

Лабораторная работа 1.1

Xenomai Real Time Linux — создание простой задачи.

Цель работы: Знакомство с Xenomai real time Linux, создание простой задачи реального времени.

Аппаратное и программное обеспечение: PC, ОС Linux, Virtual Box, Xenomai virtual box image.

Порядок выполнения работы:

1. Работа выполняется на локальной или удаленной виртуальной машине с установленным Xenomai. В случае выполнения работы на локальной виртуальной машине запустите VirtualBox и активируйте виртуальную машину с установленным Xenomai. Для входа в систему используйте имя пользователя **root** и пустой пароль. В случае использования удаленной виртуальной машины запустите терминал и введите команду:

```
ssh -l root ip_адрес_удаленной_машины
```

2. Необходимо создать рабочий каталог, в качестве имени каталога необходимо указать фамилии выполняющих работу, например:

```
mkdir Ivanov_Petrova
```

3. В рабочем каталоге необходимо создать файл lab11.c содержащий код программы запуска простой задачи в режиме реального времени. Пример кода:

```
#include <stdio.h>
#include <signal.h>
#include <unistd.h>
#include <sys/mman.h>
#include <native/task.h>
#include <native/timer.h>
#include <rtdk.h>

RT_TASK demo_task;

void demo(void *arg)
{
    RT_TASK *curtask;
    RT_TASK_INFO curtaskinfo;

    // hello world
    rt_printf("Hello World!\n");

    // запрос текущей задачи
    curtask=rt_task_self();
    rt_task_inquire(curtask,&curtaskinfo);

    // печатаем имя задачи
    rt_printf("Task name : %s \n", curtaskinfo.name);
}

int main(int argc, char* argv[])
{
    char str[10];

    // выполняем авто-инициализацию rt_print buffers если задача этого не сделала...
```

```

rt_print_auto_init(1);

// Запрет использования swap для программы...

mlockall(MCL_CURRENT|MCL_FUTURE);

rt_printf("start task\n");

/*
 * Arguments: &task,
 *           name,
 *           stack size (0=default),
 *           priority,
 *           mode (FPU, start suspended, ...)
 */

sprintf(str,"hello");

rt_task_create(&demo_task, str, 0, 50, 0);

/*
 * Arguments: &task,
 *           task function,
 *           function argument
 */

rt_task_start(&demo_task, &demo, 0);
}

```

4. Для компилирования программы необходимо выполнить предварительные команды для определения и установки flags:

```

export CFLAGS=`xeno-config --xeno-cflags`
export LDFLAGS=`xeno-config --xeno-ldflags`

```

далее выполняются команда компилирования:

```
gcc $CFLAGS $LDFLAGS -lnative -lrt -ldk lab11.c -o lab11
```

для запуска программы может потребоваться регистрация дополнительных библиотек командой:

```
export LD_LIBRARY_PATH=/usr/xenomai/lib
```

команда запуска программы:

```
./lab11
```

5. Создать вторую задачу с более низким приоритетом. Вторая задача должна выводить в консоль фамилии выполняющих работу.

Источники информации:

<http://www.cs.ru.nl/lab/xenomai/>
www.xenomai.org
www.google.ru