

DESIGN DOCUMENT:

MAYD

Aint

Yet

Developed

Created By:

Nabeel Ud Daulah

Surag Munekar

Vladislav Adzic

George Yeboah

Overview:

Our project is a three-dimensional video game in the style of a first person shooter. We will use a serial connection through the UART port to receive the input that controls the game's main character. We will send keyboard input from a separate computer through the UART port. The game and the 3D engine will be written in C. It will use a technique known as raycasting to simulate the three-dimensional environment. We will write the display of our project in VHDL. Graphics we use will be stored in SRAM

Game Description:

The game will be based in a three dimensional environment with rooms and passages. The user will be the main character in the game and will be able to move around freely in this layout. The user will control character movement using a keyboard. If we are successful in implementing this we will implement additional features such as enemies that the user will be able to shoot down and goodies that can be collected. These additional interactive elements will be implemented using sprites.

Software:

Our software will be coded using C. We will use the raycasting method to create our three dimensional environment. We start out with a two-dimensional grid layout, where each grid is a zero or a positive number. Zero grids indicate open space, where as the numbered grids indicate walls or objects (the specific number indicating the color or texture). We will then use the raycasting method to represent the three-dimensional perspective of our two dimensional layout. This is done as follows: For every vertical stripe on the screen we will send out a ray that initiates at the player position with a direction dependent on the player's looking direction and the x-coordinate on the screen. These rays will move forward on the two dimensional layout until it hits a wall. Once hit, the distance of the ray from the position of the player is calculated in order to determine the height of the wall to be drawn on the screen.

We will record the time taken to load each frame in order to determine how much the main character should move when a certain key is pressed. This is done so that player movement occurs at a constant speed, even if the processor supports can operate at a faster speed.

User input through the keyboard will be limited to 7 keyboard keys for the following actions: move forward, move backward, move (strafe) right, move (strafe) left, rotate right, rotate left, fire.

Hardware:

We will use a VGA display to display our graphics on a monitor. Unlike the font-based terminal display used in Lab 2, we will require a pixel based VGA device. For each frame displayed, we will calculate the color value of each pixel. This information will be written to the VGA controller, which in turn, sends the image to the monitor.

Block Diagram:

