Министерство образования Республики Беларусь

Учреждение образования

БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Факультет компьютерных систем и сетей

Кафедра информатики

дисциплина архитектура вычислительных систем

Отчет к лабораторной работе №1

Защищенный режим 32-разрядных процессоров

Выполнил: студент группы 053505

Слуцкий Никита Сергеевич

Проверил: ассистент кафедры информатики

Калиновская Анастасия Александровна

Минск 2022

**1. Цель работы**

Изучить особенности защищенного режима процессора. Получить практические навыки по программированию переключения процессора из реального в защищенный режим и обратно.

**2. Постановка задачи**

Написать программу, переключающую процессор в защищенный режим, выводящую на экране сообщение и затем возвращающую процессор в реальный режим.

**3. Программная реализация**

3.1 При запуске программа не выдает сообщение об ошибке, а следовательно работает верно.

3.2 Когда работаем в защищенном режиме, нельзя сделать вызов конкретно этого прерывания. Если все-таки сделать вызов прерывания, произойдет ошибка, то есть запрет на данную команду. Программа вылетает*.* А вместе с ней и весь DOSBox. Но важно понимать, что это был выбран такой способ демонстрации того, что защищённый режим включился. В общем случае не должно быть такого, что прерывания, если их заново включить, будут “ломать” программу.

3.3 Можно проверить будет ли ошибка, если вызвать прерывание в реальном режиме. Программа отрабатывает без аварийного завершения.

**5. Выводы**

В ходе выполнения данной лабораторной работы были разобраны базовые различия работы процессора в реальном и защищённом режимах. Был написан код для перехода из реального режима в защищенный и обратно, для проверки нахождения в реальном или защищенном режиме вызывалось прерывание, которое можно запустить только в реальном режиме. Цели лабораторной работы можно считать достигнутыми.

**Приложение 1.**

**use16**

**org 100h**

**start:**

**; At the beginning we are in real mode**

**mov ah, 00**

**int 21h**

**; Prohibit masked interruptions**

**cli**

**; Prohibit not basked interruptions (NMI)**

**in   al, 70h**

**or   al, 80h**

**out 70h, al**

**; Open Line A20**

**in   al, 92h**

**or   al, 2**

**out 92h, al**

**; Switch to protected mode**

**; Let's try to do smth, we can't allow in protected mode**

**mov eax, cr0**

**or    al, 1**

**mov cr0, eax**

**; In protected mode**

**; Let's try to do smth, we can't allow in protected mode**

**;mov ah, 00**

**;int 21h**

**; Switch to real mode**

**mov eax, cr0**

**and  al, 0FEh**

**mov cr0, eax**

**; Close A20 Line**

**in    al, 92h**

**and al, 0FDh**

**out  92h, al**

**; Allow not masked**

**in    al, 70h**

**and al, 7Fh**

**out  70h, al**

**; Allow masked**

**sti**

**; real working again**

**ret**