INTERNATIONAL JOURNAL OF RESEARCH IN COMPUTER APPLICATIONS AND ROBOTICS
ISSN 2320-7345

# SIMULATION FOR REPRESENTING THE WORK OF PROCESS CONTROL BLOCK (PCB)

**Maha A. Sayal**

*Computer Science Department, University of Thi-Qar, Iraq*
*maha.sayal@yahoo.com*

**Abstract: -** The basic aim of CPU keeps its busy at all time. In fact, there is only one process runs in CPU at a time, a process is inherently program after convert it to executable file loaded into memory [1] .when process requests I/O, it will leave CPU, and seeks I/O file or device, in this time CPU is switching from current process to another process to come true max of CPU utilization, user don't observe that when he uses computer. This paper illustrates role of PCB by representing it as a simulation system using visual basic programming language and performs it as a program and explain what happen in CPU when switch from process to another process.

**Keywords:** CPU, Executable file, PCB, Simulation System.

## 1. Introduction

CPU considers a brain of computer, it performs all computation and logical operations, CPU must always be busy as possible [2]. Each process contains block has all information about process, as shown in Figure.1.

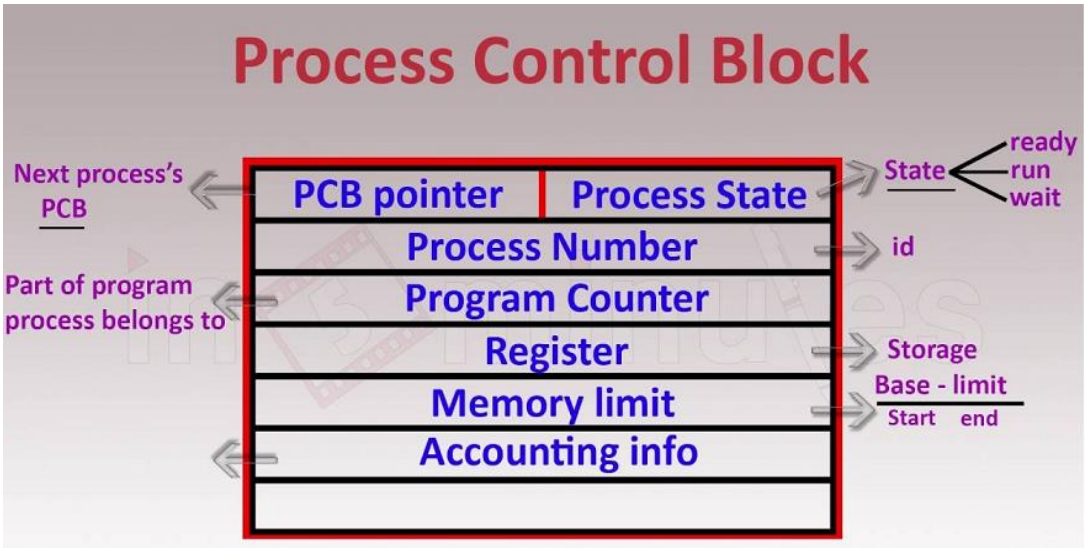


**Figure.1:** PCB

In figure above. PCB contains pointer to indicate PCB for next process, process state its mean that process in ready, run, or in wait, ID of process, program counter to locate the next instruction, register, base and limit register to bound the address that is stored in memory and accounting information, when process is out CPU, PCB will save the state of process [3]. Process may be either in short-term scheduler or in long-term scheduler, process in short-term scheduler mean that process is ready to run in CPU while process in long-term scheduler mean that process is bring from hard disk to main memory[4]. The process is transferring from long to short scheduler and after that to CPU, it can return to short scheduler when interrupt happens [5].

Interrupt is signal send to CPU when process requests I/O file or device, Operating system sends to CPU special operation by software called system call , and CPU take action by switching to other process to make CPU busy at all time[6] .

## 2. Proposed Work

A simulation system is proposed to represent this work, first must explanation switching in processes during run time. This example about two processes, the process P0 is executing but when interrupt is happened , PCB0 for process p0 will save state and PCB1 load the saved state for process1, thus until finish performing of processes[7] , as shown in Figure.2.

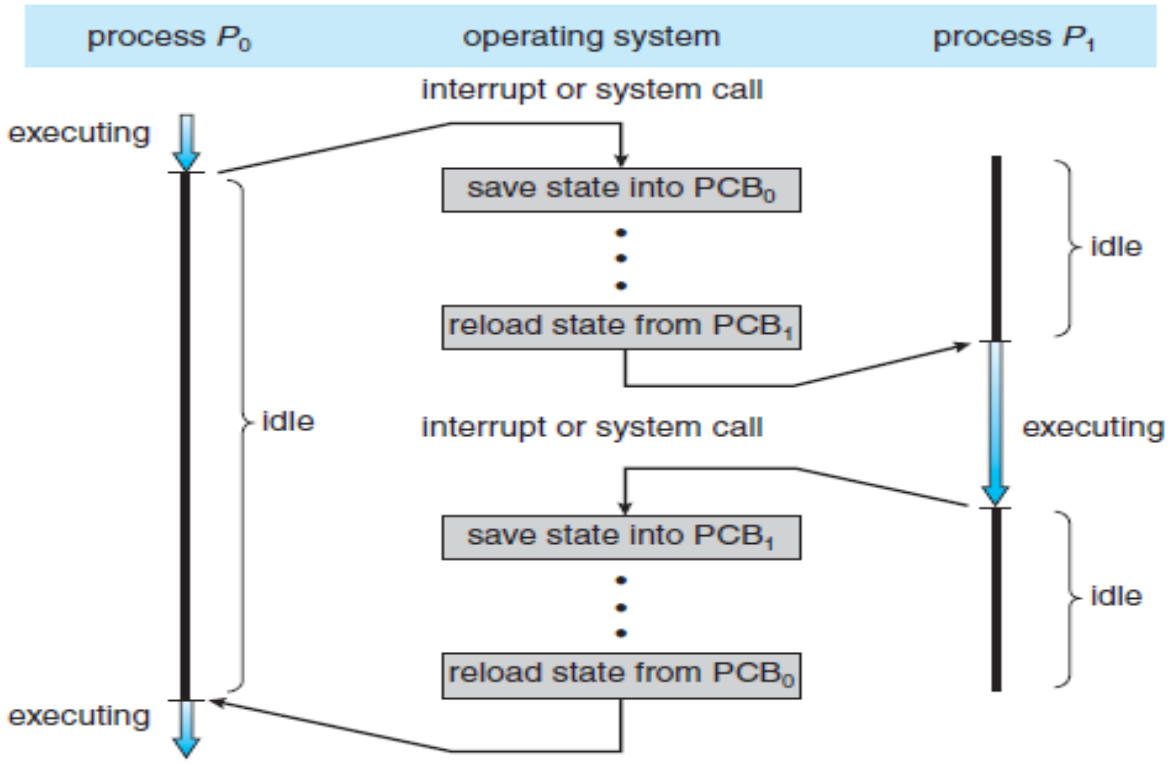


**Figure.2:** CPU transfers from process to another process

Proposed work is to represent these actions using programming language and make these actions invisible to user; he can see all them sequentially. When process inputs as a symbols to screen, this will mean that process enters to system, interrupt represent as a counter (total of bust time for all processes) when it decrements by 1 microsecond, the interrupt will happen. PCB represents as a text to explain what PCB do. In this proposed work, round robin is used as CPU scheduling algorithm .RR is used quantum number (q) to order entering processes to CPU, here a slot time do like ( q) to divide the time of CPU between processes , this algorithm use in time sharing system[7].

## 3. Methodology

In this work, Visual Basic is used as Programing language to perform a simulation system to proposed PCB. Because this language contains tools, properties and codes necessary to build system explain the work, as shown in Figure. 3.

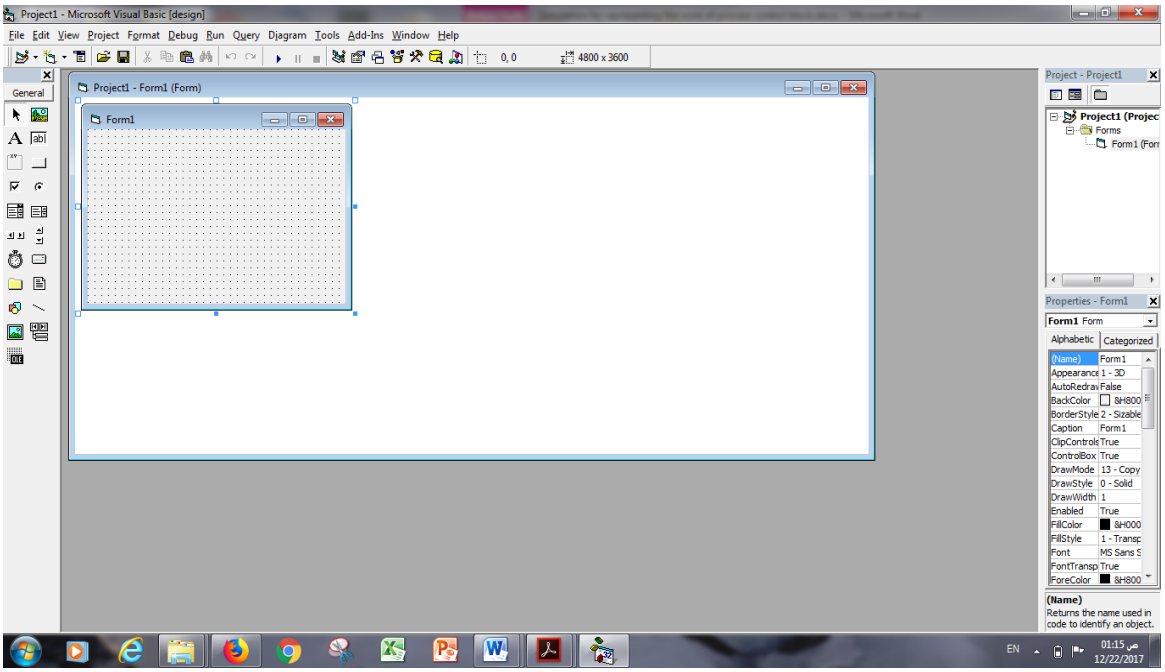


**Figure. 3:** Interface of visual basic

Tools such as form, text, label, commands are used to represent work of CPU, each process is represented by two texts one for entering bust time of process and other for explaining perform phases each microsecond for process, as shown in Figure. 4.

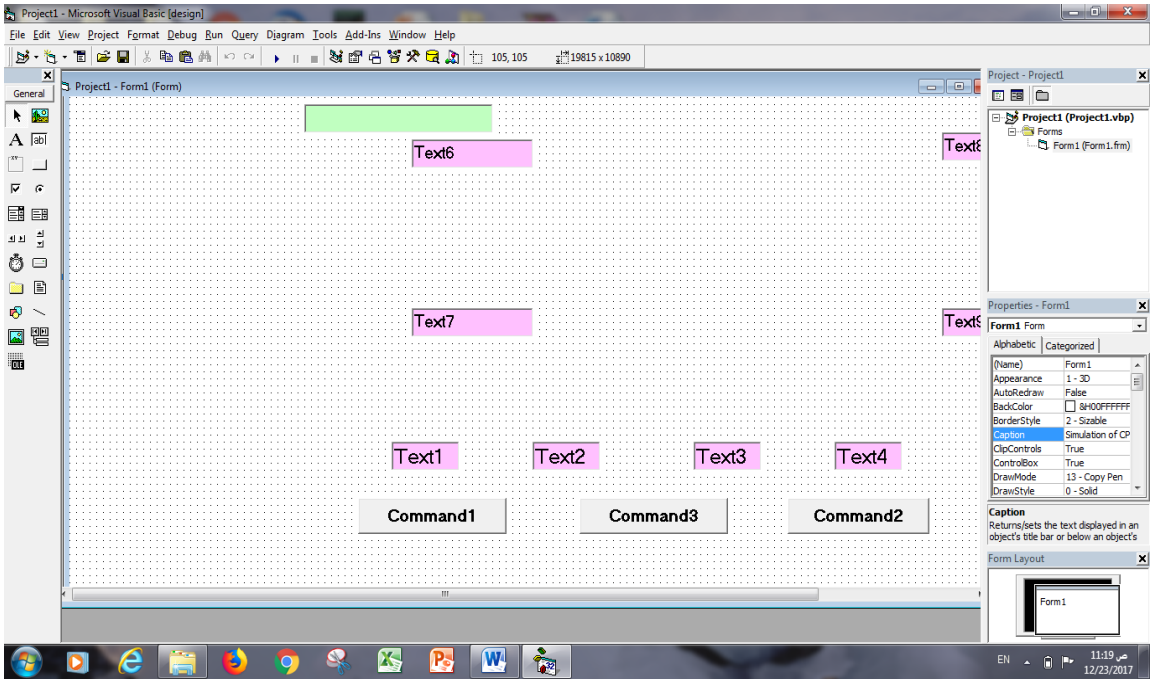


**Figure. 4:** Used Tools

Texts and labels represent how to save and load PCB for processes, while commands represent action that the CPU takes place to switch from process to another process. Interrupt in CPU represent as slot time (1micosecond) to perform system call send to CPU from kernel of operating system. The works of CPU finish after last micro second for last process, as shown in Figure.5.

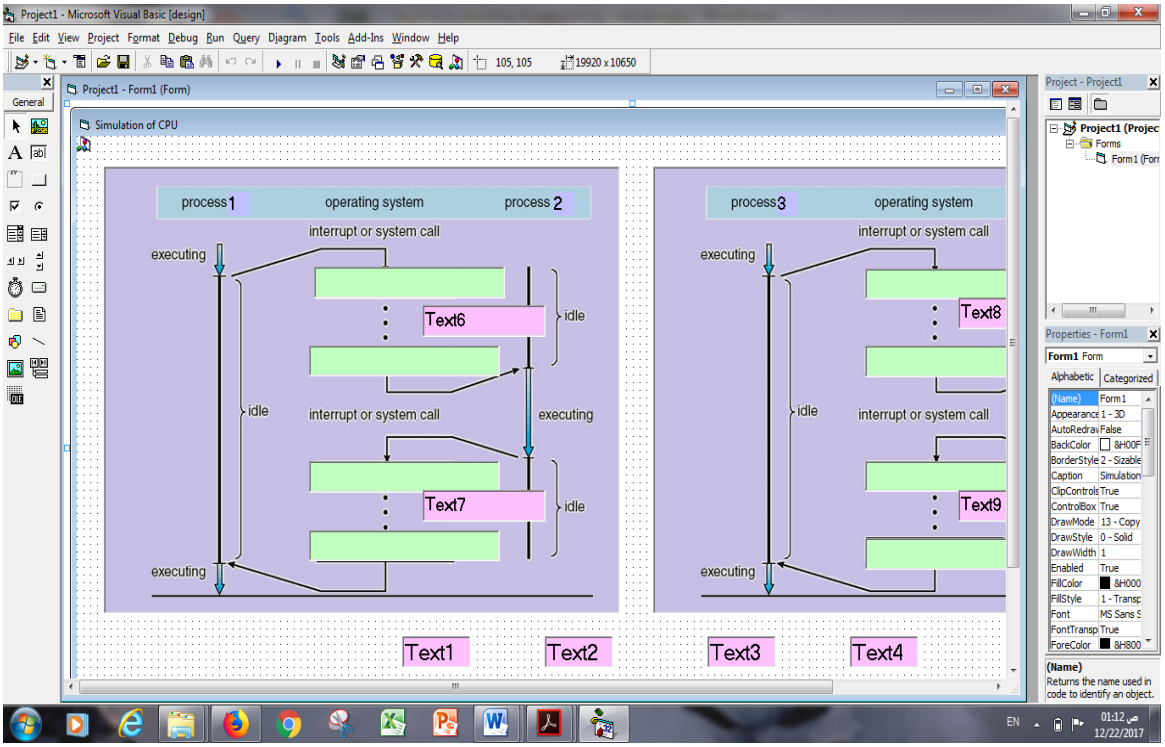


**Figure. 5:** Mechanical of work

After finishing from programing and used a proper tools, the final style of simulation system complete and user can test this system, as shown in Figure.6.

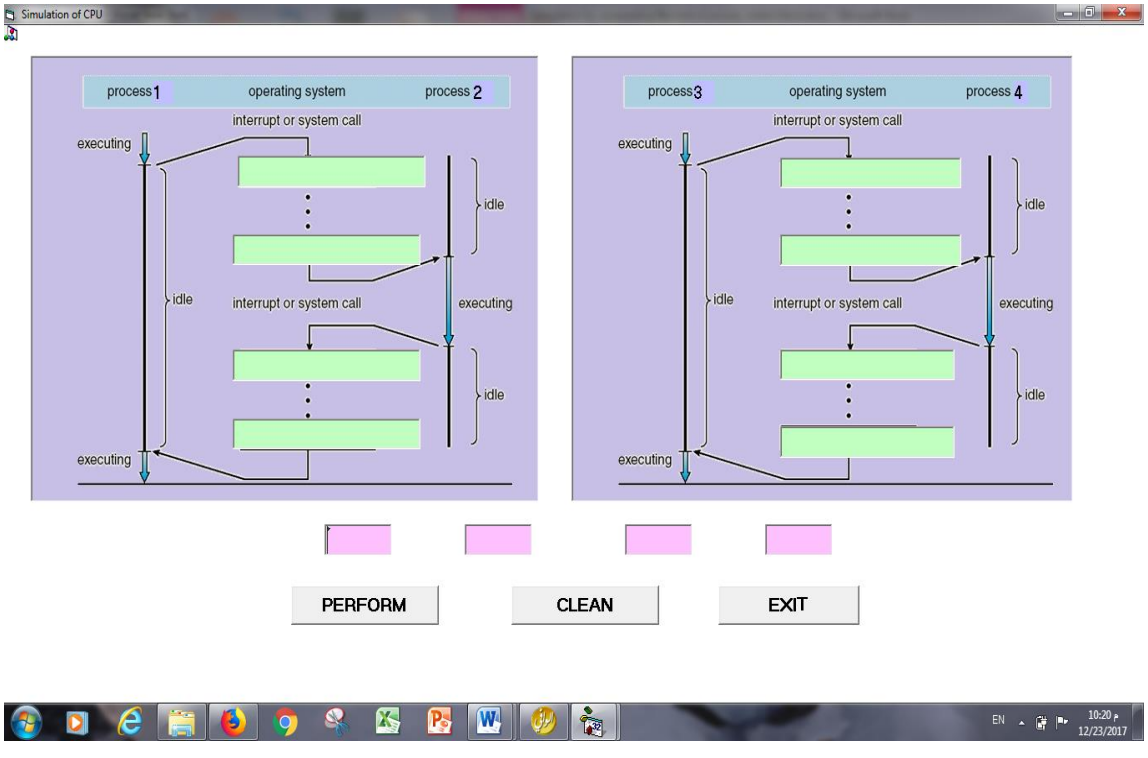


**Figure.6:** The final style of simulation system

## 4. Results

To test the system, the first step click on perform command, message of welcome is seen on screen , after that appears input box to enter the number of processes (In test we chose 4 processes), which means the processes in long-term scheduler, then it will request burst time of each process, as shown in Figure.7.

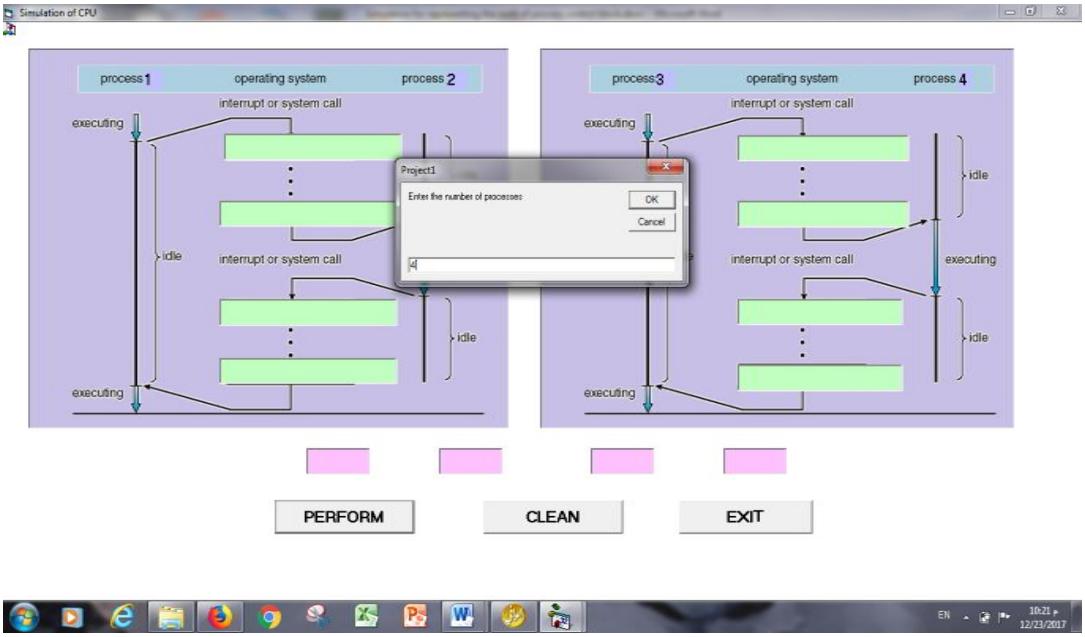


**Figure.7:** Entering the number of processes

Now it will need to inter the number of processes that are ready to run in CPU, which means the processes in short-term scheduler, as shown in Figure.8

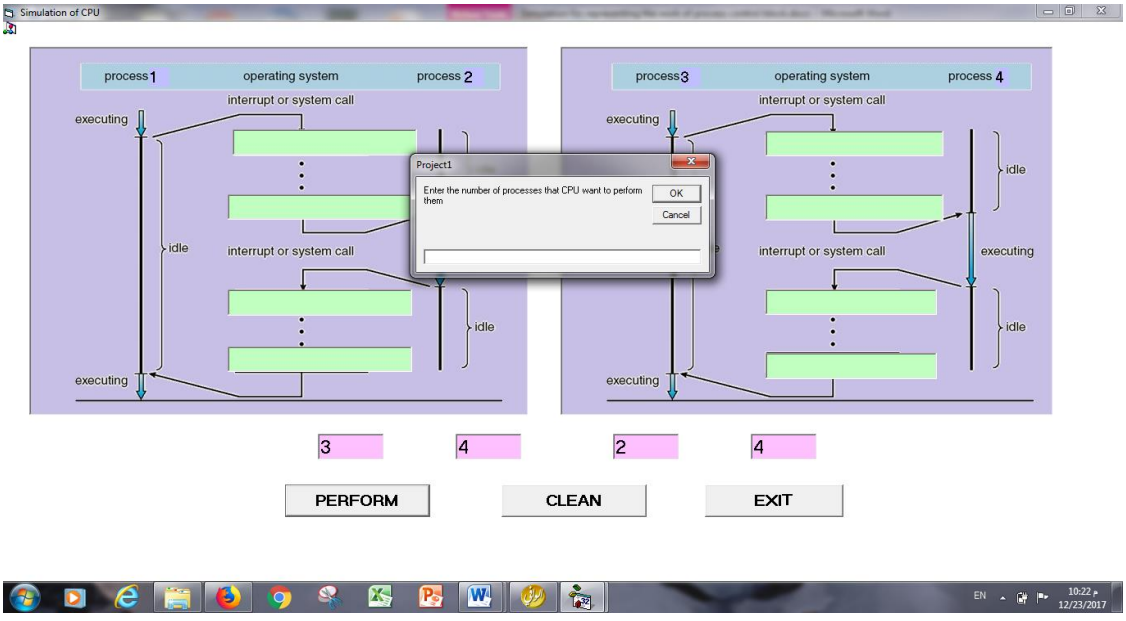


**Figure.8:** Entering processes that are ready to perform

In case there is no process is ready to enter CPU, message will appear to explain this, as shown in Figure.9.

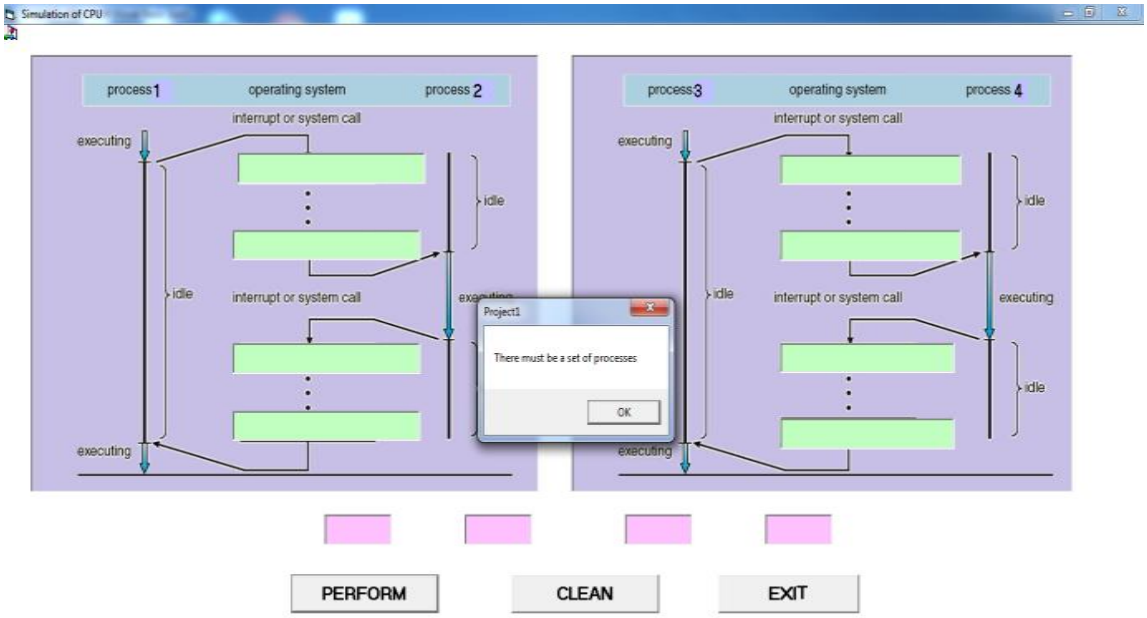


**Figure.9:** Message illustrate no process ready to enter CPU

If user enters number of processes more than processes founded in long-term scheduler, this message will appear, as shown in Figure.10.

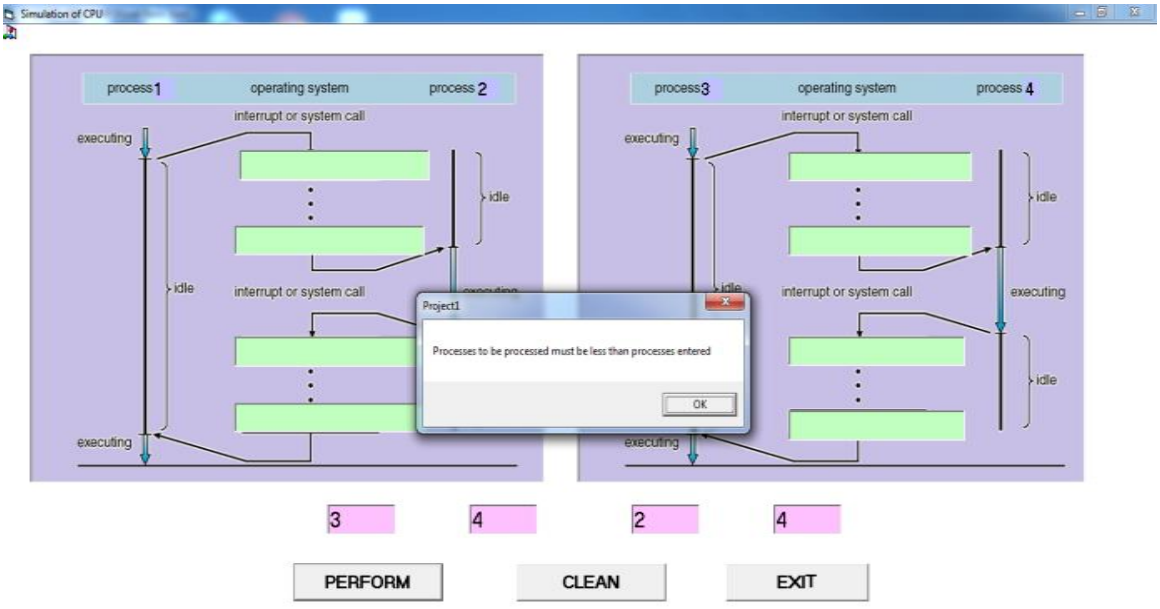


**Figure.10:** Message about number of entering processes

When perform program without cases above, CPU begin to run process using Round Robin as a CPU scheduling, CPU is multithread, this means each process on thread running concurrently, when interrupt happens (after each slot time) the CPU save the state of process (n) in PCB (n) and load the saved state of process (n+1) in PCB (n+1) for each process on each thread in CPU. In the last all processes will finish normally or abnormally (in case founded error), as shown in Figure.11.

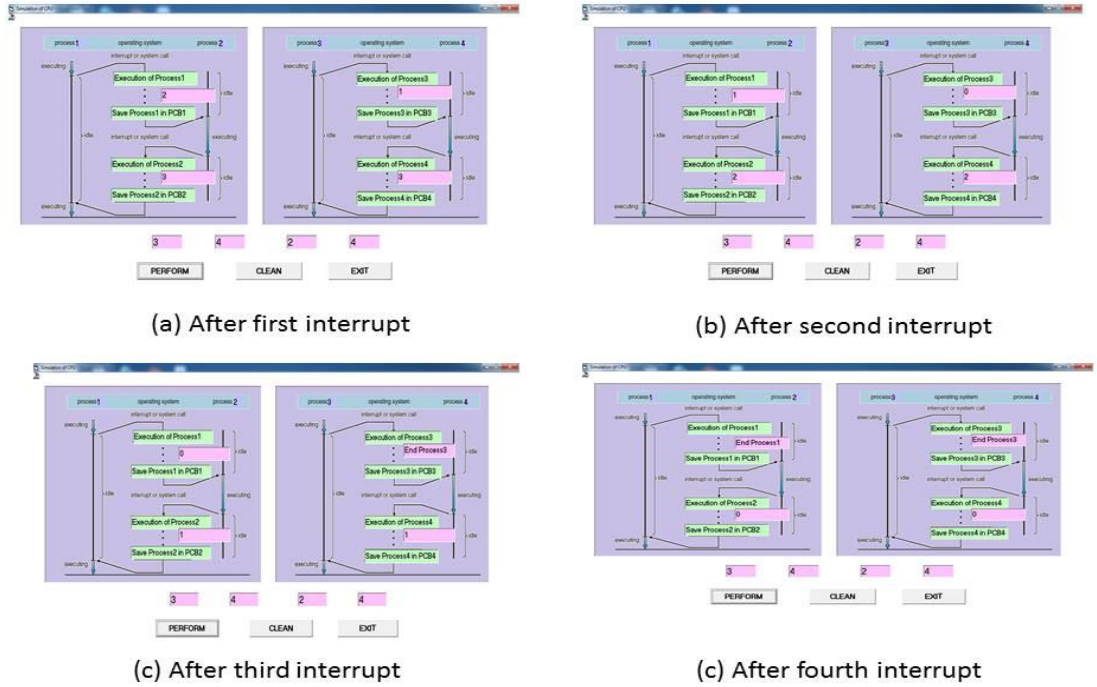


**Figure.11:** Perform program

In case entering all processes founded in long-term scheduler to short-term scheduler, all processes will be ready to run in CPU, as shown in Fig.12 (A).But when entering less from previous case, the PCB of approprate process don't appear any action, as shown in Figure.12 (B).

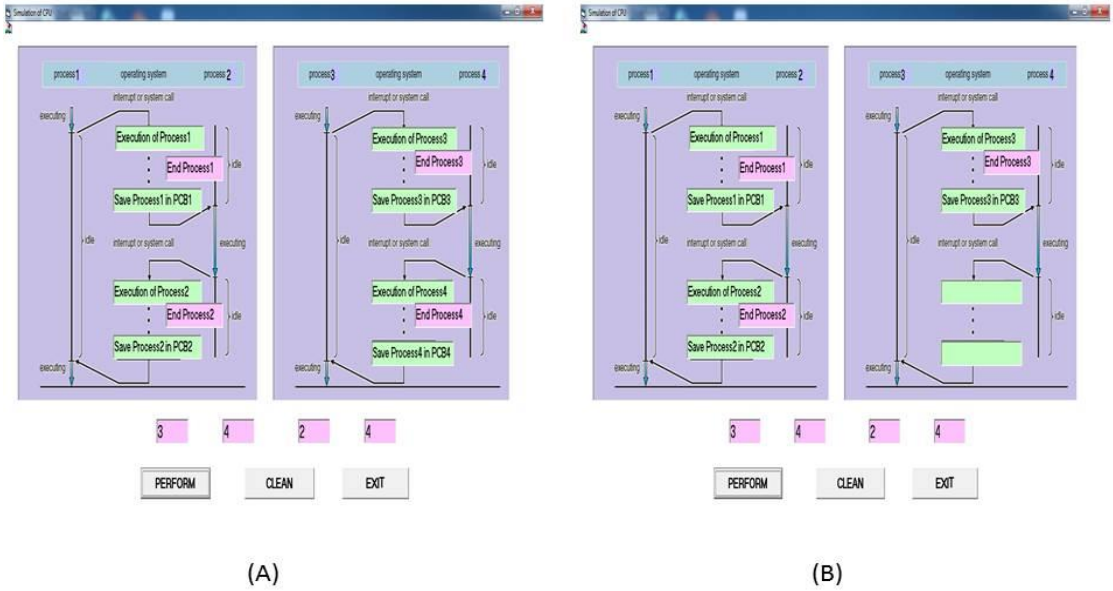


**Figure.12:** Processes that are run in CPU

## 5. Conclusion

This system can be used as educational package to explain how processes run in CPU in simple manner, it also explains role of PCB for each process, and illustrates the different between long-term scheduler and short-term scheduler in understood method.

## 6. Future Work

Improvement the simulation system by represent the states of processes and use the others CPU scheduling to select which process ready to run in CPU . Expansion this system by representing main memory and explaining how to save logical addresses in it.

## REFERENCES

[1] A.Silberschatz, P.B. Galvin, and G.Gagne, Operating Systems Concepts, 8th Edition, John Wiley & Sons, Inc. (2009).
[2] A.S.Tanenbaum, Modern Operating Systems, 3th Edition, Pearson Hall,(2009).
[3] D. W. Gerbing, Computer Components, School of Business Administration Portland State University, (2017).
[4] E. Lazowska, CSE 451: Operating Systems, Allen Center 570,(2007).
[5] G. Cox, The Central Processing Unit: Chapter 4, What Goes on Inside the Computer, Pearson Hall, (2010).
[6] A.Silberschatz, G. Gagne, and P.B.Galvin, Operating System Concepts, Ninth Edition, Chapter 3, John Wiley & Sons, Inc. (2013).
[7] J.Breecher, CS 502: Processes: Chapter 3, Operating Systems, Computer System Department, WPI Institute, (2017).