Mixed Round Robin Scheduling for Real Time Systems

Pallab Banerjee[1], Biresh Kumar[2] , Probal Banerjee[3]

[1,2,3]*Assistant Professor*
[1,2]*Department of Computer Science and Engineering.*
[3]*Department of Electronics and Telecommunication Engineering.*
[1,2]*Amity University Ranchi , Jharkhand ,India.*
[3]*Government Polytechnic Ranchi, Jharkhand, India.*

Abstract:*-Round Robin Scheduling algorithm is designed especially for time sharing Operating system and Real Time System .Jobs get the CPU for a fixed time (quantum time or time slice) in Round Robin Scheduling. It is Similar to FCFS (first come first serve scheduling),but with preemption (that is CPU interrupted at regular intervals). Ready queue is treated as a circular buffer. Process may use less than a full time slice. If process is incomplete at the end of time slice, they join in the end of ready queue. The Round Robin Scheduling algorithm has its disadvantages that is its longer average waiting time, higher context switches, higher turnaround time .In this paper a new algorithm is presented called Mixed Round Robin scheduling algorithm .Mixed Round Robin Scheduling is the mix-up of three Scheduling algorithm i.e (MIN-MAX,AVG-MAX and BPRR) .In this scheduling algorithm the main idea is to adjust the time Quantum dynamically so that this algorithm perform better performance than simple Round Robin scheduling algorithm.*

Keywords*-Operating System, Round Robin, Turnaround time, Waiting time, Context Switch.*

I. INTRODUCTION

Operating system is system software which makes an interface between end user and computer hardware, so that the user can handle the system in a convenient manner. Scheduling is the most repetitively used fundamental concept in OS. In multitasking and multiprogramming environment it is necessary to choose the process among the number of process present in the job pool according to their need. Allocation of CPU to the processes is done by scheduler, which operated by some scheduling algorithms. FCFS, SJF, Priority & RR are different type of scheduling algorithms. From these entire algorithm Round Robin algorithm is the most popular non-preemptive scheduling algorithm which works on Real time system also. In non-preemptive scheduling algorithm, CPU is assigned to a process until its execution is completed. But in preemption, running process is forced to release the CPU by the newly arrived process.

II. CPU SCHEDULING ALGORITHMS

There are many scheduling algorithm are designed like FCFS, SJF, PRIORITY, ROUND ROBIN etc. In the First-Come-First-Serve (FCFS) algorithm, process that arrives first is immediately allocated to the CPU based on FIFO policy. In Shortest Job First (SJF) algorithm, process having shortest CPU burst time will execute first. If two processes having same burst time and arrive simultaneously, then FCFS procedure is applied. Priority scheduling algorithm, provides priority (internally or externally) to each process and selects the highest priority process from the ready queue. In case of Round Robin (RR) algorithm, time interval of one time quantum is givento each process present in the circular queue.

III. PERFORMANCE METRICS

The proposed algorithm is designed to meet all scheduling criteria such as maximum CPU utilization, maximum throughput, minimum turnaround time, minimum waiting time and context switches. Here we are considering the performance criteria in each case of our experiment.

Turnaround Time (TAT)=Finish Time–Arrival Time. Average Turnaround Time should be less.

Waiting Time (WT)= Start Time- Arrival Time. Average Waiting Time should be less.

Context Switch (CS)=Switching between the processes. The number of context Switch should be less.

IV. RELATED WORK

In the last few years different approaches are used to increase the performance of Round Robin scheduling like High performance Round Robin (HPRR), Even-Odd Round Robin(EORR), Average Max Round Robin Algorithm (AMRR),Mid-Average Round Robin Scheduling(MARR),Min-Max Round Robin (MMRR),Adaptive Round Robin Scheduling using Shortest Burst Approach Based on Smart Time Slice, Multi-Dynamic time Quantum Round Robin (MDTQRR), Self-Adjustment Time Quantum in Round Robin (SARR),Dynamic Quantum with Re-adjusted Round Robin (DQRRR),Best Performance Round Robin(BPRR),Optimal Performance Round Robin(OPRR),Generic Round Robin(GRR).

III. PROPOSED ALGORITHM

1. Initialize

CS=0, AWT=0, ATT=0, i=0,S1=0,c1=0,j=1,n=100,S2=0,c2=0,p[n], TimeQuantum[5], sum=0

2. Sort the process in the ready Queue in ascending order of their BT.

//n = number of processes and i = loop variable

3. while (RQ != NULL)

//RQ = Ready Queue

a). TQ_{MM} = MAXBT – MINBT

//TQ_{MM} = Time Quantum

//MAXBT = MAXimum Burst Time

//MINBT = MINimum Burst Time (Remaining burst time of the processes)

4. If one process is there then TQ_{MM} is equal to

BT of itself

5. TimeQuantum[ ]=TQ_{MM}

6. Calculate the average of BT of the last three processes and put them in Avg.

7.For (i=N-2 to N loop)

{

a) sum=sum+ BT(Pi);

}

8. Avg=sum/3;

//Avg=Average of last three process BT.

9. TQ_{BPRR} = Avg;

10. TimeQuantum[]=TQ_{BPRR}

//end of for

11. If two processes are there then the TimeQuantum[ ]= is equal to the average of BT of the two processes.

12. If one process is there then after calculation TQ is equal to BT itself

i.e TimeQuantum[ ]=BT.

13. while(RQ!=NULL)

//MAXBT=Maximum Burst Time

14. AVG= (Sum of BT of all the process)/Number of processes.

//(Use round off function in AVG.)

15. TQ1=(AVG+MAXBT)/2

16. TimeQuantum[ ]=TQ1

//(Use round off function in TQ1) (Remaining Burst time of the process)

17. If one process is there then TimeQuantum= BT itself

18. Large_TQ=TimeQuantum[1];

19.Repeat for i=1 to 5-1

{

a)If(TimeQuantum[i]>Large_TQ)

Then

b) Large_TQ=TimeQuantum[i];

}

20. Repeat for i=1 to N loop

// Assign Large_TQ to (1 to N)processes.

{

a) Pi->Large_TQ

//Assign TQ to all the available processes.

}

//End of for.

21. Calculate the remaining Burst time of the processes.

22. If (new process arrived and BT!=0)

then go to step1

else

23. if (new process is not arrived and BT!=0)

then go to step 2

24. if (new process is arrived and BT != 0)

then go to step 1

else

25. Calculate ATT, AWT and CS.

//ATT = Average Turnaround Time /

/AWT = Average Waiting Time

//CS = number of Context Switches

26. End

IV. EXPERIMENTAL ANALYSIS

Case 1: Let's consider five processes with Burst time (P1=18, P2=34, P3=49, P4=64, P5=78) with Arrival Time =0 as shown in the Table 1. Table 2 shows the output using RR , EORR and MIXED algorithms. Figure1,2,3 shows the Gantt chart of both RR , EORR and MIXED algorithm respectively.

Table1.Process with Burst Time

Processes	Arrival Time	Burst Time
P1	0	18
P2	0	34
P3	0	49
P4	0	64
P5	0	78

Table 2: Comparison between RR algorithm, EORR algorithm and new proposed MIXED Round Robin algorithm.

Algorithm	Time Quantum	Turnaround Time	Average Waiting Time	Context Switch
RR	20	155.2	107.2	13
EORR	49,29	125.6	77	7
MIXED	64,14	115.8	67.2	6

TQ=20

P1	P2	P3	P4	P5	P2	P3	P4	P5	P3	P4	P5	P4	P5

0 18 38 58 78 98 112 132 152 172 181 201 221 225 243

Fig.1: Gantt chart of RR from Table 1 of CASE 1.

TQ=49 TQ=29

P1	P2	P3	P4	P5	P4	P5

0 18 52 101 150 199 214 243

Fig.2: Gantt chart of EORR from Table 1 of CASE 1.

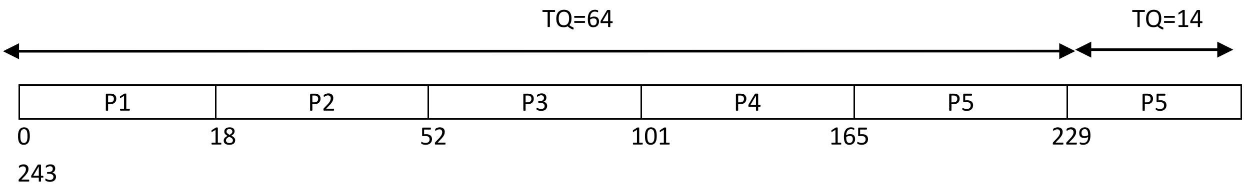

Fig.3: Gantt chart of MIXED from Table 1 of CASE 1.

Case2: Let's consider five processes with Burst time (P1=42, P2=43, P3=44, P4=45, P5=46) with Arrival Time =0 as shown in the Table 3 . Table 4 shows the output using RR , EORR and MIXED algorithms. Figure 4,5,6 shows the Gantt chart of both RR , EORR and MIXED algorithms respectively.

Table3.Process with Burst Time

Processes	Arrival Time	Burst Time
P1	0	42
P2	0	43
P3	0	44
P4	0	45
P5	0	46

Table 4: Comparison between RR algorithm, EORR algorithm and new proposed MIXED Round Robin algorithm

Algorithm	Time Quantum	Turnaround Time	Average Waiting Time	Context Switch
RR	20	210	166	15
EORR	44,2	155.8	111.8	7
MIXED	45,1	130	86	5

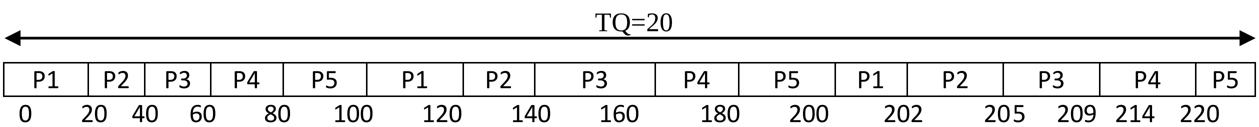

Fig.4: Gantt chart of RR from Table 3 of CASE 2.

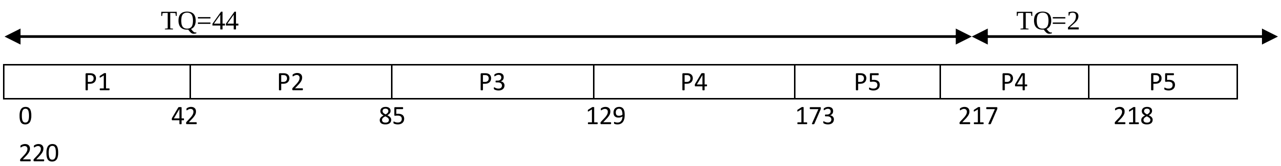

Fig.5: Gantt chart of EORR from Table 3 of CASE 2.

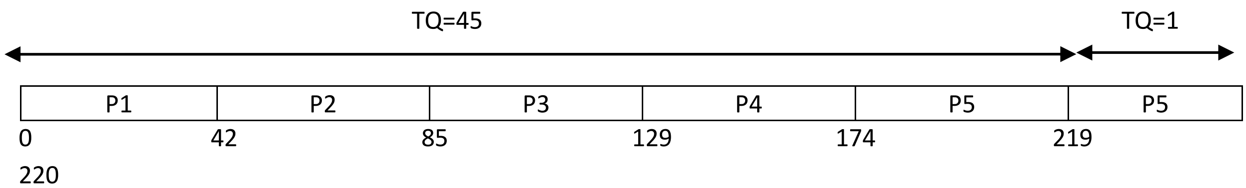

Fig.6: Gantt chart of MIXED from Table 3 of CASE 2.

Case3: Let's consider five processes with Burst time (P1=11, P2=22, P3=44, P4=88, P5=166) with Arrival Time =0 as shown in the Table 5 . Table 6 shows the output using RR , EORR and MIXED algorithms. Figure 7,8,9 shows the Gantt chart of both RR , EORR and MIXED algorithms respectively.

Table5.Process with Burst Time

Processes	Arrival Time	Burst Time
P1	0	11
P2	0	22
P3	0	44
P4	0	88
P5	0	166

Table 6: Comparison between RR algorithm, EORR algorithm and new proposed MIXED Round Robin algorithm

Algorithm	Time Quantum	Turnaround Time	Average Waiting Time	Context Switch
RR	20	167.4	101.2	19
EORR	74,93	138	72	7
MIXED	116	123.4	57.2	5

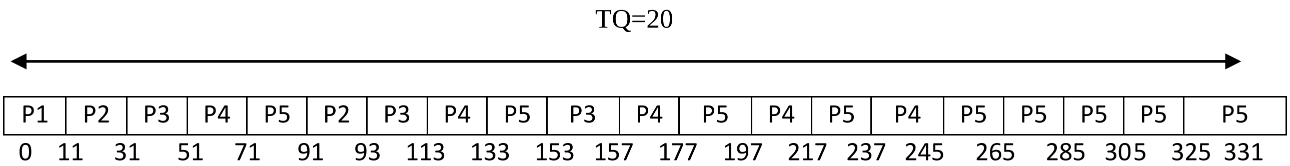

Fig.7: Gantt chart of RR from Table 5 of CASE 3.

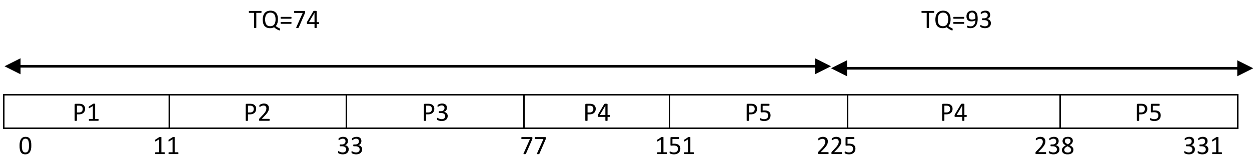

Fig.8: Gantt chart of EORR from Table 5 of CASE 3.

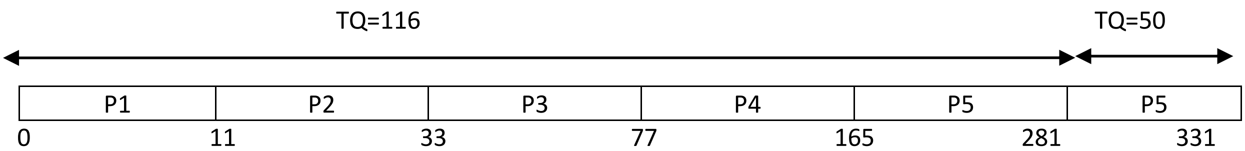

Fig.9: Gantt chart of MIXED from Table 5 of CASE 3.

Case 4: Let's consider five processes with Burst time (P1=14, P2=28, P3=39, P4=48, P5=62) with Arrival Time (P1=0,P2=9,P3=11,P4=14,P5=18) as shown in the Table 7 . Table 8 shows the output using RR , EORR and MIXED algorithms. Figure 10,11,12 shows the Gantt chart of both RR , EORR and MIXED algorithms respectively.

Table7.Process with Burst Time

Processes	Arrival Time	Burst Time
P1	0	14
P2	8	28
P3	10	39
P4	14	48
P5	18	62

Table 8: Comparison between RR algorithm, EORR algorithm and new proposed MIXED Round Robin algorithm

Algorithm	Time Quantum	Turnaround Time	Average Waiting Time	Context Switch
RR	20	109.4	71.2	11
EORR	38,13,11	104.2	66	8
MIXED	50,12	81.4	43.2	6

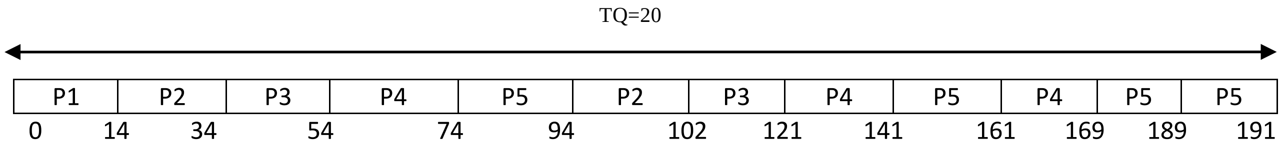

Fig.10: Gantt chart of RR from Table 7 of CASE 4.

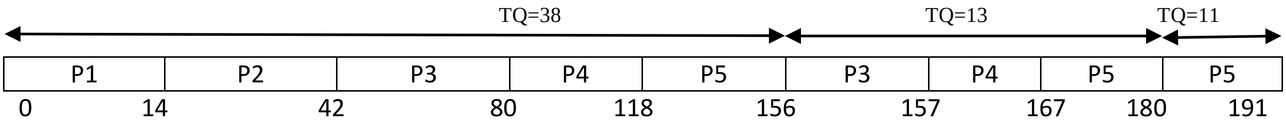

Fig.11: Gantt chart of EORR from Table 7 of CASE 4.

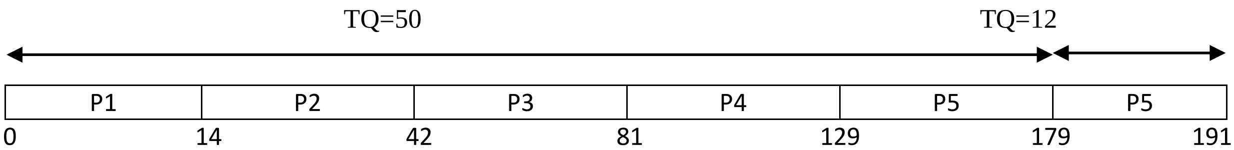

Fig.12: Gantt chart of MIXED from Table 7 of CASE 4.

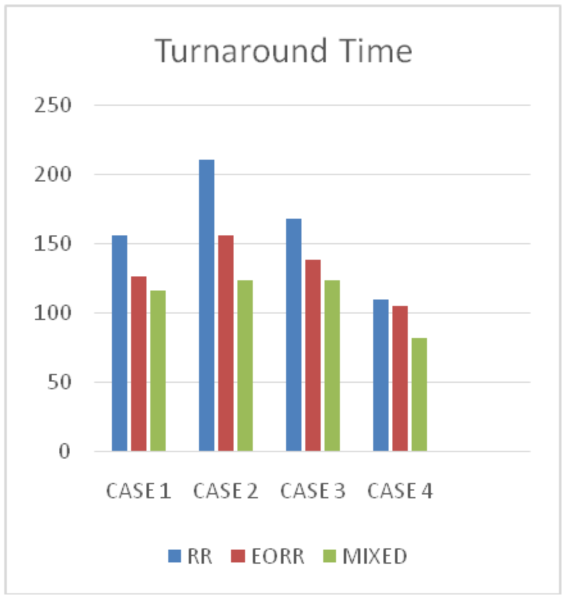

Fig.13: Comparison of average Turnaround Time of RR , EORR and MIXED RR taking arrival time into consideration.

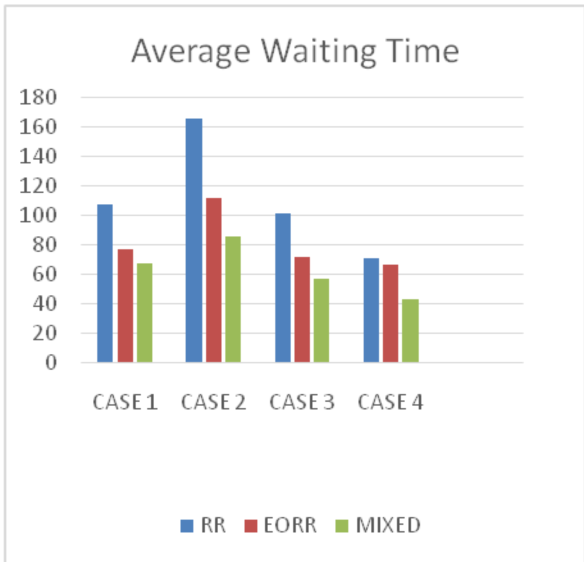

Fig.14: Comparison of average Waiting Time of RR , EORR and MIXED RR taking arrival time into consideration.

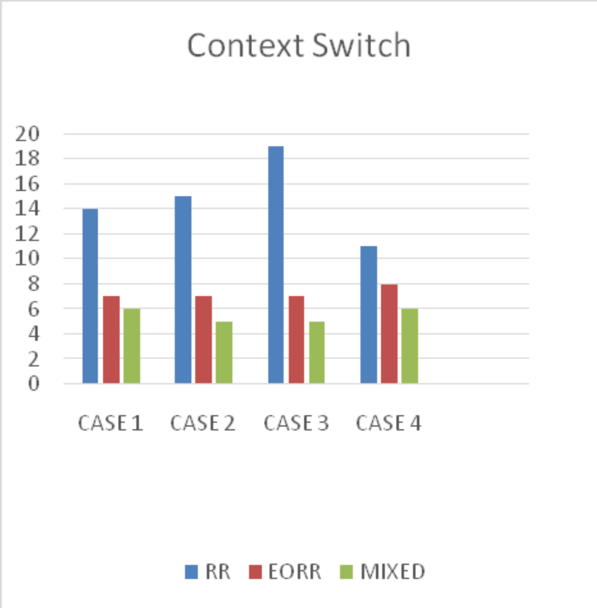

Fig.15: Comparison of Context Switching of RR , EORR and MIXED RR taking arrival time into consideration.

V. CONCLUSION

CPU is one of the most important components of the computer resources. CPU scheduling involves careful examination of waiting processes to determine the most efficient way to service the request. In this paper an Mixed round robin scheduling algorithm is proposed. This proposed CPU scheduling algorithm will give better performance than RR and EORR and that is achieved by increasing Time Quantum Dynamically and decreasing the total Turnaround Time, Average Waiting Time and Number of Context Switching.

REFERENCES

[1] "Silberschatz, A., P.B. Galvin and G. Gagne, 2008"Operating Systems Concepts. 7th Edn., John Wiley and Sons, USA, ISBN: 13: 978-0471694663 , pp: 944.

[2] Pallab banerjee, probal banerjee, shweta sonali dhal, " Comparative Performance Analysis of Average Max Round Robin Scheduling Algorithm (AMRR) using Dynamic Time Quantum with Round Robin Scheduling Algorithm using static Time Quantum ",IJITEE, ISSN: 2278-3075, Volume-1, Issue-3, August 2012.

[3] "Tanebaun, A.S., 2008" Modern Operating Systems. 3rd Edn., Prentice Hall, ISBN: 13:9780136006633, pp: 1104.

[4] Pallab banerjee, probal banerjee, shweta sonali dhal, "Performance Evaluation of a New Proposed Average Mid Max Round Robin (AMMRR) Scheduling Algorithm with Round Robin Scheduling Algorithm",IJARCSSE,ISSN:2277-128X, Volume-2, Issue- 8, August 2012.

[5] Pallab banerjee, probal banerjee, shweta sonali dhal,"Comparative Performance Analysis of Even Odd Round Robin Scheduling Algorithm (EORR) using Dynamic Time Quantum with Round Robin Scheduling Algorithm using static Time Quantum" IJARCSSE,ISSN: 2277-128X,Volume-2, Issue-9, August 2012.

[6] Pallab banerjee, probal banerjee, shweta sonali dhal,"Improved High Performance Round Robin Scheduling Algorithm(HPRR)using Dynamic Time Quantum" International Journal of Computer Information System,ISSN: 2277-128X,Volume-5, No-3, 2012.

[7] Sarojhiranwal and D.r. K.C.Roy"Adaptive Round Robin Scheduling using Shortest Burst Approach Based on Smart Time Slice".volume 2,No. 2,July-Dec 2011,pp. 319-32.

[8] Pallab banerjee, probal banerjee, shweta sonali dhal, "Comparative Performance Analysis of Mid Average Round Robin Scheduling Algorithm (MARR) using Dynamic TimeQuantum with Round Robin Scheduling Algorithm having static Time Quantum",IJECSE,ISSN: 2277- 1956, Volume-1,Issue-4, August 2012.

[9] H. S. Behera, R. Mohanty, and D. Nayak, "A New Proposed Dynamic Quantum with Re-Adjusted Round Robin Scheduling Algorithm and Its Performance Analysis," vol. 5,no. 5, pp. 10-15, August 2010.

[10] Sanjay Kumar Panda and Saurav Kumar Bhoi, "An Effective Round Robin Algorithm using Min-Max DispersionMeasure" ISSN : 0975-3397 ,Vol. 4 No. 01, January 2012.

[11] Tarek Helmy, Abdelkader Dekdouk " Burst Round Robin: As a Proportional-Share Scheduling Algorithm", IEEE Proceedings of the fourth IEEE-GCC Conference on towards Techno-Industrial Innovations, pp. 424-428, 11-14 November,2007.

[12] Yaashuwanth .C & R. Ramesh "Inteligent time slice forround robin in real time operating system", IJRRAS 2 (2), February 2010.

[13] R. J. Matarneh, " Seif-Adjustment Time Quantum in Round Robin Algorithm Depending on Burst Time of the Now Running Proceses ", American Journal of Applied Sciences 6 (10),pp. 1831-1837, 2009.

[14] H. S. Behera, Rakesh Mohanty, Sabyasachi Sahu and Sourav Kumar Bhoi. "Comparative performance analysis of multidynamic time quantum round robin (mdtqrr) algorithm with arrival time", ISSN : 0976-5166, Vol. 2, No. 2,Apr-May2011,pp.262-271.

[15]Pallab banerjee ,Prof Dr L.N.Padhy. "Comparative analysis of Maximum performance round robin(MPRR) by Dynamic Time Quantum with static time quantum",ISSN:2277-128X,pg.372-377Vol-4,Issue-11,Nov-2014.

[16] Pallab banerjee,Talat Zabin,Shweta Kumari,Pushpa Kumari "Comparative performance analysis of best performance of round robin scheduling algorithm(BPRR) using Dynamic Time quantum with priority based round robin(PBRR) CPU Scheduling algorithm in Real Time System".ISSN:2277-1956,Vol-4,Number-2, pg. 151-159,May 2015.

[17] Pallab banerjee,Riya Shree,Richa kumara Verma"Generic Round Robin Scheduling for real time systems(GRR)" ISSN:2277- 128X,pg.148 -155 Vol-7,Issue-5,May-2017