Round Robin Arbiter

Emmanuel Innocent
Department of Electronic and Electrical Engineering
Obafemi Awolowo University, Ile-Ife, Nigeria
sherlockhudep7@gmail.com

***Abstract* — In a social setting where some shared resources are limited and cannot be used simultaneously by more than one user, there is a need for an effective system to decide how the shared resources are to be used. In addition, the need for such a system is vital where fairness in the share of the resources matters. This is a similar problem in the realm of electronics and computing where resources like bus and memory need to be accessed by a number of devices and users. There is a need for a system to decide on which device gets access at a given time. This system is called an "arbiter". This paper presents the design of an arbiter called the Round Robin Arbiter using Verilog HDL.**

***Keywords*—Round Robin Arbiter , Finite State Machine, Starvation-free Arbiter**

I. DESCRIPTION

The Round Robin Arbiter uses a scheduling algorithm that ensures requests from users or devices are served or granted without starving other requests of the shared resource. This starvation free service is achieved by allocating a fixed time slice called time quanta to each request queues while going round serving each queue in cycles. Of course the choice of the time quanta is critical, it must be chosen to prevent the starvation of requests.
Highlighted below is the general features that makes the round robin scheduling algorithm a good arbiter:

- It is simple to implement
- It is starvation free
- It is relatively fair (equal chance of accessing the shared resource)

II. THE DESIGN

In this design the arbiter is implemented as a four state **high level Finite State Machine (FSM)** with states: S0, S1, S2, and S3. In each state a ring counter is updated. This ring counter serves as a token. Since this design is intended to serve only four request queues the ring counter(token) will be a four bit counter with three 0s and one 1s (e.g 0001). The state transition diagram of this FSM is shown in fig.1 and the logic diagram of the design in fig.2. In each state the output variable grant(Gnt) is a mask of the input request(Req) and the ring counter(Cnt) variable, i.e

$$Gnt[3:0] = Req[3:0] \ \& \ \text{Cnt}[3:0] \qquad (1)$$

A grant output bit of value "1" means the corresponding request bit has been granted access to the resource while a value of "0" means no access granted.

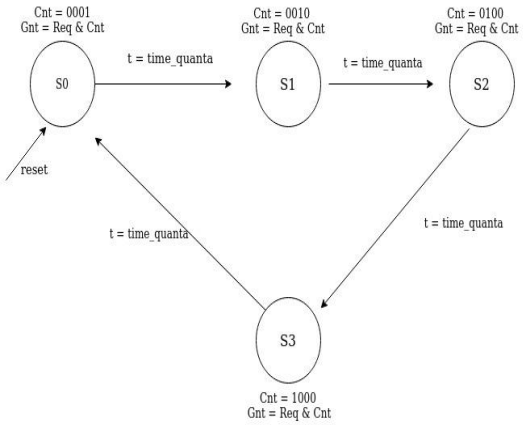

Fig. 1. State transition diagram

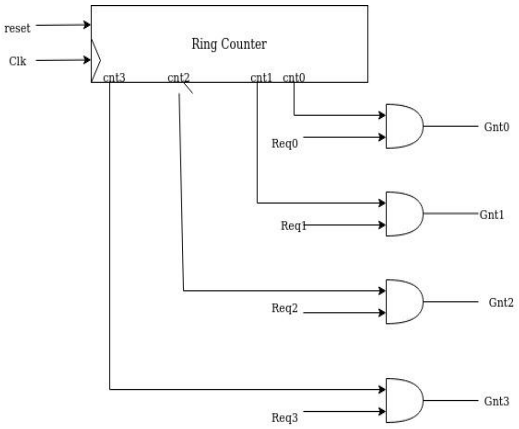

Fig. 2 A Simple Round Robin Arbiter Logic Diagram.

BIBLIOGRAPHY/REFERENCES

Implementation of Round Robin Arbiter using Verilog, K. Tharun Teja, et al. International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering (An ISO 3297: 2007 Certified Organization) Vol. 4, Issue 10, October 2015.