Read Me

loop(): inside a loop, there will be different game states: splash screen, in game, explosion animation and score display. Based on the value of gameState, the program will show different game states in actual games.

start(): when the player presses shooting button, it will change the game state to in game and game will start. It will initialize all the in game parameters.

explodeAnimation(): plays the animation of the explosion.

inverseSplashScreen(): flips every bits in splash screen.

movePlayer(): When the player use the potentiometer, the spaceship will move horizontally.

checkPlayerStatus(): checks if the spaceship got hit. If the spaceship got hit, it changes the game state to play explosion animation.

shoot(): when the player presses shoot button, the spaceship shoots a bullet.

createBullet(): creates a bullet when the player shoot.

bulletMove(): move bullets in a certain amount of time.

destroyAsteriods(int row): if the bullet hits asteroids, destroy the asteroid.

hitAsteriods(): check if bullets hit asteroids.

clearPassingAsteriod(int row): because concurrency issues some asteroids will pass the bullet and wont be destroyed, the function will clear the passing asteroids.

asteriodGeneration(): procedurally generate different types of asteroids in different positions.

asteriodMoveDown(): move asteroids down in a certain amount of time.

displayScore(): when the game state changes to display score, shows the score player got in the game.

render(byte contents[]): based on 2d bytes array, controls each LED in the matrix on and off.