TERMINATOR 3D

Tim Mamtora
Robert Cassels

6.111 Final Project

TERMINATOR 3D

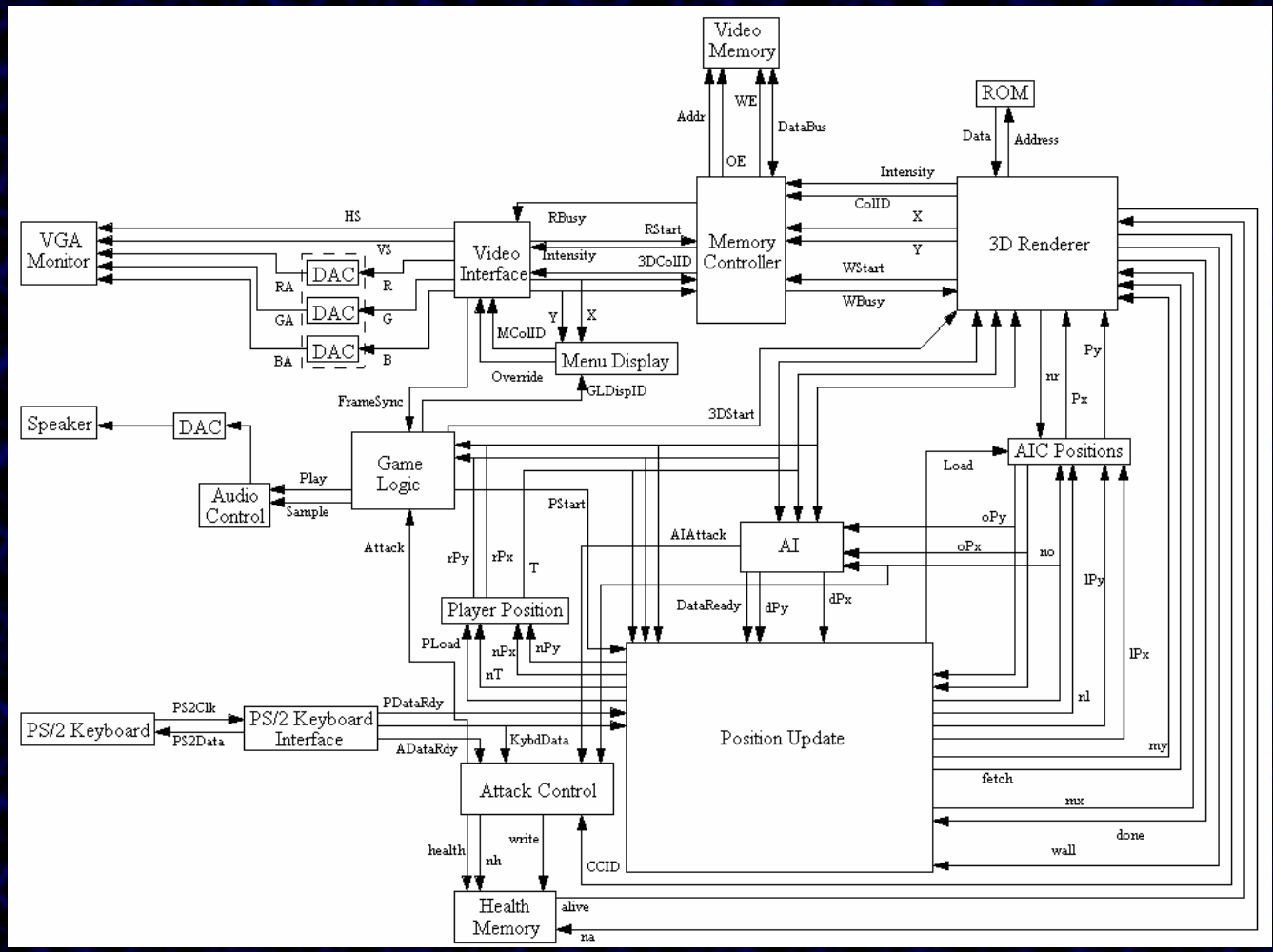
Game Overview

Something fun.....

- A third dimensional experience
- First person shoot-em-up
- Maze based level

- Your key objectives are....

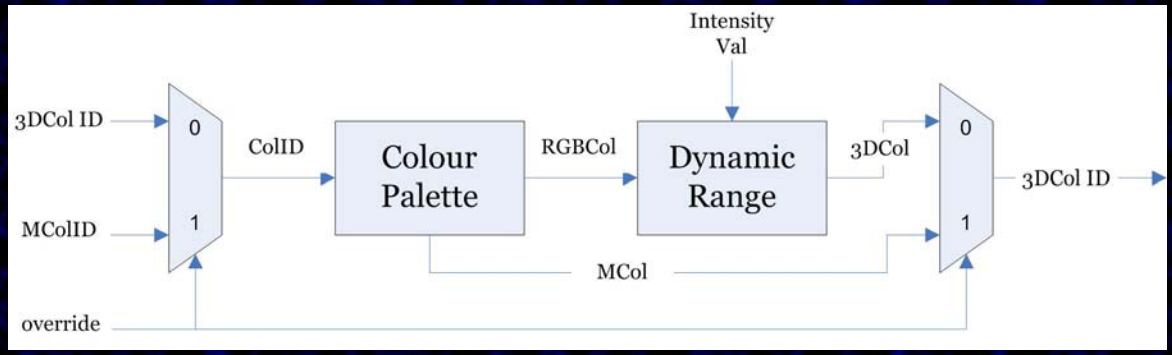
Block Diagram



User Interface

Something to look at.....

- Simple VGA interface to 320x240 resolution monitor:



- 8-bit colour ID supplied to Colour Palette LUT.
- Values scaled by Intensity Value.
- Three 8-bit DACs output 24-bit colour to Screen

TERMINATOR 3D

User Interface

Something to play with.....

- Keyboard control via PS2 interface:

- Controls to move and turn

TERMINATOR 3D

3D Ray-Casting

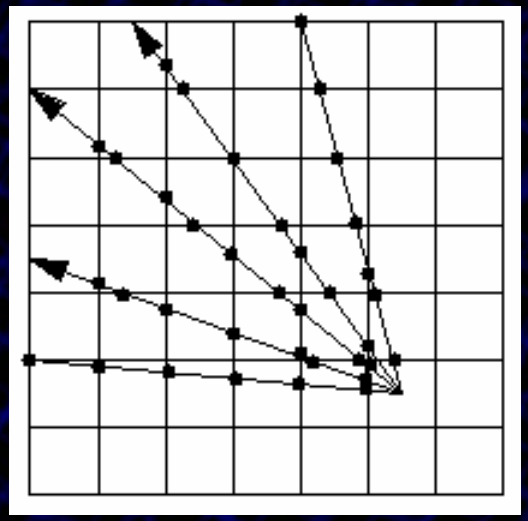
Advantages, Disadvantages and Limits:

- Faster than other 3D methods
- Worse quality
- Used in Wolfenstein 3D
- Has three major geometric constraints
    - Walls perpendicular to floors and ceilings
    - Floor and ceiling is flat
    - Walls made from equally sized cubes

TERMINATOR 3D

3D Ray-Casting

How?

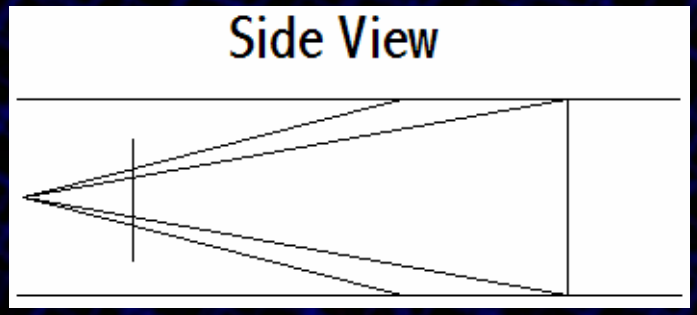


- Map consists of a 64 by 64 grid of squares
- Each square is subdivided into a 64 by 64 coordinate system
- Walls occupy a single large square
- Casts a single ray across a 2D map for each column of pixels

- Simplify - The ray can only hit a wall at a large square boundary
- Check each big square boundary for a wall

TERMINATOR 3D

3D Ray-Casting



- Viewing constraint - no looking up or down
- The distance a ray travels to a wall sets the height of the wall on the screen in pixels
- Walls are textured – must find which bit of the wall a pixel corresponds to
- Have already got which column the pixels are in from Ray-Casting
- Vertical texture coordinate is dependent on which pixel is chosen – scales with wall height
- Single colour for floors and another for ceilings

TERMINATOR 3D

Moving Through The Maze

Separate position data and position update modules:

- Player Position Data delivers player coordinates to the other game control blocks.
- AIC Position Data delivers each game character's coordinates to other game control blocks.
- Position update calculates new positions of both player and game characters and delivers these to position data, initiated by the game logic block.

TERMINATOR 3D

Opponents

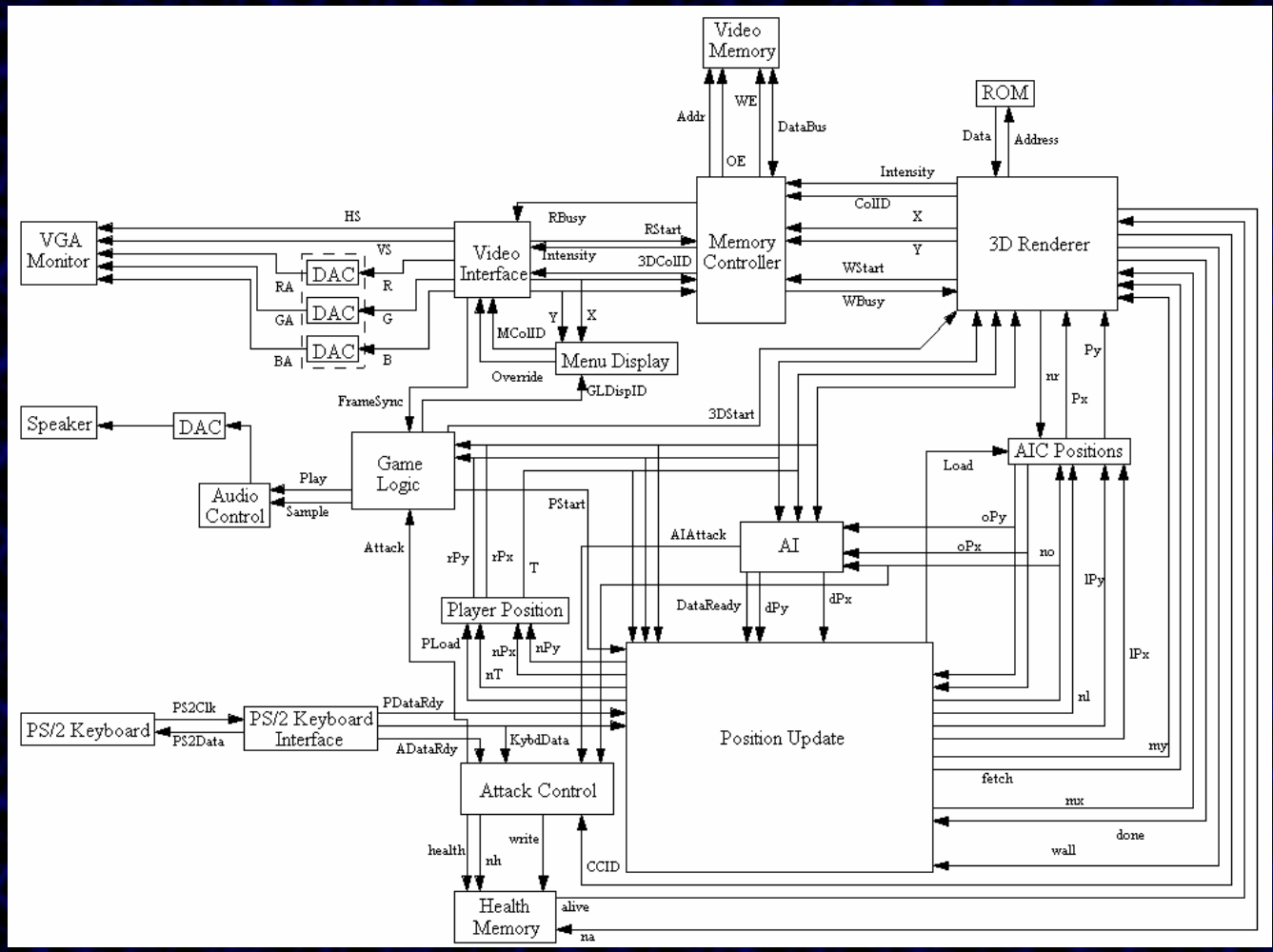
- A.I. Block controls the movements of the game characters.
- Proximity of characters governs their attack.
- Player may shoot at a character attacking them.
- Player and all characters have health points which are decremented in the event of an attack.
- Game logic coordinates 3D Rendering, Position Update and game start and end.

TERMINATOR 3D

Major Obstacles

- Amount of on chip ROM – Move it off chip
- Too many modules needing access simultaneously – Re-order processing steps so that accesses can be efficiently time division multiplexed
- VGA timing requires 25Mhz minimum, with 24-bit colour, DACs not good enough – get better ones

Questions?


Video Memory
ROM
Addr
WE
DataBus
OE
Data
Address
Intensity
ColID
X
Y
WStart
WBusy
RStart
RBusy
3DColID
HS
VS
R
G
B
RA
GA
BA
DAC
DAC
DAC
VGA Monitor
Video Interface
Memory Controller
3D Renderer
MColID
Menu Display
Override
GLDispID
FrameSync
3DStart
Speaker
DAC
Audio Control
Play
Sample
Game Logic
AIC Positions
Load
Px
Py
nr
AI
AIAttack
oPx
oPy
no
Attack
rPy
rPx
T
PStart
Player Position
DataReady
dPx
dPy
lPy
lPx
nl
PLoad
nPx
nPy
nT
PS/2 Keyboard
PS2Clk
PS2Data
PS/2 Keyboard Interface
PDataRdy
ADataRdy
KybdData
Position Update
my
fetch
mx
done
wall
Attack Control
health
write
nh
CCID
Health Memory
alive
na