# ISO/IEC 30165

Edition 1.0 2021-07

# INTERNATIONAL STANDARD

colour inside

## Internet of things (IoT) – Real-time IoT framework



ISO/IEC 30165:2021-07(en)

**About the IEC**
The International Electrotechnical Commission (IEC) is the leading global organization that prepares and publishes International Standards for all electrical, electronic and related technologies.

**About IEC publications**
The technical content of IEC publications is kept under constant review by the IEC. Please make sure that you have the latest edition, a corrigendum or an amendment might have been published.

**IEC publications search - webstore.iec.ch/advsearchform**
The advanced search enables to find IEC publications by a variety of criteria (reference number, text, technical committee, …). It also gives information on projects, replaced and withdrawn publications.

**IEC Just Published - webstore.iec.ch/justpublished**
Stay up to date on all new IEC publications. Just Published details all new publications released. Available online and once a month by email.

**IEC Customer Service Centre - webstore.iec.ch/csc**
If you wish to give us your feedback on this publication or need further assistance, please contact the Customer Service Centre: sales@iec.ch.

**IEC online collection - oc.iec.ch**
Discover our powerful search engine and read freely all the publications previews. With a subscription you will always have access to up to date content tailored to your needs.

**Electropedia - www.electropedia.org**
The world's leading online dictionary on electrotechnology, containing more than 22 000 terminological entries in English and French, with equivalent terms in 18 additional languages. Also known as the International Electrotechnical Vocabulary (IEV) online.

# ISO/IEC 30165

Edition 1.0 2021-07

# INTERNATIONAL STANDARD



colour inside

## Internet of things (IoT) – Real-time IoT framework

INTERNATIONAL
ELECTROTECHNICAL
COMMISSION

ICS 33.020

ISBN 978-2-8322-9897-8

CONTENTS

# INTERNET OF THINGS (IoT) – REAL-TIME IoT FRAMEWORK

## FOREWORD

1) ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work.
2) The formal decisions or agreements of IEC and ISO on technical matters express, as nearly as possible, an international consensus of opinion on the relevant subjects since each technical committee has representation from all interested IEC and ISO National bodies.
3) IEC and ISO documents have the form of recommendations for international use and are accepted by IEC and ISO National bodies in that sense. While all reasonable efforts are made to ensure that the technical content of IEC and ISO documents is accurate, IEC and ISO cannot be held responsible for the way in which they are used or for any misinterpretation by any end user.
4) In order to promote international uniformity, IEC and ISO National bodies undertake to apply IEC and ISO documents transparently to the maximum extent possible in their national and regional publications. Any divergence between any IEC and ISO document and the corresponding national or regional publication shall be clearly indicated in the latter.
5) IEC and ISO do not provide any attestation of conformity. Independent certification bodies provide conformity assessment services and, in some areas, access to IEC and ISO marks of conformity. IEC and ISO are not responsible for any services carried out by independent certification bodies.
6) All users should ensure that they have the latest edition of this document.
7) No liability shall attach to IEC and ISO or their directors, employees, servants or agents including individual experts and members of its technical committees and IEC and ISO National bodies for any personal injury, property damage or other damage of any nature whatsoever, whether direct or indirect, or for costs (including legal fees) and expenses arising out of the publication, use of, or reliance upon, this ISO/IEC document or any other IEC and ISO documents.
8) Attention is drawn to the Normative references cited in this document. Use of the referenced publications is indispensable for the correct application of this document.
9) Attention is drawn to the possibility that some of the elements of this ISO/IEC document may be the subject of patent rights. IEC and ISO shall not be held responsible for identifying any or all such patent rights.

ISO/IEC 30165 has been prepared by subcommittee 41: Internet of Things and Digital Twin, of ISO/IEC joint technical committee 1: Information technology. It is an International Standard.

The text of this International Standard is based on the following documents:

| FDIS | Report on voting |
|---|---|
| JTC1-SC41/216/FDIS | JTC1-SC41/229/RVD |

Full information on the voting for its approval can be found in the report on voting indicated in the above table.

The language used for the development of this International Standard is English.

This document was drafted in accordance with ISO/IEC Directives, Part 2, and developed in accordance with ISO/IEC Directives, Part 1, available at www.iec.ch/members_experts/refdocs and www.iso.org/directives.

**IMPORTANT – The 'colour inside' logo on the cover page of this publication indicates that it contains colours which are considered to be useful for the correct understanding of its contents. Users should therefore print this document using a colour printer.**

# INTRODUCTION

This document addresses a special kind of IoT system operating in real-time that is called real-time IoT (RT-IoT) systems.

The correct behaviour of a real-time system depends not only on the logical correctness, but also on the timeliness of its actions. Design and development of a real-time system are different from conventional computer systems in terms of real-time OS, embedded development, task scheduling, etc.

[1] [1] emphasizes the requirements of timeliness and predictability in real-time systems as follows.

"The challenges and trade-offs faced by the designers of real-time systems are quite different from those who design general purpose computing systems. To achieve the fundamental requirements of timeliness and predictability, not only do conventional methods for scheduling and resource management have to be redesigned, but new concepts that have not been considered in conventional systems need to be added. New paradigms are necessary to specify and validate real-time systems."

Lack of understanding of real-time systems could lead to unsuccessful RT-IoT system deployment where real-time computation is required. A deployment of an RT-IoT system based on the very general real-time capabilities defined in ISO/IEC 30141 [2] is not enough to fully support real-time requirements. Therefore, it is important to complement the real-time capabilities of IoT reference architecture for RT-IoT systems.

Basically, an RT-IoT system has features of a typical IoT system except real-time capability. ISO/IEC 30141 explains real-time capability of an IoT system as follows:

– a characteristic of a system or mode of operation in which computation is performed during the actual time that an external process occurs, in order that the computation results can be used to control, monitor, or respond in a timely manner to the external process

Considering the characteristics of real-time capability, any IoT system embraces real-time aspects to some extent simply because it continuously interacts with the physical world.

Requirements for real-time capability depend on the peer that an IoT system interfaces with. For example, a human–machine interface guarantees a maximum delay of 250 ms in presenting responses to humans, whereas 150 ms is sufficient in a telephone service. Any IoT system interfacing with physical things guarantees some extent of timeliness because any event in the physical world demands timely adjustment from the IoT system.

This document focuses on real-time capability in addition to very general description given in ISO/IEC 30141, because failing on timing constraints could cause serious damage to an IoT system or to its environment, including injury or even death of people involved. Certain RT-IoT systems, such as industrial IoT (IIoT) systems and cyber-physical systems (CPS), consider time as of high importance.

The purpose of this document is to provide a guideline for deploying an RT-IoT system to avoid pitfalls that usually occur during real-time system developments.



---

1 Numbers in square brackets refer to the Bibliography.

# INTERNET OF THINGS (IoT) – REAL-TIME IoT FRAMEWORK

## 1 Scope

This document specifies the framework of a real-time IoT (RT-IoT) system, including:

– RT-IoT system conceptual model based on domain-based IoT reference model defined in ISO/IEC 30141;

– impacts of real-time parameters in terms of four viewpoints (time, communication, control and computation).

## 2 Normative references

There are no normative references in this document.

## 3 Terms, definitions and abbreviated terms

### 3.1 Terms and definitions

For the purposes of this document, the following terms and definitions apply.

ISO and IEC maintain terminological databases for use in standardization at the following addresses:

- IEC Electropedia: available at http://www.electropedia.org/
- ISO Online browsing platform: available at http://www.iso.org/obp

**3.1.1**
**real-time IoT**
**RT-IoT**
IoT system with timing constraints

Note 1 to entry: Correct behaviour of an RT-IoT system depends not only on the logical correctness, but also on being able to meet timing constraints.

**3.1.2**
**timeliness**
property of meeting timing constraints, finishing before the deadline and giving compulsory response

### 3.2 Abbreviated terms

| | |
|---|---|
| 2G/3G/4G | second/third/fourth generation mobile networks |
| 5G | fifth generation mobile networks |
| AI | artificial intelligence |
| ASD | application and service domain |
| CPU | central process unit |
| IoT | Internet of Things |
| LAN | local area network |
| OMD | operation and management domain |
| PED | physical entity domain |

| | |
|---|---|
| QoS | Quality of Service |
| RAID | resource access and interchange domain |
| RT | real time |
| TSN | time-sensitive networking |
| UD | user domain |

# 4 Conceptual model

## 4.1 General

Figure 1 illustrates the conceptual model based on the IoT reference architecture in ISO/IEC 30141. The conceptual model of an RT-IoT system inherits domain-based reference model and extends it in terms of timing constraints.

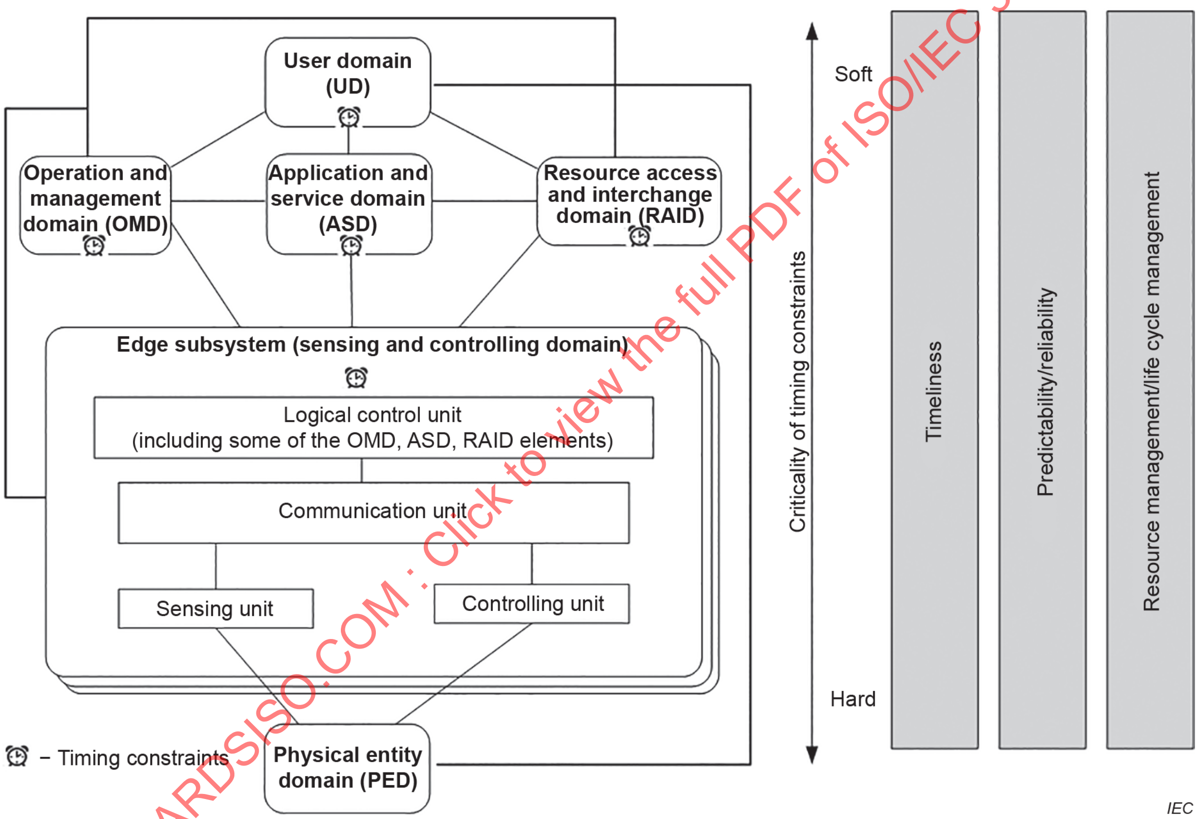


**Figure 1 – RT-IoT system conceptual model**

## 4.2 The six domains

First of all, an RT-IoT system is still an IoT system. Its conceptual model is extended from that of the six domain model for an IoT system.

## 4.3 Edge sub-system

The edge sub-system is an entity of an RT-IoT system, which encompasses IoT device entity and IoT gateway entity defined in ISO/IEC 30141.

An RT-IoT system can be deployed at different scales from large scale that has a number of edge sub-systems to small scale that has limited number of edge sub-systems.

As an IoT sub-system, the edge sub-system also includes some elements of other domains to be a complete functioning system; it is a small scale RT-IoT system. A large scale IoT system could be a system of multiple collaborating IoT systems. This document and ISO/IEC 30141 should be referenced in designing a system of multiple collaborating RT-IoT systems.

The edge sub-system is in some ways similar to edge computing system described in other literature.

### 4.4 Timing constraints

Timing constraints are the major characteristics of an RT-IoT system. Timing constraints could be within a domain or between domains, could be local or global.

Each domain and each functional component need to have a sense of time. For those domains or functional components involved in a shared timing constraint, their local clock should be synchronized with a master clock. It could be a single master clock in the whole system. On top of this, the timing constraints could be defined and satisfied.

Timing constraints are classified into two categories according to the consequence that violating the timing constraints causes in an RT-IoT system. Hard real-time sometimes requires that a deadline should never be missed, otherwise catastrophes can happen. In soft real-time, the timing constraints could be violated without causing significant consequence.

The edge sub-system interacts with the physical world generally under the requirements of hard real-time. The delay of time is shorter as the edge sub-system is closer to the physical world. This is indicated with the word hard at the bottom in Figure 1. However, the criticality of implementing and handling timing constraints becomes soft far away from the PED, where more computation power is usually available such as OMD and UD. This is indicated with the word soft at the top in Figure 1.

NOTE This document provides a guideline for deploying an RT-IoT system. While it covers in general terms, future documents will cover many related topics in more detail, such as the following.

– Support of timeliness, availability, and predictability in the vertical dimension. The timely delivery of data from an IoT device to the cloud, and vice versa.
– Real-time constraints across multiple applications.
– Real-time resource management in complex cases.
– The life-cycle management. Design and validation of an RT-IoT system depends on specific application area and follows specific practices. It might include, for instance, certification.
– Other new technologies referenced in Annex A.

## 5 Viewpoints of RT-IoT system conceptual model

### 5.1 Time view

#### 5.1.1 General

Time view represents time-related characteristics of an RT-IoT system. Entities of an RT-IoT system should have a common understanding of the clock to reference the time, although the precision can differ in each entity. Figure 2 shows the RT-IoT system time view.

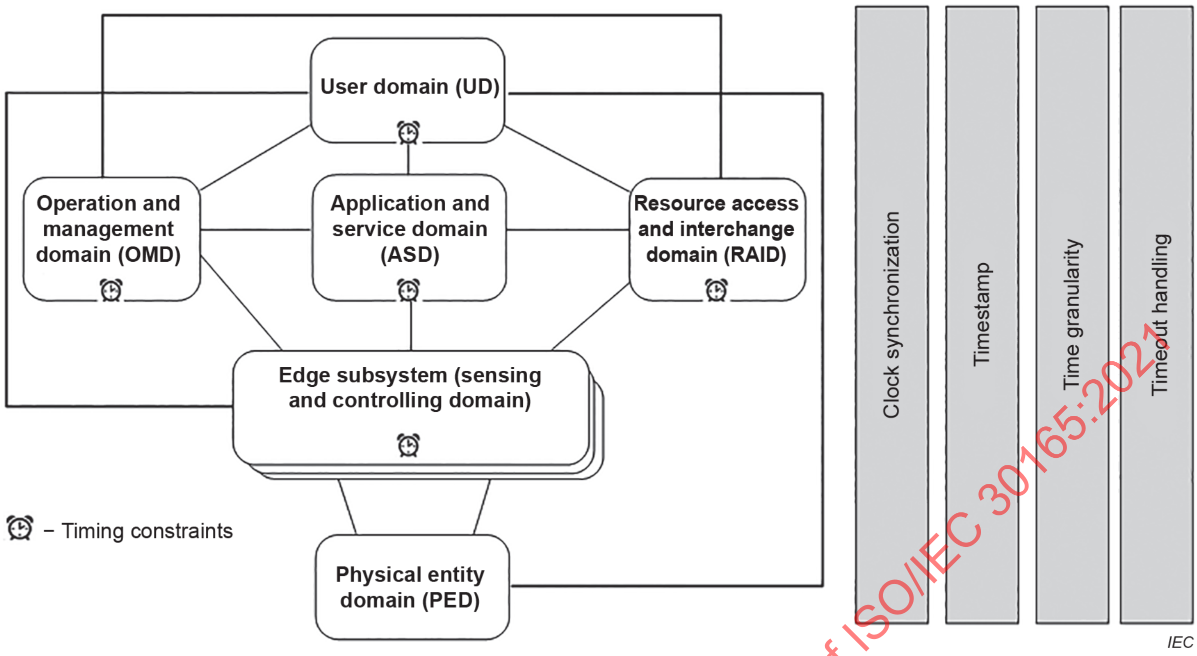


**Figure 2 – RT-IoT system time view**



### 5.1.2 Clock synchronization

In an RT-IoT system, all functional components should have a clock that is synchronized to a master clock for distributed timing constraints, if the components together are required to meet certain timing constraints. However, a master clock is optionally synchronized to the global standard time. In some cases, a component's clock may be of different synchronization precision, or even not synchronized.

IEC 61588 [3] defines a precision clock synchronization protocol for networked measurement and control systems.

The Internet has a high accuracy in time with the help of several international atomic clocks. An IoT system could synchronize with the Internet time, referencing of which could be important in troubleshooting, legal context, etc. The NTP (Network Time Protocol) standard [4] defines a synchronization protocol used on Internet.

### 5.1.3 Timestamp

In an RT-IoT system, events, data, actions, etc., are all able to be timestamped.

### 5.1.4 Time granularity

An RT-IoT system should support appropriate time granularity, or length of time defined in the timing constraints.

Being fast or quick does not mean being real-time. The time unit measured in real-time constraints could be large or small, but the key is for the system to be timely and predictable.

Different IoT systems could have different time granularity. A driverless car needs to react to external events in microseconds, whereas a smart building adjusts room temperature in minutes. Within an RT-IoT system, the actuator can act in milliseconds, but the self-diagnostics routine could run intermittently in the background. Hard real-time systems could be systems with large time granularity, and vice versa. For example, distributed control systems in the process control industry are often hard real-time systems, although their control cycles are often in seconds.

### 5.1.5 Timeout handling

An RT-IoT system should implement a timeout handling mechanism.

While an RT-IoT system should finish all tasks within the deadlines. One major feature of a real-time system is its component handling timeouts. There are different handling approaches. Some applications allow rolling back the modification. A process application can entail opening a relieving valve after a task misses the deadline. In other cases, the system is simply rebooted.

## 5.2 Communication view

### 5.2.1 General

The need for timely interaction between different elements entails real-time support from the communication networks. Deterministic network capability, if needed for timeliness, is essential. Timeliness of communication is achieved through properties such as bandwidth management, latency, and redundancy management.

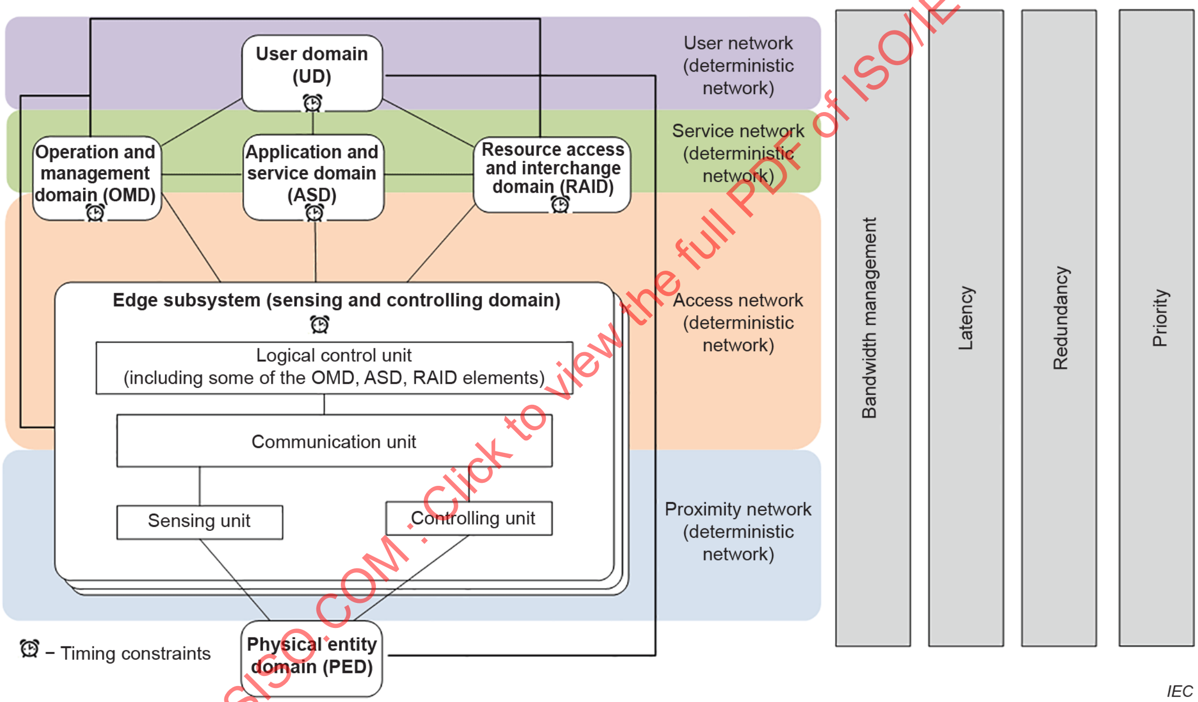


**Figure 3 – RT-IoT system communication view**

Of the networks in an IoT system, the field level networks are usually fieldbuses and usually provide hard real-time support. Soft real-time support is expected at the administrative level, i.e. actions involving a human to manage the system.

There are different network topologies and technologies among the four network types in Figure 3. They are explained in ISO/IEC 30141 [2]. Some IoT devices are part of a fixed installation and can be connected to a wired or wireless network, and some IoT devices are mobile and hence connected to a wireless network, and each network technology has a different and potentially varying latency. Regardless of all those differences, a network should provide different degrees of timing support. In an RT-IoT system, a mobile device's wireless communication could also meet stringent timing constraints.

### 5.2.2 Bandwidth management

Bandwidth is the rate at which bytes of data can be moved across a given network connection. The lower the data rate, the longer it takes to move a given amount of data across the communication link. The network traffic needs to be managed so that real-time data traffic flows smoothly. There should be no blocking for time-sensitive data. While guaranteed bandwidth needs to be allocated for real-time data, it is better for the total bandwidth to be fully utilized for all data traffic. Bandwidth could be pre-allocated or dynamically allocated.

An RT-IoT system should provide a mechanism to convey time sensitivity of a data package or a communication session to the network that routes the data accordingly. Priority is commonly used for this purpose.

The supported data rate is usually fixed by the technology involved in the network connection. The supported data rate can vary substantially among different network technologies.

The term Quality of Service (QoS) is usually used in the network community to describe data communication support. If QoS ranges from 0 to 1, then hard real-time task demands QoS = 1.

### 5.2.3 Latency and jitter

An RT-IoT system should employ techniques addressing network latency in its design. Real-time data, which can be timestamped, should be delivered before it expires. Latency is the time delay in delivering data from one node across a network connection to another node.

An RT-IoT system should employ techniques addressing network jitter in its design. It is typical for the actual time taken for the end-to-end process in a given RT-IoT system to vary to some extent – such variation in the time taken is termed jitter. Jitter can be an intrinsic aspect of the system concerned, or it can vary depending on the state of the system – in particular, the time taken could increase with the load (or throughput) of the system.

### 5.2.4 Redundancy

Network redundancy is one of the means to provide guaranteed data delivery while meeting the timing requirements. When failure occurs, the redundant data delivery should still be finished within the deadline. Redundant data could be sent simultaneously, or only when the previous delivery fails. Redundant data could also be sent on different network links.

Link aggregation is one of the means to guarantee data delivery while meeting the timing requirements. When data delivery fails, the data should be redelivered via an alternative network link within the deadline.

### 5.2.5 Priority

Data should be prioritized and treated differently by the network. High-priority data gets transmitted earlier and faster for higher priority tasks. Priority is the basic concept throughout any real-time system. It is emphasized in this view because an IoT system is by definition a networked system. A prioritized network is essential to provide real-time guarantee.

## 5.3 Control view

### 5.3.1 General

Historically, control systems were common in industrial environments, for example being used on production lines to ensure the smooth operation of those production lines. Most RT-IoT systems include actuators in the edge subsystems to act upon the environment. They are the extension of a typical control system. The practice of a control system applies to an RT-IoT system. Figure 4 shows the RT-IoT system control view.

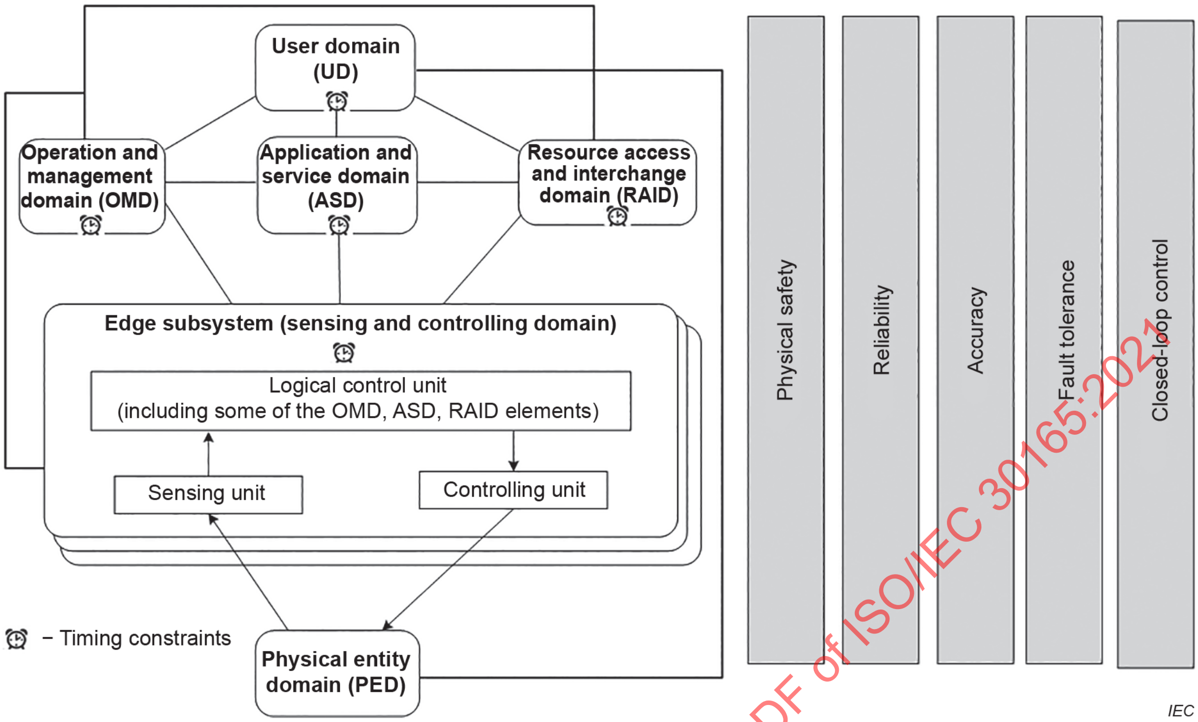


**Figure 4 – RT-IoT system control view**



### 5.3.2 Physical safety

Physical safety is typical in a real-time system. For an RT-IoT system, especially a hard real-time one, strong safety requirements should be in place safeguarding the critical IoT infrastructure and the wellbeing of the workers, users, properties and the environment.

### 5.3.3 Reliability

It is inherent that a real-time system is reliable. It is expected that some problems require deterministic analysis, computation and response. Reliability should be a design criterion for any RT-IoT system.

### 5.3.4 Accuracy

A RT-IoT system with hard timing constraints should satisfy a higher level of accuracy in its results. Any system that interprets and acts on the results should have safeguards against undesirable and unintended physical consequence.

### 5.3.5 Fault tolerance

In an RT-IoT system, fault tolerance means to accomplish every output action of a critical task successfully in spite of component failures. The failures could be of hardware or software. There are many techniques developed over the years, for example, voting scheme, self-checking, recovery methods, surveillance and reconfiguration, etc.

### 5.3.6 Closed loop control

Closed-loop is a basic concept in automation control. Many RT-IoT systems are used in automation industry that employ many closed loops. An RT-IoT system should make use of automation control theories and practices.

A control loop typically involves some sensors observing a real-world process and using the inputs from the sensors to direct one or more actuators to influence or control that real-world process. Sensors could also receive outputs from the IoT system; actuators could also provide sensory information.

Higher layer could also implement advanced control algorithms or data analytic modules to form soft real-time loops. They belong to the OMD and could be realized in the cloud.

## 5.4 Computation view

### 5.4.1 General

Designing an RT-IoT system should follow the practice of designing a real-time system.

A real-time computer system is much different from a general-purpose computer system. The real-time system field has been in existence since the invention of the computer and has matured with thorough theories and methodologies. A successful RT-IoT system should be cognizant of them.

Subclause 5.4 lists the key elements of a real-time computer system. See also Figure 5.

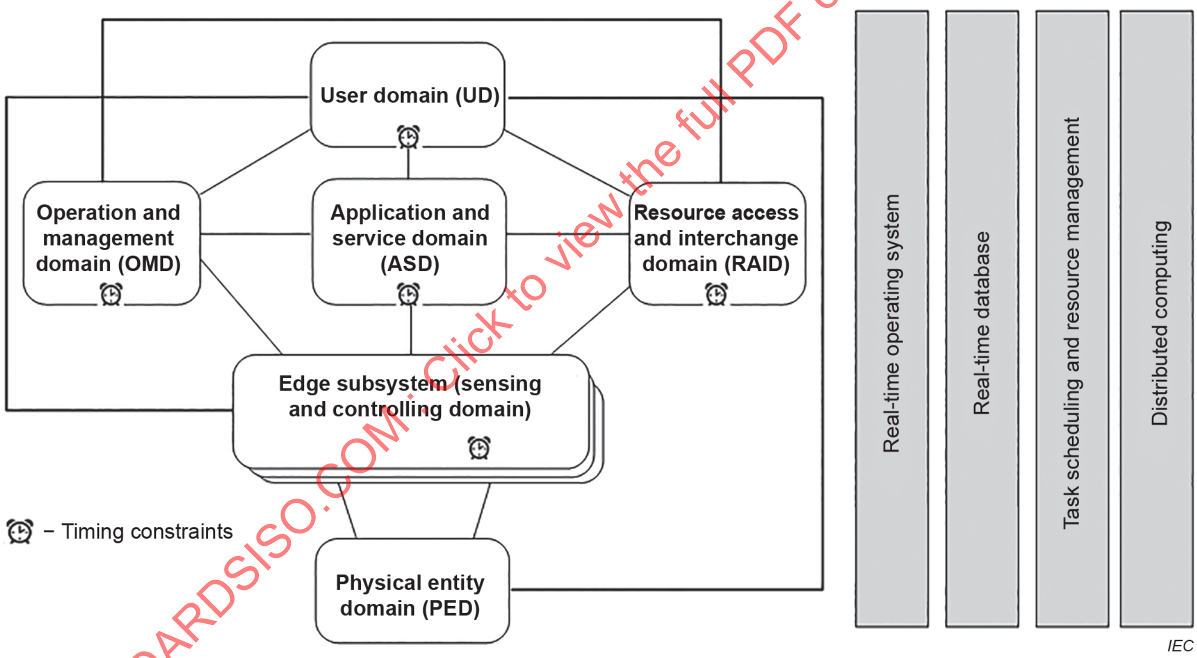


**Figure 5 – RT-IoT system computation view**

### 5.4.2 Real-time task model

A task in an RT-IoT system with real-time constraints should be a real-time task.

A real-time task is modelled with parameters: computation time, deadline, and sometimes the start time. A task could be periodic or aperiodic; parameters such as period, minimum separation time and jitter are also defined. When tasks work together, they often access shared resources, called critical sections.

### 5.4.3 Real-time operating systems

For better support of real-time capability, edge sub-system should consider using a real-time operating system as well as OMD, ASD and ACD.

A real-time operating system is typically portable, componentized, configurable, with small footprint, flexible with priority and pre-emption, and highly reliable.

### 5.4.4 Real-time databases

For better support of real-time capability, real-time database should be used where appropriate in the RT-IoT system.

As systems grow, data management becomes a bigger challenge, so databases come into being. Real-time databases are designed for real-time applications.

In addition to the benefits of a database, a real-time database is different due to the time aspects of the data: not all data are permanent, some are temporal; serializable schedules need to consider temporal correctness; precision sometimes may be traded for timeliness since the latter can be more important.

### 5.4.5 Task scheduling and resource management

A set of real-time tasks in the RT-IoT system should be scheduled in a way so that no timing constraints will be violated.

Real-time scheduling includes single-CPU, multi-CPU, and over networks. Shared resources further complicate the scheduling problem. Scheduling policy could be fixed priority or dynamic priority, pre-emptive or non-pre-emptive, interrupt-driven or time-triggered. The scheduling policies, like dynamic best effort or dynamic planning, are also used for aperiodic tasks.

Real-time task and scheduling has been studied extensively in the real-time systems community.

Transactional real-time tasks are often encountered in an RT-IoT system.

### 5.4.6 Distributed computing

As a system with different edge systems, an RT-IoT system should also coordinate the timing constraints among distributed components. Operating system, database, scheduling, etc. all should be adapted accordingly.

# Annex A
(informative)

## State of the technology

This document defines an RT-IoT system. The technologies underneath an IoT system evolve fairly quickly. The same is true for RT-IoT systems. Annex A lists some of the technology development in 2019 that is beneficial to RT-IoT development.

- 5G: As an enabler 5G provides higher speeds, low-latency, and better Quality of Service (QoS), for real-time applications. To meet timing constraints, the network speed and bandwidth needs to be good enough for any applications. There are already limited RT-IoT applications built with 2G/3G/4G. Many high-demand applications demand latency in microseconds, which becomes possible with 5G. For example, connected cars, online surgery, etc. However, for 5G to truly make those happen, it needs to include real-time support in its infrastructure, for which this document provides the guidelines.
- Smart manufacturing: Smart manufacturing relies on extensive information collected from the factory floor and comprehensive control of the production. A modern factory is an IoT system; a modern factory is also a real-time system.
- Edge computing: Edge computing corresponds to the edge subsystem in this document, although its scope and target differ.
- Time sensitive networking (TSN): There is a joint project of IEC SC 65C/WG 18 and IEEE 802 to define time-sensitive networking profiles for industrial automation. The profiles select features, options, configurations, defaults, protocols, and procedures of bridges, end stations, and LANs to build industrial automation networks.
- Artificial intelligence (AI) and machine learning: AI is being applied to many fields. Applying AI to real-time systems is a new active field.

The following current standard activities also could make use of this document.

- The International Standards being developed in ISO/IEC JTC 1/SC 41. For example, the transport interoperability draft contains a special section on real-time.
- Edge computing.
- Industrial IoT.