Шейдер кубика

* \_MainClrCube("Main color", Color) = (1,1,1,1) Основной цвет куба, зависящий от типа фигуры
* \_InCubeFaceClr("In face", Color) = (1,1,1,1) Цвет внутренних граней куба
* \_OutCubeFaceClr("Out faces", Color) = (1,1,1,1) Цвет внешних граней куба
* \_MainClrCubePw("Pw main", float) = 1 коэффициент умножения для основного цвета куба
* \_InCubeFaceClrPw ("Pw in faces", Float) = 1 коэффициент умножения для цвета внутренних граней
* \_OutCubeFaceClrPw ("Pw out faces", Float) = 1 коэффициент умножения для цвета внешний граней
* \_CubePower1 ("Cmn pow 1", Float) = 1 свободный коэффициент 1 только для шейдера куба
* \_CubePower2 ("Cmn pow 2", Float) = 1 свободный коэффициент 2 только для шейдера куба
* \_CubeClrMap("Cube Color Map", 2D) = "white" текстура цветовой карты для шейдера кубика
* \_CubeAlphaMap(“Cube Alpha Map”, 2D) = “white” карта альфа канала для шейдера кубика

Шейдер рамки

* \_BorderCurrentClr("Current Color", Color) = (1,1,1,1) актуальный цвет рамки, соответствующий цвету падающего кубика
* \_BorderNextClr("Next Color", Color) = (1,1,1,1) следующий цвет рамки, соответствующий следующей фигуре
* \_BorderSecClr(“Sec color”, color) = (1,1,1,1) вторичный цвет рамки, не зависящий от падающей фигуры
* \_BorderClrCounter("Color counter", float) = 1 счетчик, определяющий степень интреполяции (смешивания) между актуальным и следующим цветом
* \_BorderCommonPw("Cmn Pw", float) = 1 общий множитель яркости цветов для рамки
* \_BorderCurrClrPw(“Current Color Power”, float) = 1 множитель яркости актуального цвета
* \_BorderNextClrPw(“Next Color Power”, float) = 1 множитель яркости следующего цвета
* \_BorderCurrClrStep("Curr Color Interpol Step", float) = 1 степень интерполяции актуального цвета
* \_BorderNextClrStep("Next Color Interpol Step", float) = 1 степень интерполяции следующего цвета
* \_BorderPower1 ("Cmn pow 1", Float) = 1 свободный коэффициент 1 только для шейдера рамки
* \_BorderPower2 ("Cmn pow 2", Float) = 1 свободный коэффициент 2 только для шейдера рамки
* \_BorderClrMap("Brd Color Map", 2D) = "white" текстура цветовой карты для шейдера рамки
* \_BorderAlphaMap(“Brd Alpha Map”, 2D) = “white” карта альфа канала для шейдера рамки