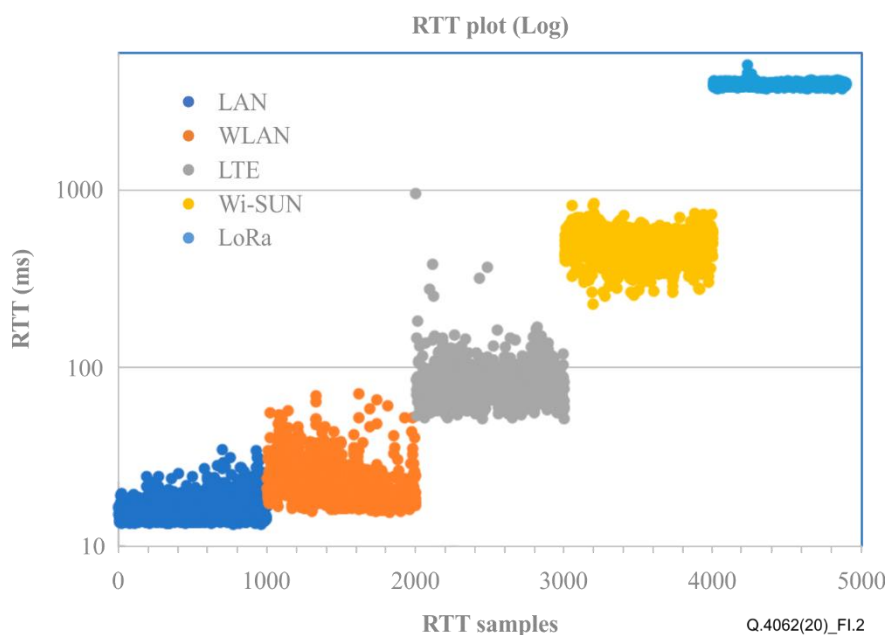


**Table I.1 – The regions of the target RTTs ( $t_{rtt}$ ) to classify the target network in an inter-network based on Figure I.1**

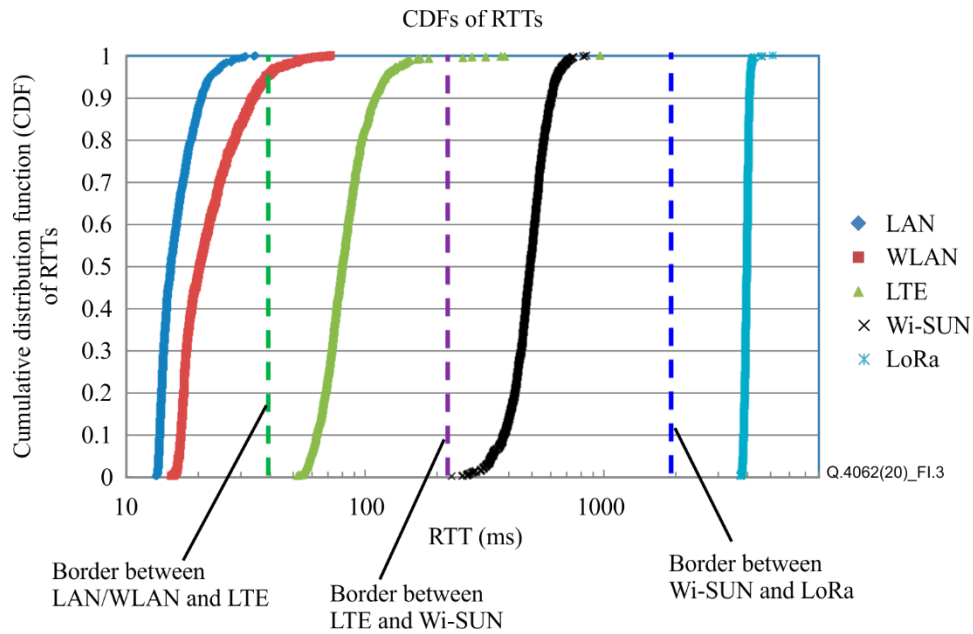
Range of target RTT ( $t_{rtt}$ )	Classification output
$t_{rtt} < 1.3$ ms	LAN
$1.3$ ms $\leq t_{rtt} < 14$ ms	WLAN
$14$ ms $\leq t_{rtt} < 220$ ms	LTE
$220$ ms $\leq t_{rtt} < 2\,000$ ms	Wi-SUN
$2\,000$ ms $\leq t_{rtt}$	Extremely narrow band network

Figure I.2 shows samples of RTT values measured over the Internet to validate the above classification method. The target networks to be tested were LAN, WLAN, 3GPP LTE, Wi-SUN, and LoRa. RTT values were experimentally obtained over the Internet with a testing server which was geographically apart from the target networks. Figure I.3 shows the cumulative distribution function (CDF) of RTT samples for the target networks. As shown in these figures, the range of RTTs overlaps between LAN and WLAN. In other words, the RTT range of LAN is similar to that of WLAN, so it is difficult to distinguish between LAN and WLAN, and thus they should be classified as "LAN/WLAN".

For the network classification test over the Internet, the regions of the target RTTs  $t_{rtt}$  to classify the target network as LAN/WLAN, LTE, Wi-SUN, and LoRa (or other extremely narrow band networks), that can be obtained from Figures I.2 and I.3, are summarized in Table I.2.



**Figure I.2 – RTTs measured over the Internet for the target networks including LAN, WLAN, 3GPP LTE, Wi-SUN and LoRa**



**Figure I.3 – Examples of CDF of RTTs in LAN, WLAN, 3GPP LTE, Wi-SUN and LoRa obtained over the Internet, and thresholds to classify the network types beyond the Internet**

**Table I.2 – The regions of the target RTTs ( $t_{rtt}$ ) to classify the target network beyond the Internet based on Figures I.2 and I.3**

Range of target RTT ( $t_{rtt}$ )	Classification output
$t_{rtt} < 40$ ms	LAN/WLAN
$40$ ms $\leq t_{rtt} < 220$ ms	LTE
$220$ ms $\leq t_{rtt} < 2\ 000$ ms	Wi-SUN
$2\ 000$ ms $\leq t_{rtt}$	LoRa (or extremely narrow band network)

## Appendix II

### Examples of IoT device detection and classification

(This appendix does not form an integral part of this Recommendation.)

For remote testing of IoT devices, device detection and classification is necessary. One of the possible device detection and classification approaches is presented in clause 8.5.1.

A searching system with structure described in this appendix can be used for device detection and classification.

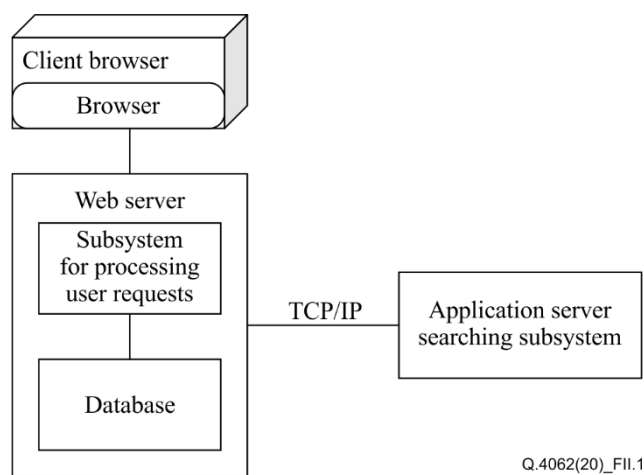
A searching system includes several subsystems:

- **A storage subsystem** – is designed to store data about detected devices in a database. The overall system is centralized, so all information is stored in one database.
- **A subsystem for processing user requests** – is a web application with which users can enter search requests and receive answers in the form of a list of devices that match the requests.
- **A subsystem for scanning devices connected to the network** – the subsystem collects information about devices, as well as transferring the information to the storage subsystem.

The subsystem for scanning devices connected to the network includes two modules:

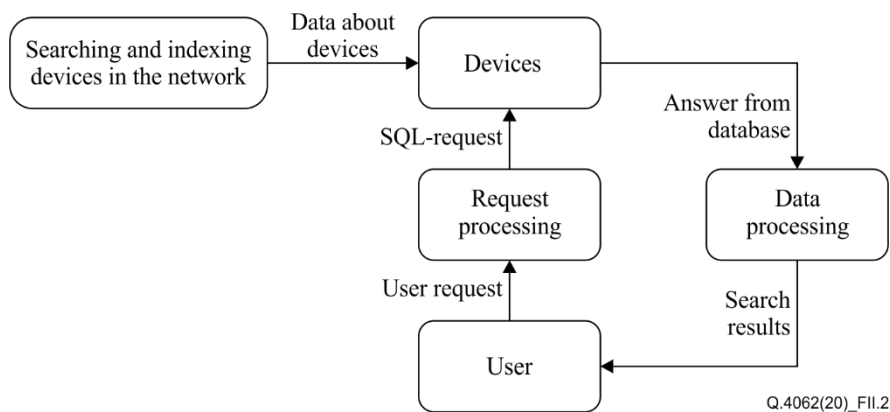
- **A device search module** – is designed to search and determine the availability of devices in the network, scan device ports and determine the application layer protocols that are used by the device to transmit the data.
- **An indexing module** – is designed to analyse and transfer the information, which were received in the device search model, to the database.

Figure II.1 shows the searching system structure.



**Figure II.1 – Searching system structure**

The data flow exchanged between the subsystems is shown in Figure II.2.



**Figure II.2 – UML diagram of data flow**

Figure II.2 shows the data flow that occurs between the data storage subsystem and the user request processing subsystem. The structure of the transmitted data includes:

- Information about devices;
- User requests in the form of HTTP GET request parameters;
- Data responses in JavaScript object notation (JSON) format to user requests (using JSON format is an example of implementation);
- Structured query language (SQL) queries to interact with the database.

In the database, there are following tables:

- IP table – for saving devices IP addresses;
- Open\_ports\_info table – for saving information about all open ports that are detected in the network;
- Protocol table – for saving the names of all protocols that the search program recognizes;
- Region\_info table – for saving information about the region in which the detected devices are located;
- Type table – for saving the names of types of devices that were detected in the network.

The search subsystem has two modules for solving two related tasks:

- Search for devices in the network and collect information about each of them;
- Transforming, analysing and forwarding the received information to the database.

The first module performs the following tasks:

- Determining devices availability by IPv4;
- Searching for open ports through which the device sends data;
- Defining the application layer protocol for each open port;
- Sending test requests and receiving responses for each application protocol.

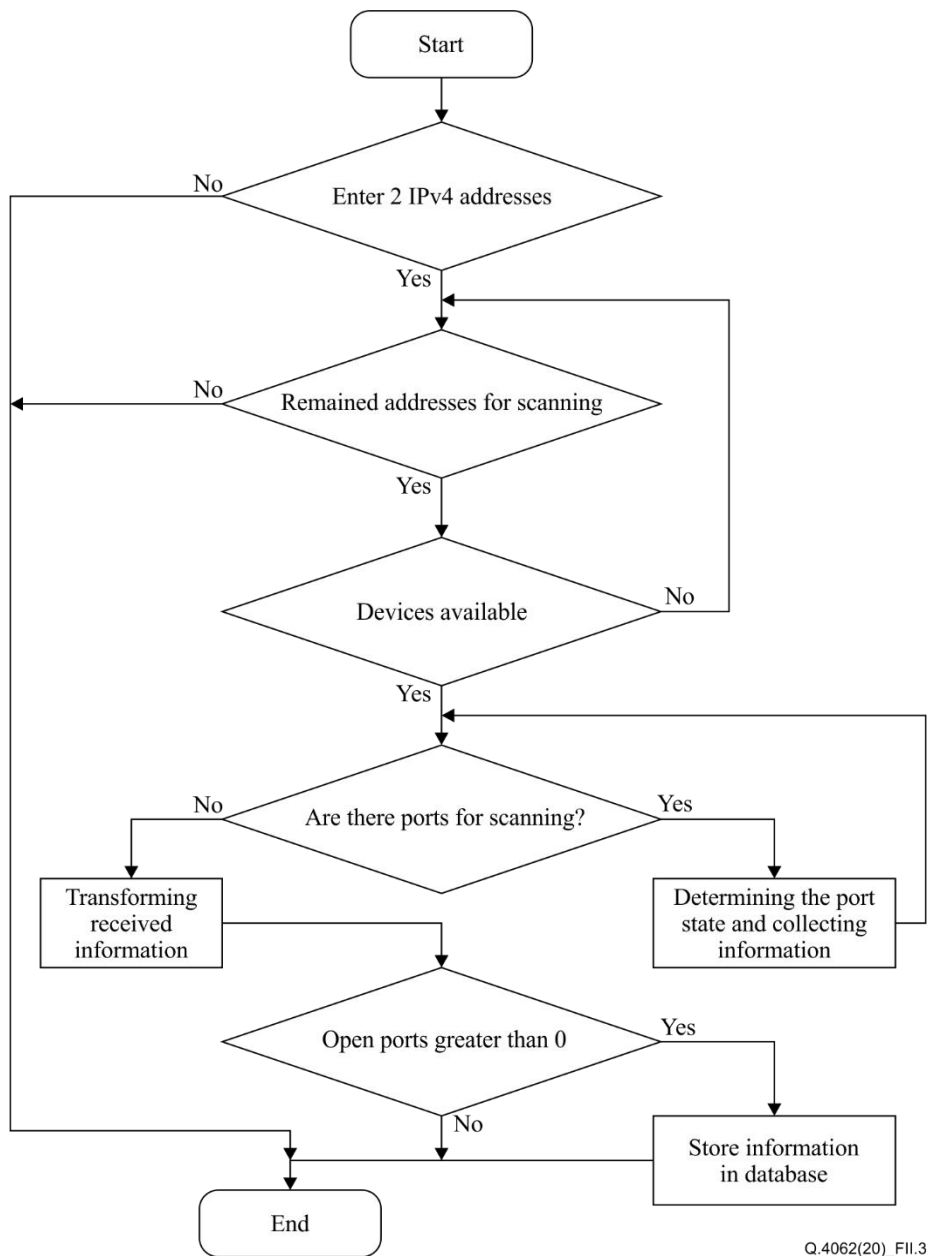
Device availability is determined by sending ICMP echo-request packets to the target address and receiving ICMP echo-reply responses as how the ping utility works.

The search for open ports provides a search for existing communication channels. The task is to find as many of these communication channels as possible and identify those that are in a standby state. There are several ways to scan TCP ports:

- Scanning using the connect () function, which allows to connect to one of the ports on the remote device. If the port with which the connection is being attempted is open, then a connection to the server will be made; otherwise, the port is closed. This method provides a high searching speed if the methods of asynchronous or non-blocked input-output is available. However, the disadvantage of this method is the high probability of detecting a scan followed by filtering.
- Scanning with the SYN flag – the scanner sends a packet that contains the SYN flag to the server for creating a connection and waits for a response. If a packet with the ACK flag is replied, it means that the port of the device is open. If a packet with the RST flag comes in the response, then this means that the port is closed, and the connection cannot be established. In the case of an ACK packet, the scanner immediately sends a packet with the RST flag in response to reset the connection. This scanning is hardly detected. However, this is available only for scanners on UNIX operating systems, and it is required to have root permission.
- Scanning with the FIN flag – this method is used to bypass security features and mask port scanning attempts. The scanner sends a packet with a FIN flag and awaits a response. If the answer does not come, then the port is open, since FIN packets are ignored by open ports. If a packet with the RST flag comes in response, then the port is closed. However, this method has its drawbacks. Not all operating systems support the scheme described above, therefore, there is scanning immunity with the flag FIN.

The second module of the search subsystem determines what type of devices belongs to and their location, and then saves the information to the database. The type of device is determined based on the protocols that it uses, as well as on the responses that were received from open ports as mentioned above.

Figure II.3 shows a block diagram of the operation of searching for devices connected to the network.



**Figure II.3 – A block diagram of the operation of searching**

## Appendix III

### Detail test procedure for LoRa connectivity

(This appendix does not form an integral part of this Recommendation.)

An oscilloscope and a special multichannel power probe can be driven via an instrument and test set remote controller module. A test plan should include both signalling and power measurements which show the detailed value of the power consumption in different signalling states as well as an estimation of overall lifetime given a certain battery capacity.

#### III.1 LoRa devices test

Hardware and tools to use:

- 1) Multichannel probe.
- 2) Instrument and test set remote controller module.
- 3) Battery life measurement for LoRa.
- 4) Real RF channel modelling module.

Parameters for testing IoT devices:

- a) Multichannel probe with 2 or 4 simultaneous voltage and current measurement channels depending on IoT device under test.  
Each channel should have 18-bit resolution analogue to digital converters (ADC), 5 MSa/s sampling rate, channels are able to handle voltages up to  $\pm 15$  Volts and 10 Ampere when using the internal shunt. The accuracy can be increased by selecting the correct range for the ADC when working with lower voltages and currents.
- b) Depending on the IoT device under test, it is possible that an instrument and test set remote controller module control several instruments that are used during the testing phase. A communication tester with power measurement software in network emulation mode or an oscilloscope with an RF generator can be used. (see Table A.9a)
- c) Battery life measurement for LoRa devices: in order to calculate the total service time or end of life using the same battery set, the measurement of the power consumption per packet transmission is necessary. The devices are generally in sleep mode for most of the lifetime and only get active in operational mode in order to transmit data to the LoRa gateway. The power consumption in sleep mode needs to be measured in order to endorse the battery lifetime. (see Table A.9b)
- d) IoT standards, for example LoRaWAN define the media access protocol (MAC) and the system architecture for a wide area network (WAN). Using Real RF channel modelling module, it is possible to simulate different communication technologies such as RFID, ZigBee, 6LoWPAN, ETSI M2M, IEEE, 3GPP LTE and LTE-A, and TIA SDC.

#### III.2 LoRa receivers test

Hardware and tools to use:

- 1) Vector signal generator with a built-in arbitrary waveform generator (ARB).
- 2) USB stick with a set of LoRa ARB waveform files for a signal generator.
- 3) Software/hardware test tool provided by the manufacturer of the LoRa hardware module.
- 4) Analogue signal generator.
- 5) RF power combiner.
- 6) Device under test (DUT).

Parameters for testing receivers:

- a) RF blocking measurement at a LoRa receiver

The blocking test is used to check the behaviour of the receiver when an interference signal is applied. The packet error rate (PER) value is measured using the LoRa test tool.

- b) Reception of the test RF LoRa signal

For the Rx sensitivity test, load a set of LoRa ARB waveform files in the vector signal generator for testing the sensitivity of the receiver. While the signal power is being reduced from generator output, the LoRa test tool is used to monitor the packet error rate (PER). (see Table A.10)

### III.3 LoRa transmitter test

Hardware and tools to use:

- 1) Spectrum analyser.
- 2) Software/hardware test tool provided by the manufacturer of the LoRa hardware module.
- 3) Device under test (DUT).

A test signal is generated using a test tool of the transmitter module manufacturer. The transmit signal generated is fed and analysed using the compact spectrum analyser displaying the results at high resolution on a large touchscreen with gesture control.

Parameters for testing transmitters:

- a) 6 dB bandwidth

In the frequency range 902 MHz to 928 MHz the 6 dB signal bandwidth of a digitally modulated signal must be at least 500 kHz for LoRa. (Table A.11)

- b) Occupied bandwidth

The total output power and the band power respectively are determined by integrating the power over the signal bandwidth. The signal bandwidth corresponds to the occupied bandwidth (OBW). The OBW is the bandwidth in which 99% of the signal power is contained. (Table A.12)

- c) Emission output power

The measurement performed using the band power measurement function of the spectrum analyser. The following conditions must be fulfilled: Band power  $\leq 30$  dBm (Table A.13)

- d) Power spectral density

According to FCC 15.247(e), the power spectral density of a transmitter in the frequency range 902 MHz to 928 MHz must at no time exceed the value of 8 dBm relative to a bandwidth of 3 kHz during an ongoing data transmission.

The following conditions must be fulfilled:

$$\text{Power marker } M1 \leq 8\text{dBm. (Table A.14)}$$

- e) Emissions in non-restricted bands

According to FCC 15.247(d), the radiated power outside the ISM band (902 GHz to 928 GHz) must be at least 30 dB below the maximum RF emission within the ISM band. Below is an example demonstrating the analysis of the RF emissions of a LoRa signal with SF7 at the lower and upper band limit.

The following condition must be fulfilled:

$$\text{REF}_{hi} - M1 \geq 30 \text{ dB}$$

$$\text{REF}_{lo} - M1 \geq 30 \text{ dB (Table A.15)}$$



f) 20 dB bandwidth

According to FCC 15.247, in the frequency range 902 MHz to 928 MHz the 20 dB bandwidth of a frequency hopping spread spectrum (FHSS) transmit signal must not exceed the value of 500 kHz. For a LoRa signal in FHSS mode. This means that the 20 dB bandwidth of 500 kHz must not be exceeded for the signal bandwidths 125 kHz and 250 kHz.

The following condition must be fulfilled:

$$n \text{ dB down BW} \leq 500 \text{ kHz. (Table A.16)}$$

g) Power Spectral Density (Hybrid Mode)

DUT settings = LoRa, 915 MHz, a) SF7, 125 kHz, b) SF7, 250 kHz, c) SF12, 125 kHz, d) SF12, 250 kHz.

The following conditions must be fulfilled for both SF7 and SF12:

$$\text{Power marker M1} \leq 8 \text{ dBm (Table A.17)}$$

### III.4 Additional testing

a) Interworking (Table A.18)

Simulation several IoT coexistence on a signal generator.

b) Packet collision simulation (Table A.19)

PER of a LoRa link in case of collision between LoRa packets modulated with different spreading factors.

c) Ethernet decoding (Table A.20)

Analyse Ethernet protocol variants by decoding the signal and searching within the decoded events. LoRa gateway testing.

d) Sensors protocol triggering and decoding (Table A.21)

Protocol decoding: The digitized signal data is displayed on the screen together with the decoded content of the messages in readable form, and the decode results are listed in a table.

## Bibliography

- [b-ITU-T Q.4060] Recommendation ITU-T Q.4060 (2018), *The structure of the testing of heterogeneous Internet of things gateways in a laboratory environment.*
- [b-ITU-T Y.1564] Recommendation ITU-T Y.1544 (2008), *Multicast IP performance parameters.*
- [b-ITU-T Y.4500.13] Recommendation ITU-T Y.4500.13/Q.3954 (2018), *oneM2M – Interoperability testing.*
- [b-IETF RFC 2544] IETF RFC 2544 (1999), *Benchmarking Methodology for Network Interconnect Devices.*
- [b-Heegard] C. Heegard, *Range versus Rate in IEEE 802.11 Wireless Local Area Networks.*  
<http://alantro.com/~heegard/papers/RvR.pdf>.



## SERIES OF ITU-T RECOMMENDATIONS

Series A	Organization of the work of ITU-T
Series D	Tariff and accounting principles and international telecommunication/ICT economic and policy issues
Series E	Overall network operation, telephone service, service operation and human factors
Series F	Non-telephone telecommunication services
Series G	Transmission systems and media, digital systems and networks
Series H	Audiovisual and multimedia systems
Series I	Integrated services digital network
Series J	Cable networks and transmission of television, sound programme and other multimedia signals
Series K	Protection against interference
Series L	Environment and ICTs, climate change, e-waste, energy efficiency; construction, installation and protection of cables and other elements of outside plant
Series M	Telecommunication management, including TMN and network maintenance
Series N	Maintenance: international sound programme and television transmission circuits
Series O	Specifications of measuring equipment
Series P	Telephone transmission quality, telephone installations, local line networks
<b>Series Q</b>	<b>Switching and signalling, and associated measurements and tests</b>
Series R	Telegraph transmission
Series S	Telegraph services terminal equipment
Series T	Terminals for telematic services
Series U	Telegraph switching
Series V	Data communication over the telephone network
Series X	Data networks, open system communications and security
Series Y	Global information infrastructure, Internet protocol aspects, next-generation networks, Internet of Things and smart cities
Series Z	Languages and general software aspects for telecommunication systems