# Real-Time Operating System (RTOS) with Different Application: A Systematic Mapping

Sarah Khan

***Abstract* — It is very essential to develop the design and structure of Real time operating system (RTOS) exclusively if we are using this for some special tenders. There any researches believed that (RTOS) are conventional approach to designing devices (RTOS) are used to facilitate to implement different criteria and projects such as time to constrained products, the clustering, the fire alarm system, the wireless sensors powered by renewable energy sensors powered, the stability integrated modular Avionics (IMA) and the alternative programs. In this research, several publications have investigated to perceive the performance of RTOS. These study emphases on a review of RTOS in different criteria and projects to evaluate their performance on different Oss and computing platforms. In this publication data is collected comprehensively factors. In this paper, statistics and results helped in implementation of a more engrossed technique towards the implementation and improvement of RTOS. This paper, highest RTOS are clustering and performance for all applications, white alternate programs measured as least important.**

***Index Terms* — Real time operating system (RTOS), RTOS application, RTOS criteria, Real.**

## I. Introduction

Real time computing systems play vital part in our society. That is why many multifaceted systems are based partially and completely are on the control of computer. There are many example of their applications that have need of real time computing are time constrained products, clustering, fire alarm system are time constrained products, clustering, fire alarm system, wireless sensors, stability, Integrated Modular Avionics (IMI), Wireless Sensors Networks (WSNs), alternate programs.

A robust assurance of a real time system under the performance can be accomplish by using more sophisticated design methodologies. In critical application the control system to able to handle all scenarios, containing extreme burden conditions and its scheme must be controlled by defeatist assumption, on the problem caused by real surroundings [5].

Usually, a real time system is instigated as a conventional of sequence of task that are managed by software called Real time operating system (RTOS). According to the set of constraints every task performs computational activity. So, the objective of Real time operating system (RTOS) is to control or manage assignment of the system, even microprocessor to the task in order to get the objectives [7].
The persistence of this paper to implore the transactions that arise among certain features that affect the functionality of the OS. It will assuredly cover the technique for advancement of further consistent and refined RTOS.

There are four different sections in which the organization of paper is focused. Section II gives background of the RTOS which includes key potential properties of OS. Section III gives the idea about Research Methodology in which Research questions, Search strategy, Inclusive and exclusive criteria, Table I gives review of several existing ROTS Methods and Technique. Section IV shows the results and discussion.

## II. Background

Operating system is the essential part of every single procedural arena for the reason that it maintenances the user to access documents and files of all other programs. Some renowned operating systems are Mac Os, Linux, and Micros of windows and UNIX. For the selection of operating system many techniques are developed and afterwards categories in all-purpose under the region of hardware, interface, security, virtualization and software [2].

Those operating systems which assist real time application requirements also with the ability to process data are the real time operating systems. Examples of real time operating system are operating system for machinery control industrial control, industrial control and scientific instruments [7].

The jitter of a real time operating system is the changeability in the time required by the Os to accept and process an application request. Hard RTOS are the system which has low jitter as well as soft RTOS are the system which has high jitter.

If we talk about time to process data and jitter, there are key potential properties of OS which includes:

-*Reliability and Stability*

The possibility that the OS is not fail or crash.

-*Scalability*

This is the ability of OS to enhance its performance if further properties are added.

-*Availability*

This is the possibility of the OS is actively processing its request and not also in crash mode.

-*Usability*

This is the step the OS has been established in the market.

-*Security*

This is the step that the OS is not disposed to external attacks.

.

Published on January 18, 2021.
Sarah Khan
(e-mail: sarahkhanengg@gmail.com).

*DOI: http://dx.doi.org/10.24018/ejers.2021.6.1.2322*

*-Portability and Clustering*

This is the facility of the OS to shift and / or allocate is operations between cluster of computers.

*-User Interface*

The capability of the OS to relate with the user.

*-Certification*

To provide certain properties the OS has been established.

## III. Research Methodology

### A. Research Questions

TABLE I: Research Questions

| ID | Research question | Rationale |
|---|---|---|
| RQ1 | What are the best tools/Algorithm for RTOS? | It aims to refer to the most commonly used algorithms. Results will support investigators to identify the strong point and weaknesses of algorithms used in RTOS. |
| RQ2 | Which data analysis technique have been used? | It is important to discover which data analysis technique has been used. |
| RQ3 | What are challenges faced by taking different application of RTOS? | In this regard, we discussed the most common challenge that can be answered after taking data from different publications. Outcomes of this inquiry will support researchers to comprehend the challenges. |
| RQ4 | What are the common and widely used methods for checking different criteria of RTOS? | The objective is to take some most common parameters are carefully chosen as establishing the basic on which an operating system will be selected, which are commonly used are reliability, security, portability, availability, usability, clustering and performance, scalability, certification, stability, alternate programs and interface. |

### B. Search Strategy

We conducted this mapping study from different databases and services (ACM Digital Library, Google Scholar, IEEE xplore and Springer Link), to collect the maximum possible number of related research studies in the area of real time operating system different applications.

The searched databases are presented in Table II.

TABLE II: Database Name

| |
|---|
| Google Scholar |
| ACM Digital Library |
| IEEE Xplore |
| Springer Link |
| Frontiers in Big data |

The search study for generation of query is to define search string. In this portion, the papers' keywords, abstracts, and titles were investigated, using the following relations with the combination of Boolean operator "AND", "OR": ("RTOS" AND "Application") OR ("RTOS" AND "clustering").

### C. Inclusion and Exclusion Criteria

In this development, we guarantee that only substantial studies about real time operating system applications are involved for additional examination. It was accomplished by put on an inclusion-exclusion plotting study to regulate whether this study has following all the steps.

*Inclusion criteria:*

- Studies focusing on applications of real time operating system [3].
- Studies discussing key potential properties of RTOS [3].
- Studies focusing on clustering, fire alarm system, wireless sensors [3].
- Studies focusing on the application that require real time computing are time constrained products [3].
- Studies involving the Integrated Modular Avionics (IMI) [3].
- Studies based on surveys are used for systematic mapping of this paper.

*Exclusion criteria:*

- Studies included charters they mostly intensive on hardware on the side [3].
- Studies focusing on representative samples and articles.
- Studies following on Longitudinal study design.
- Studies focusing on application and criteria of real time operating system.

### D. A Relative Study on Several Text Extraction Techniques

Depends on brief overview and survey on applications of real time operating system, this is a relative review which is provided for researches on the basics of technological enhancement and methodologies proposed. Table III shows the different proposed algorithm. These proposed methods are on natural characteristics, recognition accuracy and also error rate percentage.

## IV. Results and Discussions

OS can be characterized into different categorizes by clustering operating systems through numerous approaches which are recorded above. Some most common parameters are designated as establishing the elementary on which an OS will be selected, which are commonly used are reliability, security, portability, scalability, availability, usability, clustering, certification, stability, alternate programs and interface. The motive behind the occurrence of each conditions inside publication studied were firm and analyze numerical values, literature and selected particular criteria. Data were given in few literatures in percentage criteria were used with RTOS. The given data was summed up from different literatures to show the performance of each application in graph.

TABLE III: A RELATIVE REVIEW OF SEVERAL EXISTING ROTS METHODS AND TECHNIQUE

| | Paper | Author | Year | Model | Advantages | Disadvantages |
|---|---|---|---|---|---|---|
| [1] | Evolution analysis of a UAV real-time operating system from a network perspective | Zheng, Guanping XIAO | 2018 | Software network modeling<br>k-core decomposition | Faster computation | Complexity in implementation. |
| [2] | Use of real-time operating systems in the integrated modular avionics | Eugeny Fedosov, Igor Koverninsky, Anna Kan | 2016 | ARINC 429 | It has multiple features like navigation system, GPS and climate control in aircraft which is easier to talk. | Errors cannot be solved automatically we need to solve manually. |
| [3] | Towards an OS for the Internet of Things | Emmanuel Baccelli, Oliver Hahm | 2014 | Monolithic microkernel | Very small can be easily modulated and isolated without touching | If any error occur in the system we have to update the whole operating system. |
| [4] | An Adaptive Scheduler for Real-Time Operating Systems to Extend WSN Nodes Lifetime | Roberto Rodriguez-Zurrunero | 2018 | Duty Cycle Decision Algorithm | Adjusts the obligation cycle during run time without the need of human. | Complexity in implementation |
| [5] | Real-Time Operating System FreeRTOS Application for Fire Alarm Project in Reduced Scale | Luca de Oliveira Turci | 2017 | Arduino IDE platform using the FreeRTOS libraries | Very small, cheap and fast execution | They do not provide multi - tasking.<br>Uses lot of resources |
| [6] | The Formal Design Model of a Real-Time Operating System (RTOS+): Conceptual and Architectural Frameworks | Yingxu Wang<br>Cyprian F. Ngolah<br>Yousheng Tian | 2010 | Process Scheduling | It is quite easier and simple to understand. | The cycle with less execution time endures. |
| [7] | Real Time Operating System Options in Connected Embedded Equipment for Distributed Data Acquisition | Teodor Sumalan<br>Eugen Lupu<br>Radu Arsinte | 2018 | Raspberry Pi 3 | Small in size like credit card.<br>Support all types of codes.<br>Fast processing.<br>Can be used as a portable computer. | Compatibility issue cannot run on multiple platforms.<br>Cannot use until 7 – 8 hours otherwise it will become overheat. |
| [8] | Real Time Operating System (Rtos) For Efficient Multitasking Using Lpc2148 | Prabhakar.B1, Natarajan.V2 | 2015 | LPC2148 | Less power computation.<br>Perform each task at a time so they run faster than other. | Processor performance rely upon the execution, |
| [9] | Memory Management Strategies for Different Real Time Operating Systems | Zafar Ali | 2016 | TLSF,<br>ERMM | Avoid internal fragmentation | Paging, external fragmentation |
| [10] | Automotive Real-time Operating Systems: A Model-Based Configuration Approach | Georg Macher<br>Muesluem Atas<br>Eric Armengaud | 2015 | model-based development | Easy to validate a kernel | Time consuming |

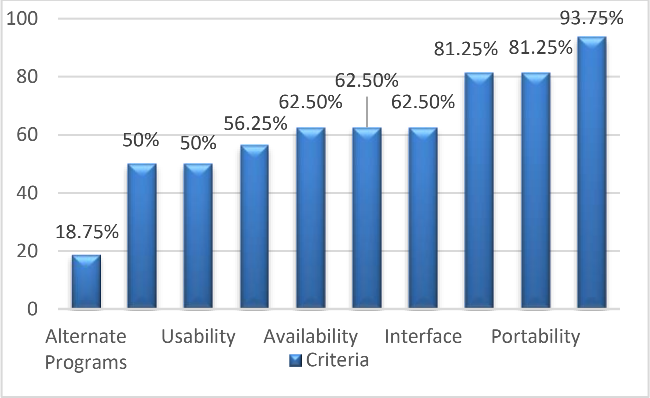

Fig. 1. Play models used RTOS.

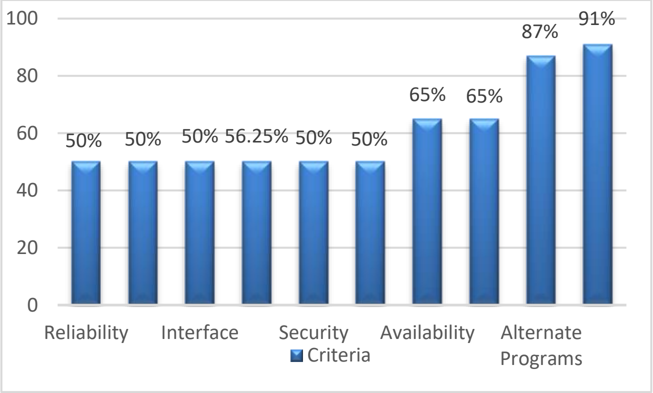

Fig:2. Manufacturing industry used RTOS.

According to graphical representation the percentage criteria in given literature about play models is illustrated in Fig. 1. It was observed from the graphical representation that clustering and performance measure of OS is at highest percent that is 93.75%.

According to the graphical representation the percentage criteria in given literature about manufacturing industries illustrated in Fig. 2. We observed the individual performance of each application using RTOS.

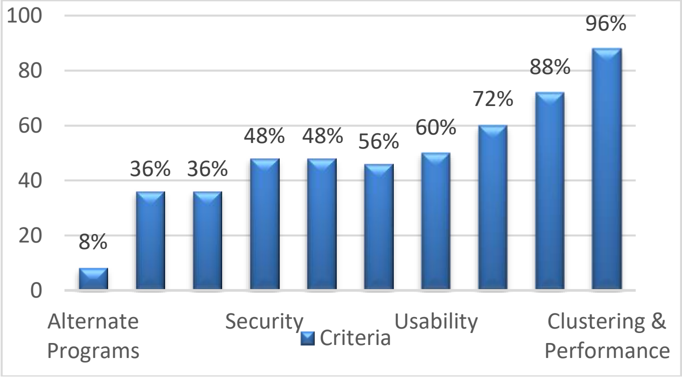

Fig. 3. RTOS used in other application.

Similar behavior observed in Fig. 1 and Fig. 3 regarding clustering and performance which shows highest percentage criteria 96%. However, the percentage criteria towards alternate programs is lowest at 8%.

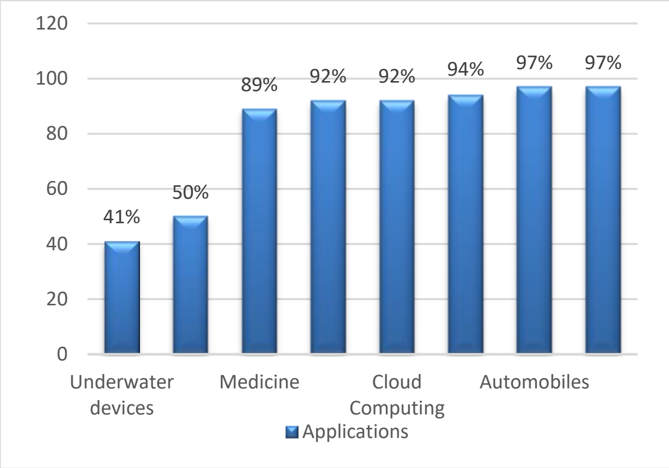

Fig. 4. RTOS used in supercomputing.

According to the graphical representation this application display response for clustering and performance that is 97% which proved that the newly proposed high-performance options.

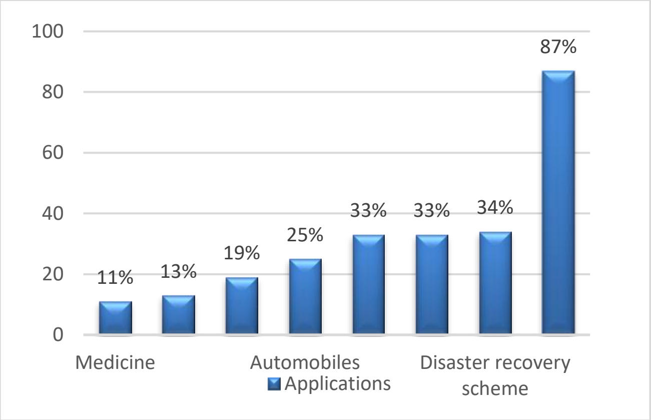

Fig. 5. Alternate programs in different applications.

Alternate programs have been observed that least substantial criteria within the applications. As shown the Fig. 5 it is observed that manufacturing industry is at its highest at 87%.

This paper gives the review about RTOS different applications to analyze their competencies on different computing platforms. We divided operating system into five classes in direction to evaluation the number of procedures so that the effectiveness and usability of each operating system is formed.

## REFERENCES

[1] Zheng, Guanping XIAO, Evolution analysis of a UAV real-time operating system from a network perspective, Chinese Journal of Aeronautics, 2018.

[2] Eugeny Fedosov, Igor Koverninsky, Use of real-time operating systems in the integrated modular avionics, Procedia Computer Science 103, 2017.

[3] Emmanuel Bacilli, Oliver Hahm, Mesut Günes, Matthias Wählisch and Thoma Schmidt, Towards an OS for the Internet of Things, PhD thesis, April 2013.

[4] Roberto Rodriguez-Zurrunero, an Adaptive Scheduler for Real-Time Operating Systems to Extend WSN Nodes Lifetime, Hindawi Wireless Communications and Mobile Computing Volume, 2018.

[5] Luca de Oliveira Turci, Real-Time Operating System FreeRTOS Application for Fire Alarm Project in Reduced Scale, International Journal of Computing and Digital Systems, ISSN (2210-142X), 2017.

[6] Yingxu Wang, the Formal Design Model of a Real-Time Operating System (RTOS+): Conceptual and Architectural Frameworks, International Journal of Software Science and Computational Intelligence, 105-122, 2010.

[7] Teodor Sumalan, Eugen Lupu, Radu Arsinte, Real Time Operating System Options in Connected Embedded Equipment for Distributed Data Acquisition, Carpathian Journal of Electronic and Computer Engineering 11/2 (2018) 35-38, 2018.

[8] Prabhakar.B, Natarajan.V, real time operating system (RTOS) for efficient multitasking using LPC2148, International Journal of Advanced Technology in Engineering and Science, Volume No.03, Issue No. 03, March 2015.

[9] Zafar Ali, Memory Management Strategies for Different Real Time Operating Systems, Hindawi Wireless Communications and Mobile Computing Volume, 2017.

[10] Georg Macher, Muesluem Atas, Eric Armengaud, Automotive Real-time Operating Systems: A Model-Based Configuration Approach, , ICESS, volume 3605 of Lecture Notes in Computer Science, pages 437–442. Springer, 2004.

[11] S. Baskiyar, A Survey of Contemporary Real-time Operating Systems, Informatica 29 (2005) 233–240.

[12] Hussein El Ghor, El-Hadi M Aggoune, Real-time Operating System for Wireless Sensors Powered by Renewable Energy Source, International Journal of Computer Applications (0975 8887), Volume 81 - No. 12, November 2013.

DOI: http://dx.doi.org/10.24018/ejers.2021.6.1.2322