Федеральное государственное образовательное бюджетное

учреждение высшего образования

**«Финансовый университет при Правительстве Российской Федерации»**

**(Финансовый университет)**

Колледж информатики и программирования

**ОТЧЁТ**

**По лабораторной работе №2**

Студент: Чернов Михаил Сергеевич   
Дисциплина/Профессиональный модуль: Архитектура Аппаратных средств  
Группы: 2ИСИП-519

Преподаватель

Сибирев И.В.

Оценка за работу :\_\_\_\_\_\_\_

**Москва – 2020г.**

**Лабораторная работа №2**

**Задание.**

**Познакомиться с ассемблером. Вывести в порты рисунок. Сделать для рисунка анимацию.**

**L2.1 Файл эксель**

**В нем нарисовать закрашенными квадратиками Ваш авторский рисунок (8 на 16 клеток).**

**Например, котика... Домик уже занят.**

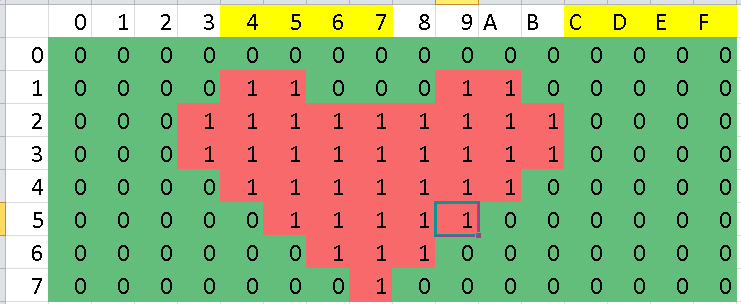
**L2.2 Привести бинарные коды домика в шеснадцатеричные коды.**

**L2.3 Нарисовать ваш домик в портах**

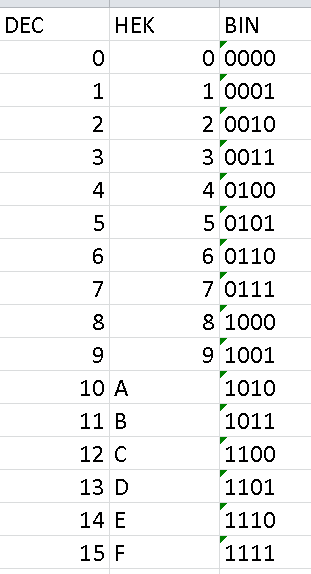
**L2.4 При помощи бесконечного цикла заставить летать домик с низу в верх, с права на лево, и моргать вместе с фоном**

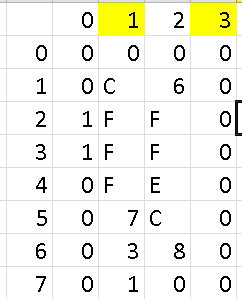
**L2.5 Выучить все команды которые использовали (до уровня более или менее, суметь воспроизводить).**

Задание на лабораторную работу.

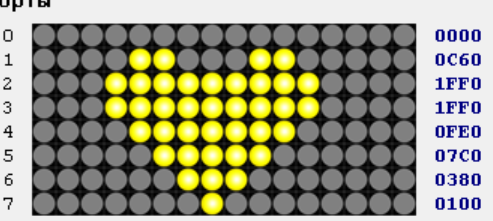
Ход выполнения лабораторной работы.  
Мы нарисовали сердце при помощи нулей и единиц 8 на 16 клеток.  


Мы воспользовались таблицей перевода из двоичных в шеснадцатеричные коды.

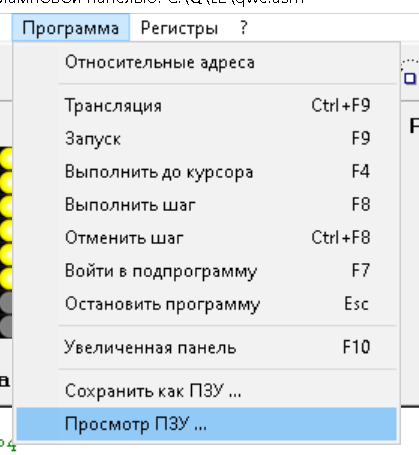
  
Так мы получили шеснадцатеричные коды рисунка сердца.



Мы написали ассемблерные код, выводящий в порты рисунок…



|  |
| --- |
| mov 0000,r0  out r0,p0  mov 0c60,r1  out r1,p1  mov 1ff0,r2  out r2,p2  mov 1ff0,r3  out r3,p3  mov 0Fe0,r0  out r0,p4  mov 07c0,r1  out r1,p5  mov 0380,r2  out r2,p6  mov 0100,r3  out r3,p7  stop |

Теперь делаем анимацию при помощи бесконечного цикла и системных процедур…  
 

|  |
| --- |
| Анимация: циклическое движение с лева на право, с низу в верх, и моргание… m:  ;дВИГАЕМ КРЫШУ  IN P0,R0  ROR 1,r0  not r0  OUT R0,P0  ;  IN P1,R0  ROR 1,r0  not r0  OUT R0,P1  ;  IN P2,R0  ROR 1,r0  not r0  OUT R0,P2  ;  IN P3,R0  ROR 1,r0  not r0  OUT R0,P3  ;  IN P4,R0  ROR 1,r0  not r0  OUT R0,P4  ;  IN P5,R0  ROR 1,r0  not r0  OUT R0,P5  ;  IN P6,R0  ROR 1,r0  not r0  OUT R0,P6  ;  IN P7,R0  ROR 1,r0  not r0  OUT R0,P7  ;  system 3  JMP m  Stop |

