本项目是从零开发而来的，所有的细节都在底层实现！

ActSpace系统结构

主菜单

游戏场景

Start

Return

素材

模块说明

main.asm模块主要包括一个游戏的整体逻辑结构，调用了很多其他模块实现好的子程序。

gameSys.asm是游戏核心管理系统模块，它实现了重要游戏资源的管理功能，爆炸效果的实现也在其中。

menu.asm模块实现的是主菜单用户界面。

scene.asm模块实现了场景的布局和图片处理功能。

control.asm模块实现了键盘操作的功能。

BMPF.asm模块实现的是图片文件的加载功能。

fighter.asm模块实现的是玩家战机及其子弹对象功能。

foes.asm模块实现的是敌机和敌机子弹对象功能。

bump.asm模块实现的是游戏对象碰撞检测功能。

gamePara定义一系列共用参数供各模块使用。

简要流程

预备资源

调主菜单

选Start

游戏初始化

加载场景

游戏循环

esc键？

结束？

N

刷新纪录

Y

关键技术

1.游戏动画的实现

项目基于DOS系统开发，适合处理较简单、规格小的图片。

静态背景、目标移动的方式。

采用异或技术实现移动。（任意数a与任意数b异或两次恢复原数）

XOR

XOR

Ⅰ

Ⅱ

Ⅲ

对象的显示与背景恢复

2.游戏对象的生成销毁

由于同一片刻，对象的数量是有限的，为了实现的简便，遍历的高效，采用数组批量管控对象。

3.游戏对象的碰撞判定

采用对象位置中心坐标与矩形检测。