НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ

«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ»

ФАКУЛЬТЕТ ІНФОРМАТИКИ І ОБЧИСЛЮВАЛЬНОЇ ТЕХНІКИ

КАФЕДРА ОБЧИСЛЮВАЛЬНОЇ ТЕХНІКИ

**Лабораторна робота №5**

*з дисципліни* ***«****Архітектура комп’ютерів 2****»***

Виконав:

студент 3 курсу

групи ІО-81

Мельнійчук Б.Ю.

Перевірив:

Каплунов А. В.

Київ 2020 р.

Посилання на вхідні збудовані файли

https://github.com/Bohdan00735/Ak\_2\_BBB

Завдання Basic

Лістинг hello.c був змінений на наступний

/\*

\* Copyright (c) 2017, GlobalLogic Ukraine LLC

\* All rights reserved.

\*

\* Redistribution and use in source and binary forms, with or without

\* modification, are permitted provided that the following conditions are met:

\* 1. Redistributions of source code must retain the above copyright

\* notice, this list of conditions and the following disclaimer.

\* 2. Redistributions in binary form must reproduce the above copyright

\* notice, this list of conditions and the following disclaimer in the

\* documentation and/or other materials provided with the distribution.

\* 3. All advertising materials mentioning features or use of this software

\* must display the following acknowledgement:

\* This product includes software developed by the GlobalLogic.

\* 4. Neither the name of the GlobalLogic nor the

\* names of its contributors may be used to endorse or promote products

\* derived from this software without specific prior written permission.

\*

\* THIS SOFTWARE IS PROVIDED BY GLOBALLOGIC UKRAINE LLC ``AS IS`` AND ANY

\* EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED

\* WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE

\* DISCLAIMED. IN NO EVENT SHALL GLOBALLOGIC UKRAINE LLC BE LIABLE FOR ANY

\* DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES

\* (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;

\* LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND

\* ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT

\* (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS

\* SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

\*/

#include <linux/init.h>

#include <linux/module.h>

#include <linux/moduleparam.h>

#include <linux/printk.h>

#include <linux/types.h>

MODULE\_AUTHOR("Melniichuk Bohdan IO-81");

MODULE\_DESCRIPTION("First training kernel module wich repeat phrase");

MODULE\_LICENSE("Dual BSD/GPL");

static uint amount = 1;

module\_param(amount,uint,0);

MODULE\_PARM\_DESC(amount, "That`s how many times the line should be printed");

static int \_\_init init\_fuction(void)

{

uint i = 0;

printk(KERN\_INFO "times: %d\n", amount);

if(amount == 0) {

printk(KERN\_WARNING "The parameter is 0");

}else if(amount >=5 && amount <= 10) {

printk(KERN\_WARNING "The parameter is between 5 and 10");

} else if(amount > 10) {

printk(KERN\_ERR "The parameter is greater than 10");

return -EINVAL;

}

for(i = 0; i < amount; i++){

printk(KERN\_INFO "Hello, world!\n");

}

return 0;

}

static void \_\_exit cleanup\_function(void)

{

/\* Do nothing here right now \*/

}

module\_init(init\_fuction);

module\_exit(cleanup\_function);

Після чого змонтований та завантажений файл .ko в теку з емулятором. Тут використовувався як на мене найпростіший спосіб, з використанням наступного makefile

ifneq ($(KERNELRELEASE),)

# kbuild part of makefile

obj-m := hello.o

else

# normal makefile

KDIR ?= /lib/modules/`uname -r`/build

default:

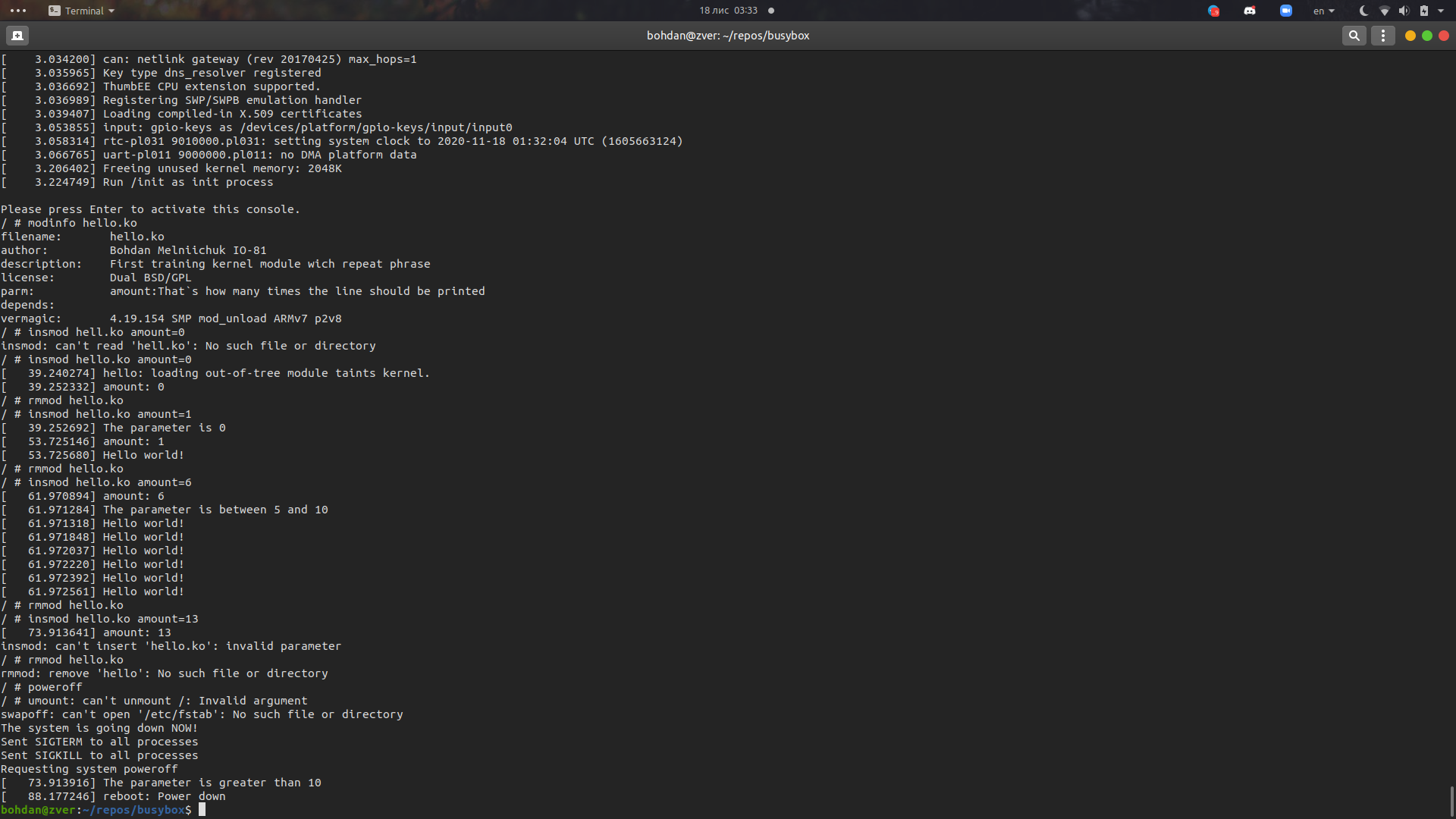
$(MAKE) -C $(KDIR) M=$$PWD

clean:

$(MAKE) -C $(KDIR) M=$$PWD clean

endif

Робота на запущеному емуляторі



Завдання Advanced

Тут були створені наступні:

hello2.c

#include "hello1.h"

static uint amount = 1;

module\_param(amount,uint,0);

MODULE\_PARM\_DESC(amount, "That`s how many times the line should be printed");

static int \_\_init init\_fuction(void)

{

uint i = 0;

printk(KERN\_INFO "times: %d\n", amount);

if(amount == 0) {

printk(KERN\_WARNING "The parameter is 0");

}else if(amount >=5 && amount <= 10) {

printk(KERN\_WARNING "The parameter is between 5 and 10");

} else if(amount > 10) {

printk(KERN\_ERR "The parameter is greater than 10");

return -EINVAL;

}

for(i = 0; i < amount; i++){

print\_hello();

}

return 0;

}

static void \_\_exit cleanup\_function(void)

{

/\* Do nothing here right now \*/

}

module\_init(init\_fuction);

module\_exit(cleanup\_function);

**hello1.c файл, що експортує функцію**

#include "hello1.h"

void print\_hello(void){

printk(KERN\_INFO "Hello, world!\n");

}

***inc/hello1.h заголовний файл***

#include <linux/init.h>

#include <linux/module.h>

#include <linux/moduleparam.h>

#include <linux/printk.h>

#include <linux/types.h>

MODULE\_AUTHOR("Melniichuk Bohdan IO-81");

MODULE\_DESCRIPTION("First training kernel module wich repeat phrase");

MODULE\_LICENSE("Dual BSD/GPL");

extern void print\_hello(void);

**Також було змінено файл Kbuild**

# kbuild part of makefile

obj-m := hello1.o hello2.o

ccflags-y += -I$(obj)/inc

**Makefile**

ifneq ($(KERNELRELEASE),)

include Kbuild

else

# normal makefile

KDIR ?= /lib/modules/`uname -r`/build

default:

$(MAKE) -C $(KDIR) M=$$PWD

clean:

$(MAKE) -C $(KDIR) M=$$PWD clean

endif

Ці файли були також зібрані, додані та запущені. Результат виконання

