# AN EFFICIENT TOKEN BASED ALGORITHM FOR MUTUAL EXCLUSION IN DISTRIBUTED SYSTEM

*Fahim Kawsar* Shariful Hasan Shaikot** Md. Shahariar Saikat[0]*
*Md. Abdur Razzaque[00] M. A. Mottalib[00]*

## ABSTARCT

***Many distributed computations involving the sharing of resources among various processes require that a resource be allocated to a single process at a time. Therefore, mutual exclusion is a fundamental problem in any distributed computing system. This problem must be solved to synchronize the access to shared resources in order to maintain their consistency and integrity. The major goal of this paper is to get the reader acquainted with a new approach towards the ring based technique for mutual exclusion in a distributed system. An algorithm is proposed based on the idea of generating token by the competing processes to enter the critical section and thus eliminating idle time message passing and reducing communication overhead.***

***Keywords: Mutual exclusion, critical section, token, ring structure.***

## 1. INTRODUCTION

A distributed computing system is a collection of autonomous computing sites that do not share a global or common memory and communicate solely by exchanging messages over a communication facility. In a distributed computing system any given site (also referred to as "node") has only a partial or incomplete view of the total system and a system-wide common clock does not exist. Processes must share common hardware or software resources, cooperating in such a way that they can work in parallel and independently of each other. The access to a shared resource must be synchronized to ensure that only one process is making use of the resource at a given time. The problem of coordinating the execution of critical sections by each process is solved by providing mutually exclusive access in time to the critical section (CS). Each process must request permission to enter its critical section and must release it after it has completed its execution. A mutual exclusion algorithm must satisfy the following requirements [1, 2]:

i. At most one process can execute its critical section at a given time.
ii. If no process is in its critical section, any process requesting to enter its critical section must be allowed to do so at finite time.
iii. When competing processes concurrently request to enter their respective critical sections, the selection cannot be postponed indefinitely.
iv. A requesting process cannot be prevented by another one to enter its critical section within a finite delay.

To simplify, an algorithm must provide mutually exclusive access to the source, ensure deadlock freedom, ensure starvation freedom, and must provide some fairness in the order that requests are granted. The algorithm presented in this paper is based on the token ring approach and satisfies the mentioned requirements in a way that minimize the communication overhead and ensure deadlock freedom, ensure starvation freedom. The

* *Nakajima Lab, Waseda University, Japan.*
** *Dept. of Computer Science, Asian University of Bangladesh, Dhaka, Bangladesh.*
0 *Grameen Software Limited, Dhaka, Bangladesh.*
00 *CIT Dept. Islamic University of Technology, Gazipur-1704, Bangladesh.*