CSC469
ASSIGNMENT 1

DANIEL BLOEMENDAL
#997678936

Contents

Date: October 9, 2013.

1. Tracking process activity

1.1. **Hypothesis.** The goal of the experiment was to investigate the activity of a single active process running on a modern Linux 3.2 system. In particular we were interested in discovering how long a process is active on the system before it is disrupted by a timer interrupt. Furthermore, we wanted to know how long it takes for the operating system to service the timer interrupt before returning control to the single active process.

Our expectation was that the timer interrupt would be very inexpensive to service. Since no context switch is occurring, the operating system can simply reschedule the already active process and continue its operation. Therefore we expected the process would receive almost all of the CPU time with short consistent interruptions determined by the frequency of the timer interrupt.

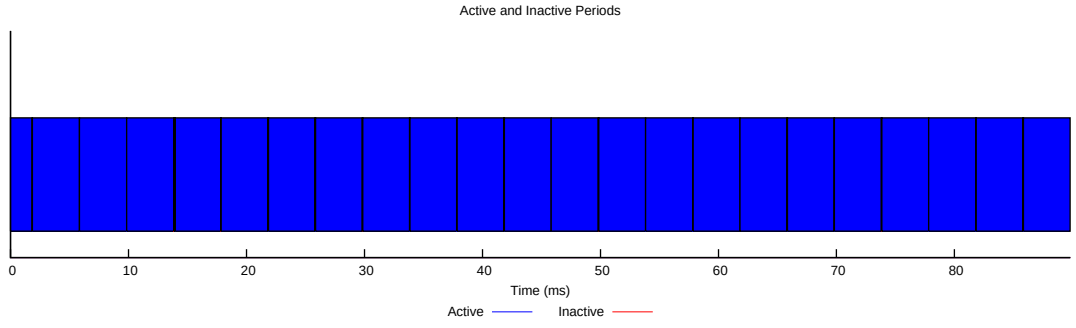
1.2. **Hardware.** The experiment was run on a CDF lab computer with the specifications listed in figure 1.

Figure 1. b2240-06 hardware

Host	b2240-06.cdf.toronto.edu
CPU	Intel® Core™ i5-3570 3.40GHz clock 6MB cache 4 cores
Memory	20GB physical 1GB swap
Kernel	Linux 3.2.0 x86_64

1.3. **Data.** The plot in figure 2 based on 50 gathered samples of active & inactive periods. We considered a process to be inactive if at least 2500 CPU cycles occurred outside of the process. In other words the CPU cycle threshold for the experiment was 2500.

Figure 2. Plot



The following is a table with the first 15 samples of the experiment

FIGURE 3. Samples

Sample	Active (ms)	Inactive (ms)
1	1.841980	0.008492
2	3.990101	0.003561
3	3.994773	0.004417
4	3.994298	0.017977
5	0.105673	0.008597
6	3.854323	0.008083
7	0.003849	0.001198
8	0.013294	0.008201
9	3.963119	0.008064
10	0.004783	0.001016
11	0.019713	0.008198
12	3.956560	0.008127
13	0.004877	0.002799
14	0.011026	0.008207
15	3.963632	0.008121

2. OBSERVATIONS

E-mail address: d.bloemendal@utoronto.ca