

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
ГУАП

КАФЕДРА № 14

ОТЧЕТ
ЗАЩИЩЕН С ОЦЕНКОЙ
ПРЕПОДАВАТЕЛЬ

доц., к.т.н,
должность, уч. степень, звание

подпись, дата

О. А. Кононов
инициалы, фамилия

Отчет о лабораторной работе №2

«Процессы и потоки»

по курсу: «Системы реального времени»

РАБОТУ ВЫПОЛНИЛ

СТУДЕНТ ГР. №

1742

подпись, дата

Д. В. Коробков
инициалы, фамилия

Санкт-Петербург 2020

1. Цель

Научится создавать и запускать программу с потоками с помощью программы на Си для ОС QNX.

2. Текст программы

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

pthread_t thread_id1;
pthread_t thread_id2;

void *long_thread1(void *notused)
{
    int n;
    for(n=0;n<5;n++)
    {
        printf("This is first thread TID %d - N of thread %d\n",
thread_id1,n);
        sleep(2);
    }
}

void *long_thread2(void *notused)
{
    int m;
    for(m =0; m <5; m ++ )
    {
        printf("This is second thread TID %d - N of thread %d\n",
thread_id2, m);
        sleep(1);
    }
}

int main(void)
{
    printf("Program threads PID %d\n",getpid());
    pthread_create(&thread_id1, NULL, long_thread1, NULL);
    pthread_create(&thread_id2, NULL, long_thread2, NULL);
    sleep(40);

    return 1;
}
```

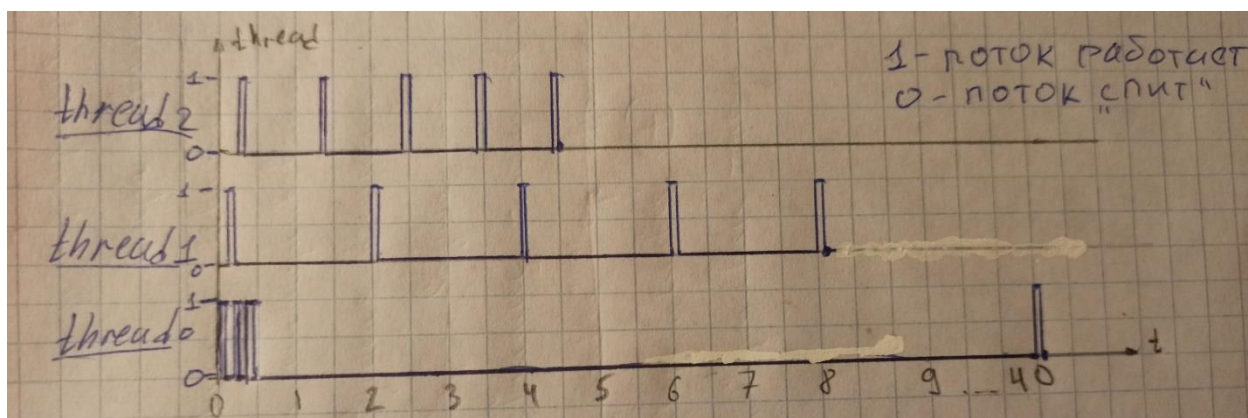
3. Последовательность действий

В программе создаются и запускаются на исполнение два потока. Когда один поток приостанавливается, сразу начинает работу другой. Приостановка реализована функцией `sleep(n)`, которая останавливает процесс на `n` секунд. На экране можно наблюдать, как по очереди работают два процесса

4. Результат

```
ttyp0: a.out
# ls
.          a.out          lab4.c
..         fuck          lab5.c
.lastlogin fuck.c         labushka lyub imaya.s
.mozilla  lab2.c          ls
.ph       lab3client.c  ls.c
.profile  lab3serv.c
4331ilin  lab4
# gcc lab2.x
gcc: lab2.x: No such file or directory
gcc: no input files
# gcc lab2.c
# ./a.out
Program threads PID 581651
This is first thread tid 0 - N of thread 0
This is second thread tid 0 - N od thread 0
This is second thread tid 3 - N od thread 1
This is first thread tid 2 - N of thread 1
This is second thread tid 3 - N od thread 2
This is second thread tid 3 - N od thread 3
This is first thread tid 2 - N of thread 2
This is second thread tid 3 - N od thread 4
This is first thread tid 2 - N of thread 3
This is first thread tid 2 - N of thread 4
```

5. Временная диаграмма исполнения потоков



6. Вывод

Освоил создание потоков в QNX используя программу, написанную на языке Си. Понял, что если объявлять переменную потока `tid` как глобальной, то поток может начать работу до присвоения ей значения. Поэтому в результатах видно, что сначала `tid = 0` на скрине работы (не успел присвоиться `tid`), а далее потоки выполняются верно.