

КАФЕДРА №41

инициалы, фамилия

инициалы, фамилия

Санкт-Петербург 2020

1. Постановка задачи

Изучить тайм-ауты ядра на примере программы, приведенной в методическом пособии по лабораторной работе №4 и использующей тайм-ауты ядра и функцию pthreadjoin().

2. Листинг программы

```
#include <stdio.h>
#include <pthread.h>
#include <inttypes.h>
#include <errno.h>
#include <sys/neutrino.h>

#define SEC_NSEC 1000000000LL
void *long_thread(void *notused)
{
    printf("This thread runs longer than 10 seconds \n");
    sleep(20);
}

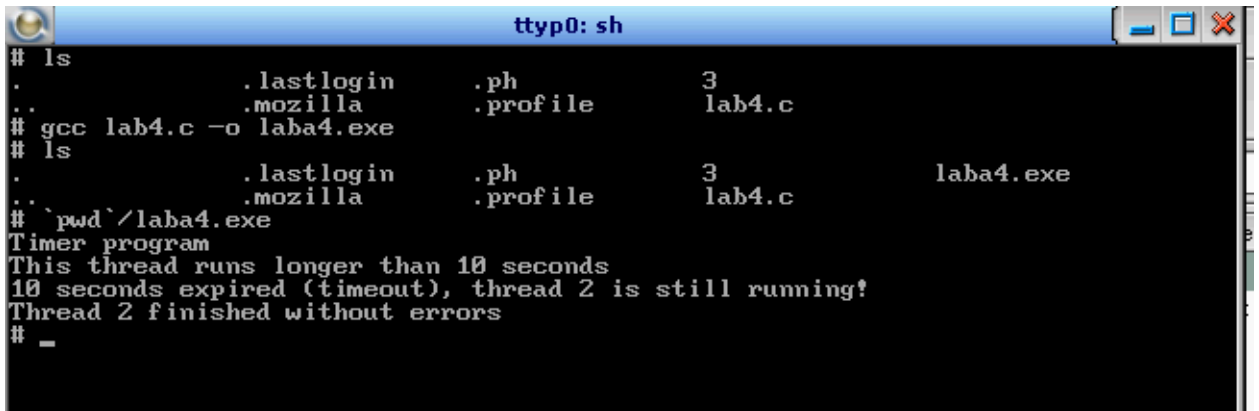
int main(void)
{
    uint64_t timeout;
    struct sigevent event;
    int rval;
    pthread_t thread_id;
    printf("Timer program\n");
    event.sigev_notify = SIGEV_UNBLOCK;
    pthread_create (&thread_id, NULL, long_thread, NULL);
    timeout = 10LL*SEC_NSEC;
    TimerTimeout(CLOCK_REALTIME, _NTO_TIMEOUT_JOIN, &event, &timeout, NULL);
    rval = pthread_join(thread_id, NULL);
    if (rval == ETIMEDOUT)
    {
        printf("10 seconds expired (timeout), thread %d is still running!\n", thread_id);
    }
    sleep(5);
    TimerTimeout (CLOCK_REALTIME, _NTO_TIMEOUT_JOIN, &event, &timeout, NULL);
    rval = pthread_join(thread_id, NULL);
    if (rval == ETIMEDOUT) {
        printf("Thread %d has been running more than 25 seconds (timeout)", thread_id);
    } else {
        printf("Thread %d finished without errors\n", thread_id);
    }
    return 1;
}
```

Рисунок 1 – Текст программы

3. Последовательность действий

Запустить программу на исполнение и сопоставлять то, что она выводит на экран с текстом программы.

4. Результаты работы программы



```
# ls
.      .lastlogin  .ph      3
..     .mozilla   .profile  laba4.c
# gcc lab4.c -o laba4.exe
# ls
.      .lastlogin  .ph      3      laba4.exe
..     .mozilla   .profile  laba4.c
# `pwd`/laba4.exe
Timer program
This thread runs longer than 10 seconds
10 seconds expired (timeout), thread 2 is still running!
Thread 2 finished without errors
# _
```

Рисунок 2 – Результаты работы программы

5. Временная диаграмма исполнения потоков

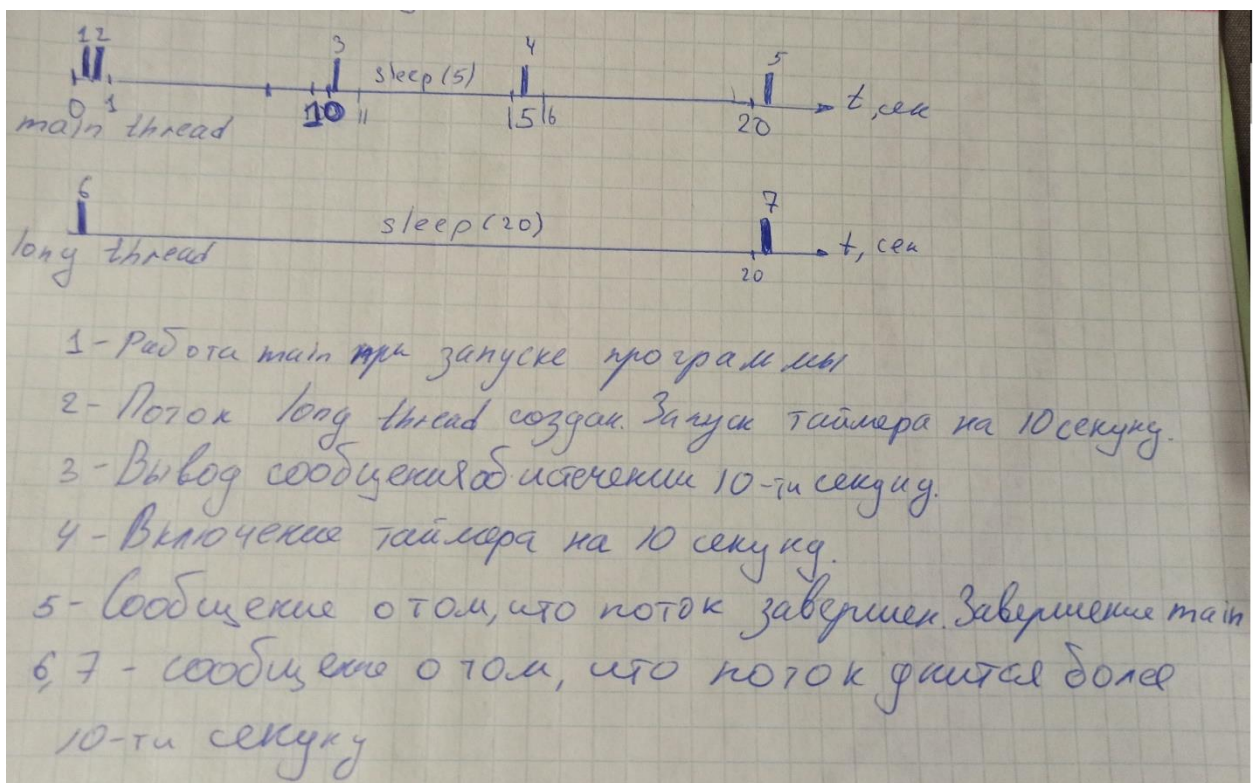


Рисунок 3 – Временная диаграмма исполнения потоков

6. Вывод

В результате выполнения лабораторной работы были изучены тайм-ауты ядра и работа функции pthreadjoin(). Была запущена программа, содержащая в методическом пособии, на исполнение в операционной системе QNX/Neutrino, и был изучен ход ее работы, а также сопоставлен вывод информации на экран с кодом программы. Программа работает корректно, что подтверждается снимком экрана в пункте 4. Помимо этого, была составлена временная диаграмма исполнения потоков.