Nodus of Evil

Instruction Manual

Thank you for Trying out Nodus of Evil

Game Description

In a land controlled by fearsome gods, the battlefield is set. The cruel god Satrina awaits her opponent. The victor will shape the world in their image.

This world is mine

-Satrina

How To Play

Press left or right
To move
accordingly

Press A To
jump

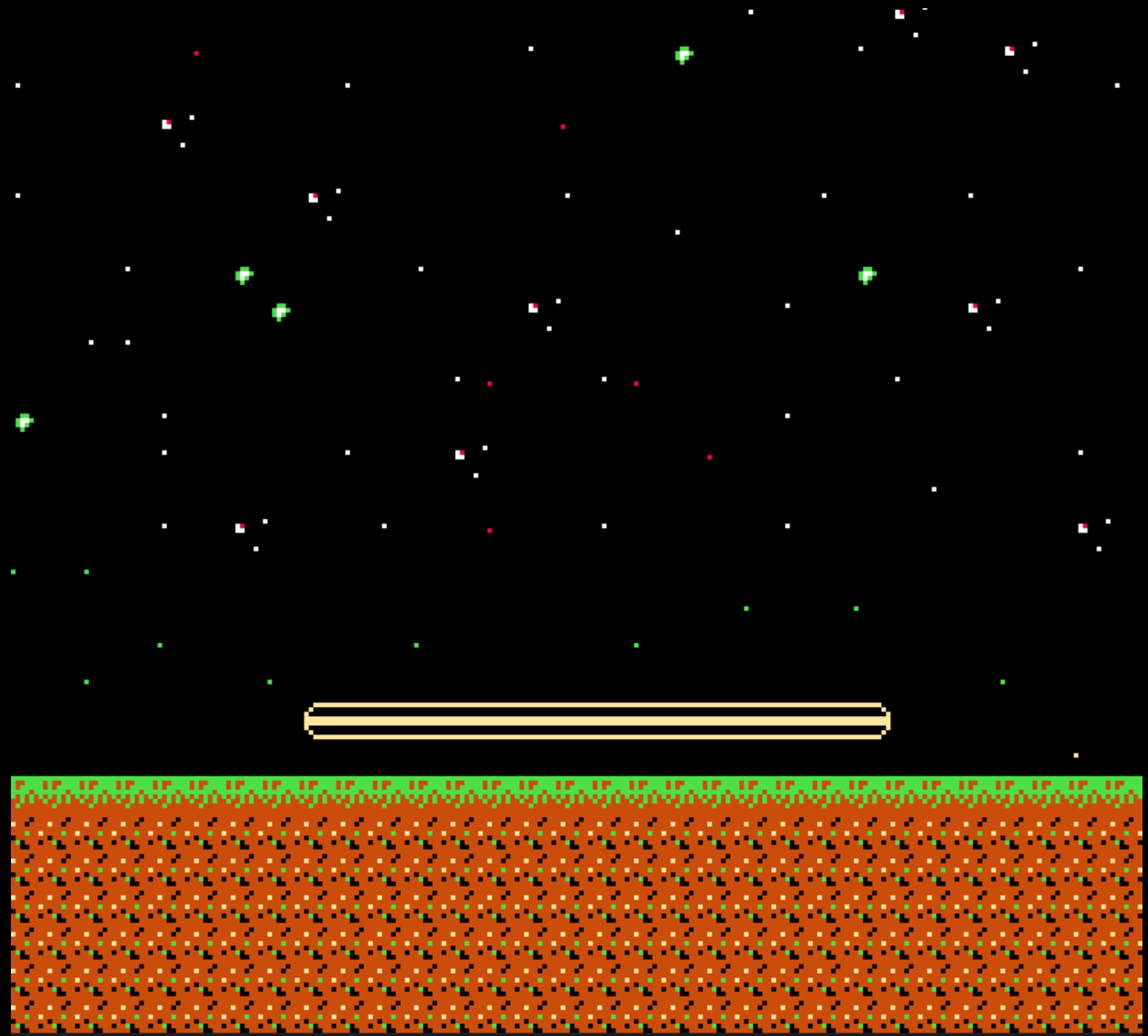
How it was Made Background

- NEXXT was used to create a nameTable.
- Using the NEXXT canvas, the background was designed
- In the actual code, information provided by nexxt was used to load the appropriate values into the ppu.
- These values allowed for the display of the background.

```
500 ; ----------------------------
501     LDY #$c0
502     LDX #$40
503     load_floor:
504     LDA PPUSTATUS
505     LDA #$22
506     STA PPUADDR
507     TYA
508     INY
509     STA PPUADDR
510     STX PPUDATA
511     cpy #$df
512     bne load_floor
513
514
515
516
```

A loop designed to load the floor tiles

The battlefield is set...



How it was made Character

- Pixilart.com was used to design the main character sprite
- The character sprite is made up of 6 different Tiles.
- Recreating the sprite in NEXXT allowed the graphics to be included in the game's chr file.

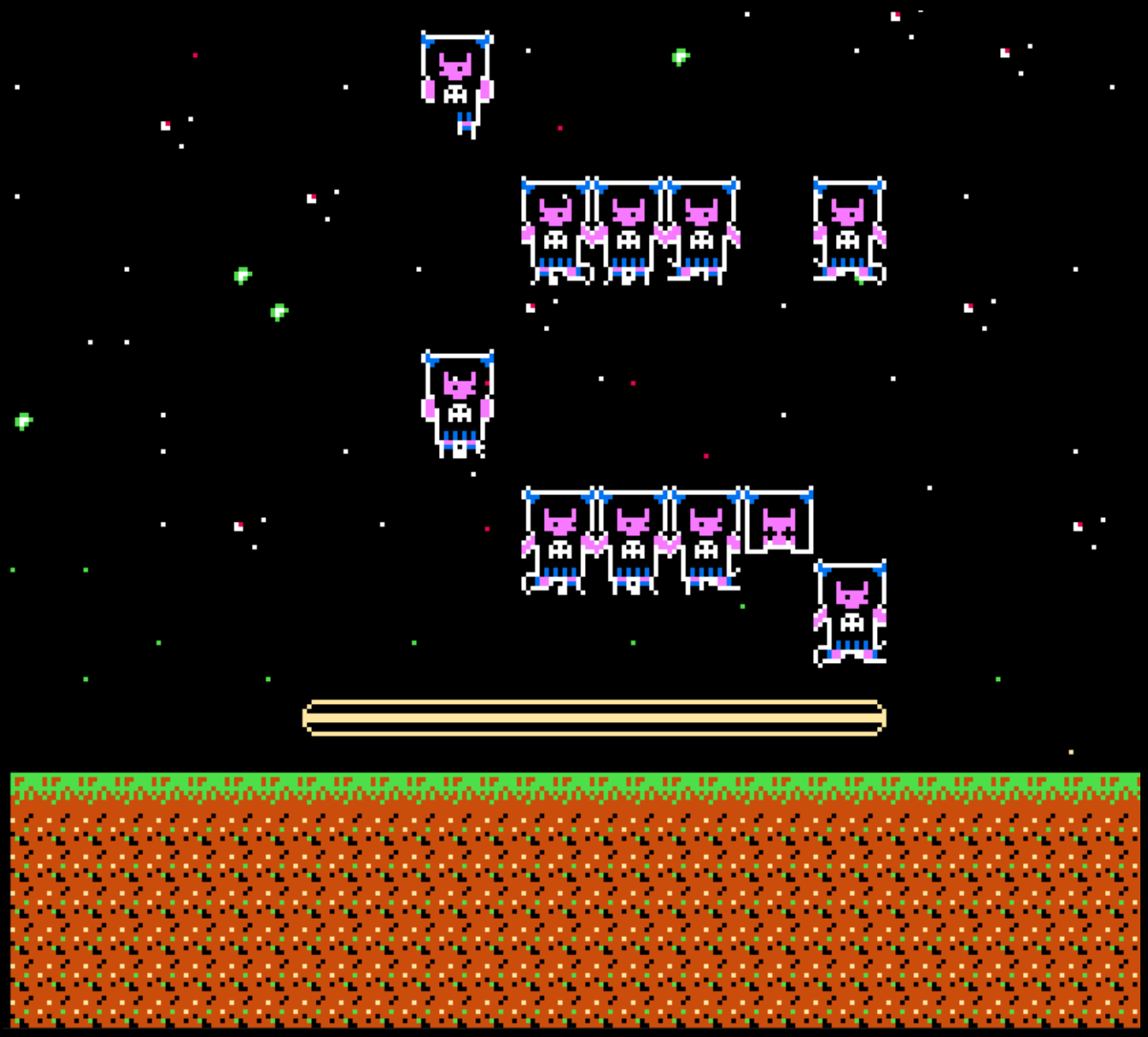
```
LDA satrina_y ; Y default: #$97
STA $0200
LDA #$02
STA $0201
LDA #$01
STA $0202 ; ATTRIBUTE TABLE, to FLIP, do $41
LDA satrina_x
STA $0203 ; X Location LEFT HEAD

LDA satrina_y; Y default #$97
STA $0204
LDA #$03
STA $0205
LDA #$01
STA $0206 ; ATTRIBUTE TABLE, to FLIP, do $41
LDA satrina_x ; X
CLC
ADC #$08
STA $0207 ; X Location RIGHT HEAD

LDA satrina_y ; Y default #$9f
STA $0208
LDA #$12
STA $0209
LDA #$01
STA $020a ; ATTRIBUTE TABLE, to FLIP, do $41
LDA satrina_x ; X
STA $020b ; X Location LEFT BODY
```

Code for loading the first 3 sprites corresponding to satrina

Satrina arrives



How it was made Animation

- During the non maskable interrupt. The render and draw procedures are called.
- animDelay and animCount were created in the zeropage.
- Everytime animdelay reaches 10, animcount increases by 1.
- Draw renders the sprite according to animcount.

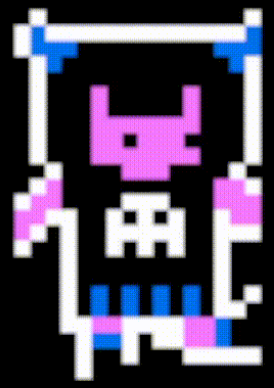
```
pass:


    LDX animDelay
    cpx #$0f  ; Delay between frames
    beq executeAnim
    INX
    STX animDelay
    jmp endRender


executeAnim:
    LDA satrina_dir
    CMP #$00
    beq flipped
    LDA #$01
    STA $0202
    STA $0206
    STA $020a
    STA $020e
    LDX #$00
    STX animDelay
    LDX animCount
    cpx #$00
    beq RunFirstStage
    cpx #$01
    beq RunIdleStage
    cpx #$02
    beq RunFinalStage
```

Code handling the animation

Satrina is warming up...



How it was made Movement

- Player x and y values were declared in the zeropage.
- These values were incremented or decreased according to the current gamepad status.
- Vertical movement was calculated by applying a sudden vertical velocity and decreasing it every frame. This resulted in a jumping arc.

```
jumping_pressed:
    ; LDA satrina_y
    ; CLC
    ; SBC #$01
    ; STA satrina_y
    LDA #$0f
    STA satrina_y_velocity
    LDA #$00
    STA satrina_on_ground
    JMP exit
```

Code showing the upwards velocity being set.

How it was made Collision

- Boundaries were defined using X and Y positions. If satrina would approach these, the appropriate direction would become locked.
- Boundaries include the left border, the right border, the platform right border, the platform left border.
- In order for satrina to land on the platform, if she is within the platform x position range, the ground is shifted upwards.

```
PlatformCheckRight:
    LDA satrina_y
    CMP #$8f
    BCC move_in_direction
    LDA #$01
    STA LockRight
    JMP move_in_direction


PlatformCheckLeft:
    LDA satrina_y
    CMP #$8f
    BCC move_in_direction
    LDA #$01
    STA LockLeft
    JMP move_in_direction
```

Code showing the platform collision

Satrina examines the battlefield...

Thank you

GitHub Repository

- https://github.com/AcevedoC17/CIIC4082Project1