

Object Position and Orientation Tracker

Andrew Lee & Tiffany Wang

6.111 Introductory Digital Systems Lab

Fall 2005

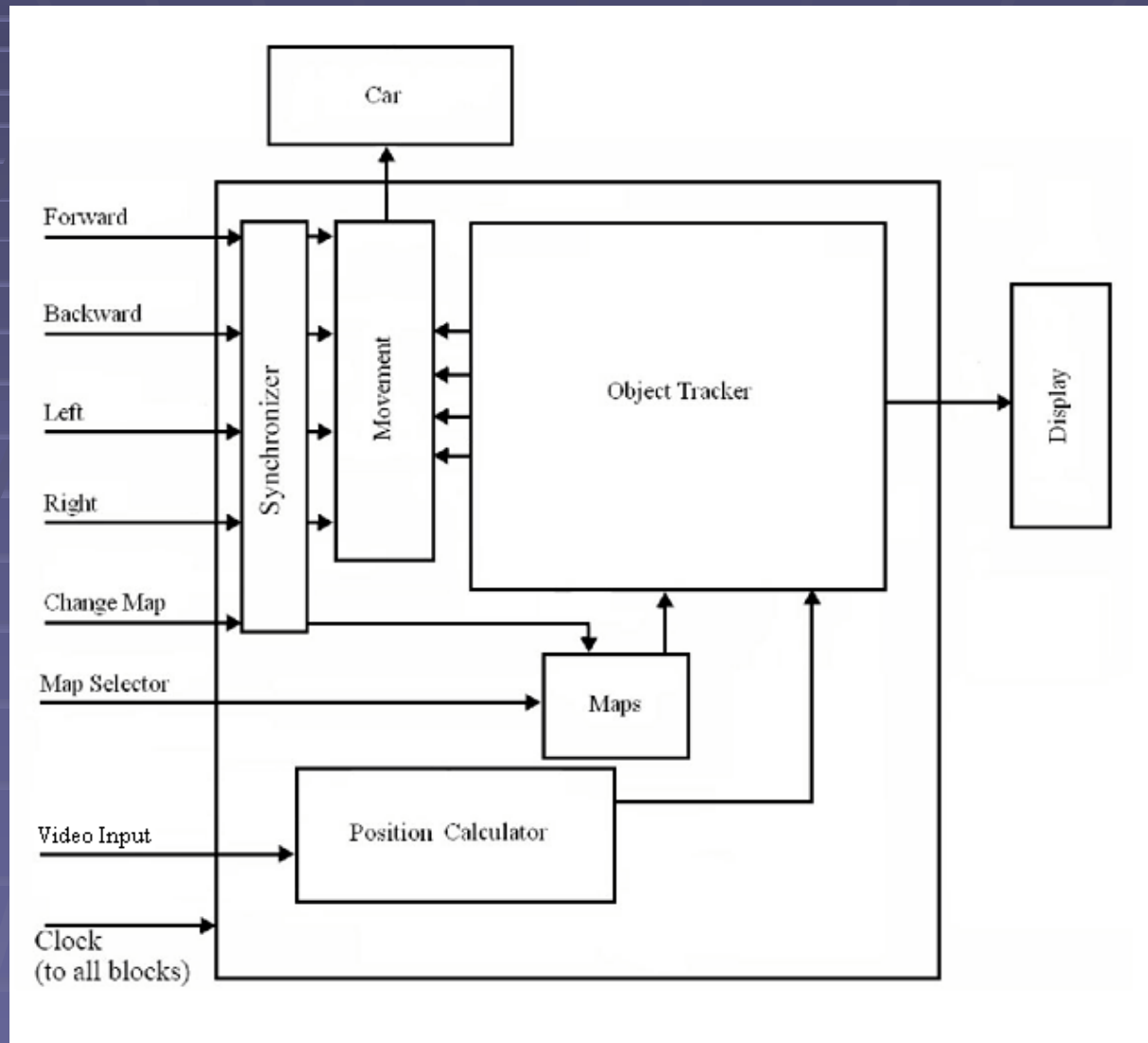
Presentation Agenda

- Overview
- Block Diagram
- Modules/Functionality
- Project Goals

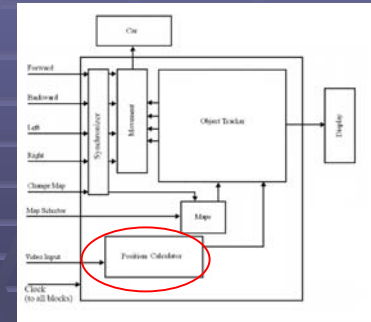
Overview

- Objective
 - digital tracking system used to calculate and display the exact position and orientation of an object within a predefined space
- Project Setup
 - object modeled by small remote controlled car
 - use of video processing to calculate position/orientation of car as it moves within space
 - multiple virtually pre-defined terrains to navigate
 - object position and terrain elements displayed on monitor

Block Diagram



Position Calculator



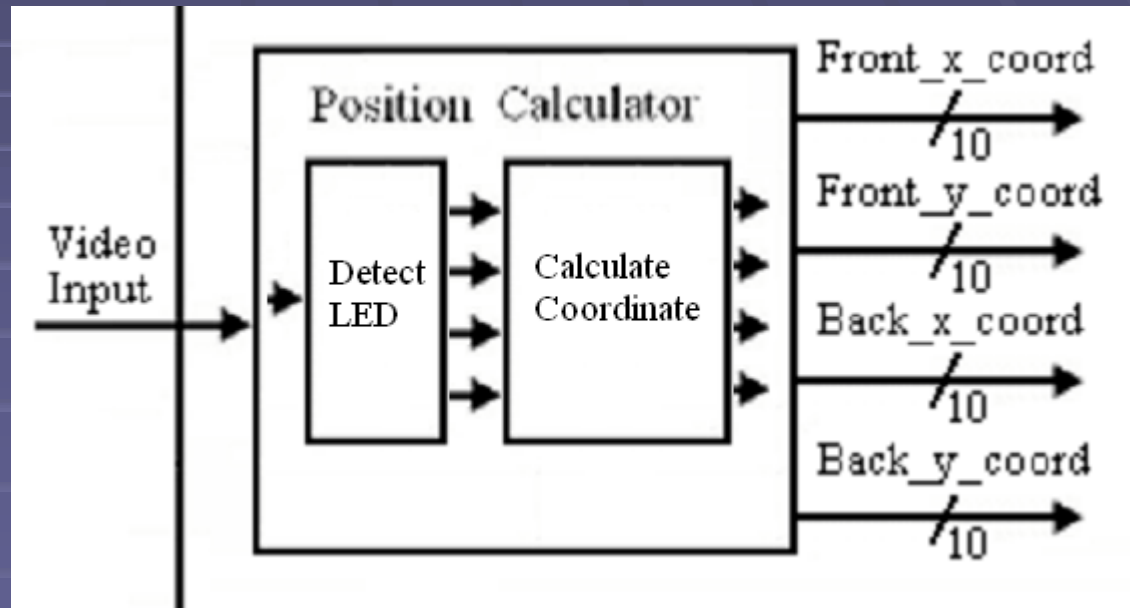
- Analyzes video image to detect LEDs
- determines coordinates of the car from the LED positions

Inputs:

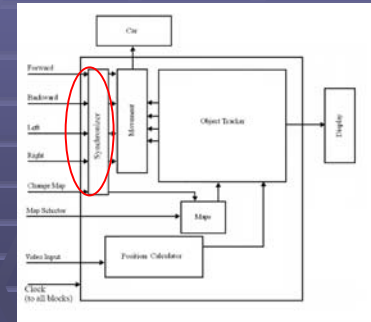
- Video Image

Outputs:

- Front_x_coord [9:0]
- Front_y_coord [9:0]
- Back_x_coord [9:0]
- Back_y_coord [9:0]



Synchronizer



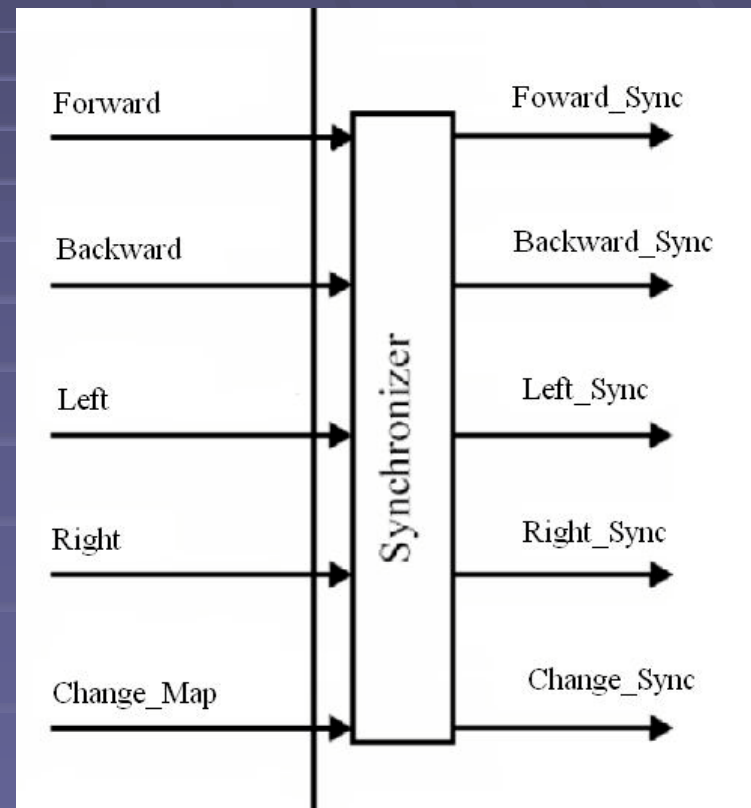
- Synchronize asynchronous button pushes to system clock

Inputs (from user):

- Forward
- Backward
- Left
- Right
- Change_Map

Outputs:

- Forward_Sync
- Backward_Sync
- Left_Sync
- Right_Sync
- Change_Sync



Maps

- manages user map inputs, notifies *Object Tracker* of selected terrain

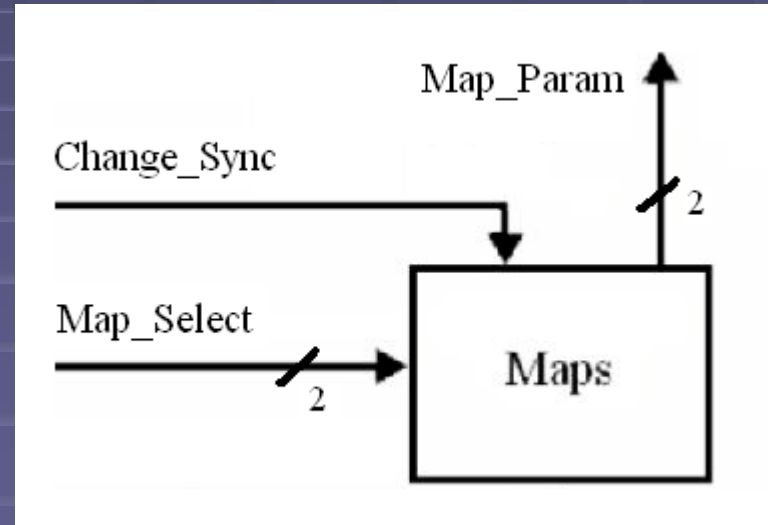
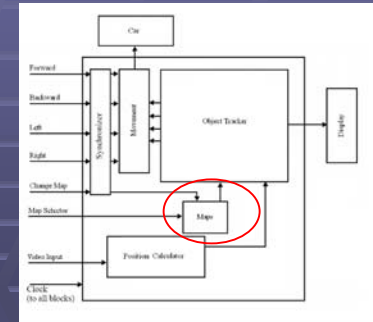
Inputs:

- Map_Select [1:0] (from user)
- Change_Sync (from *Synchronizer*)

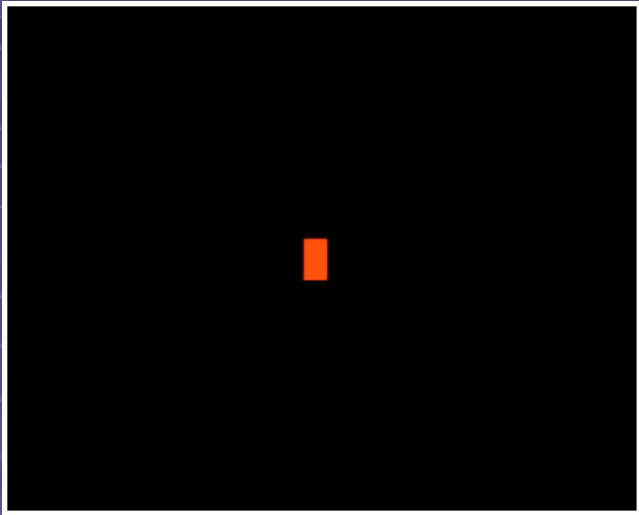
Outputs:

- Map_Param [1:0] (to *Object Tracker*)

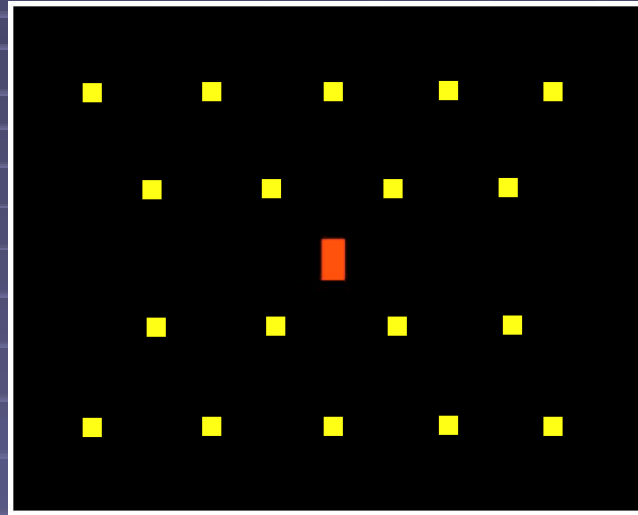
Parameter		
SWITCH 1	SWITCH 0	TERRAIN
0	0	open
0	1	maze
1	0	minefield
1	1	race track



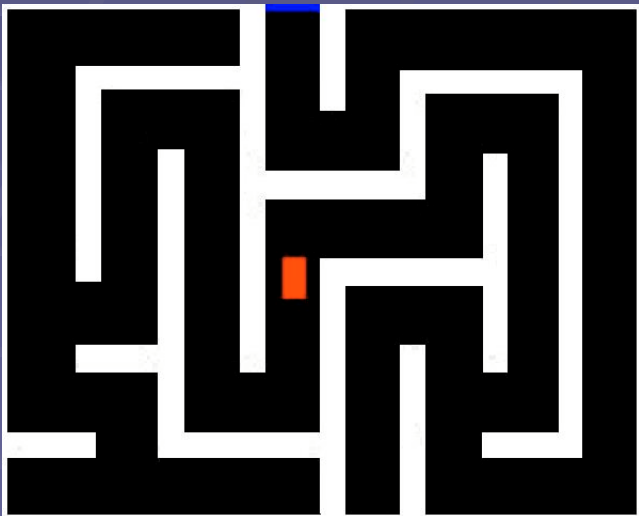
Maps



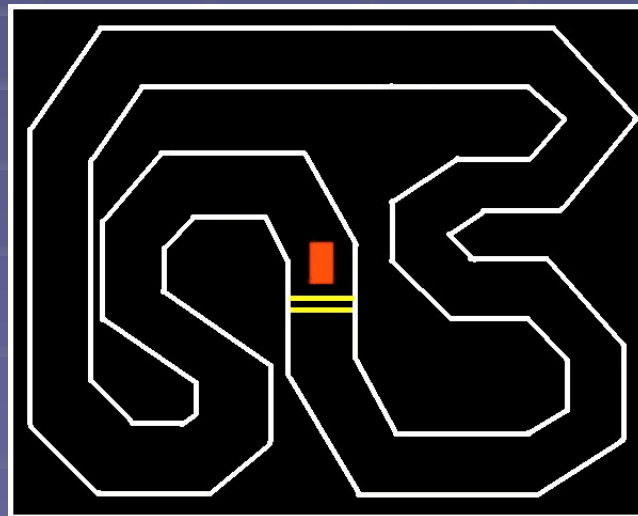
Open



Mine Field



Maze



Race Track

Object Tracker

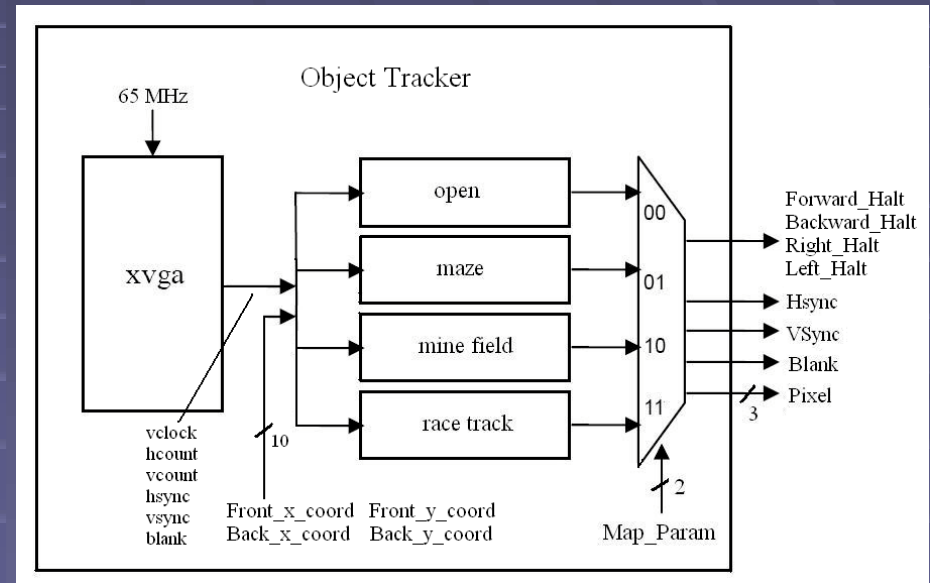
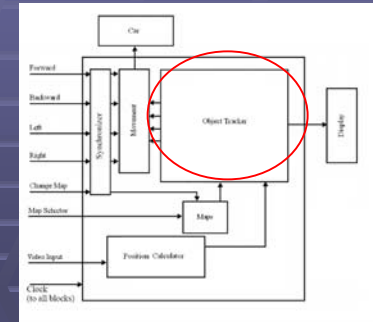
- contains xvga module and modules for each terrain
- provides feedback for car controller and screen display signals

Inputs:

- 65 MHz clock
- Map_Param [1:0] (from *Maps*)
- front_x_coord [9:0],
front_y_coord [9:0],
back_x_coord [9:0],
back_y_coord [9:0] (from *Position Calculator*)

Outputs:

- to *Movement*: Forward_Halt, Backward_Halt, Right_Halt, Left_Halt
- to *Display*: Vsync, Hsync, Blank, Pixel [2:0]



Terrain Modules

- holds specifications and element submodules for each terrain
- performs boundary checking
- determines feedback for car controller and screen display signals

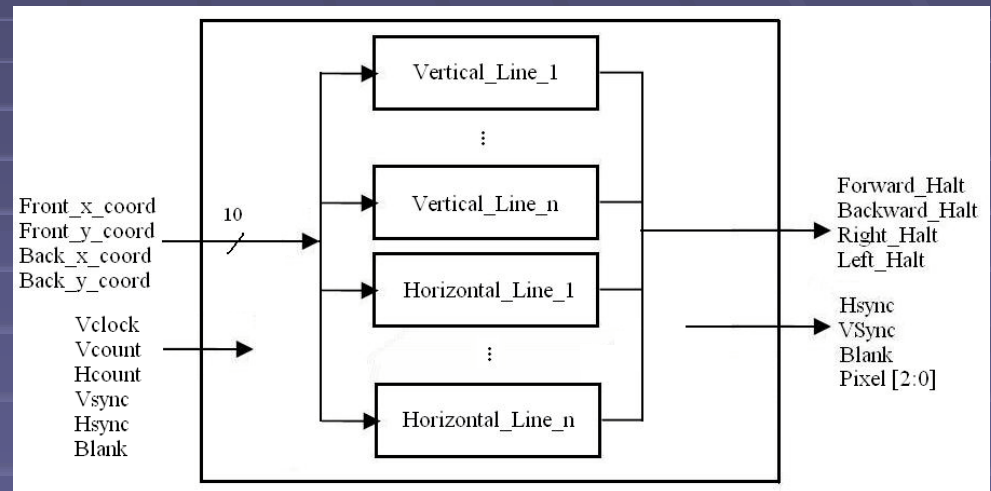
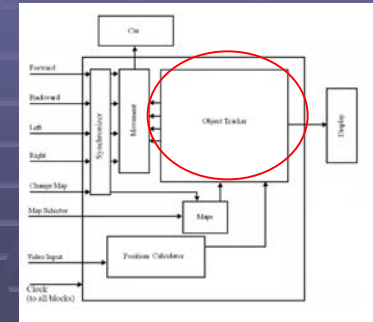
Inputs:

- display (from xvga): vclock, hcount, vcount, hsync, vsync, blank

- front_x_coord [9:0], front_y_coord [9:0], back_x_coord [9:0], back_y_coord [9:0] (from *Position Calculator*)

Outputs:

- control feedback: Forward_Halt, Backward_Halt, Right_Halt, Left_Halt
- screen display: Vsync, Hsync, Blank, Pixel [2:0]



Object Tracker

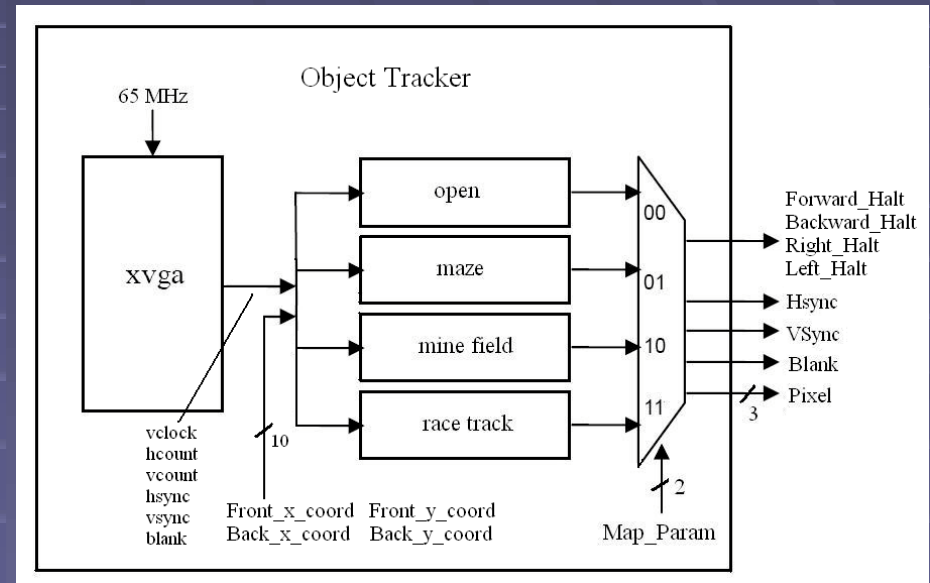
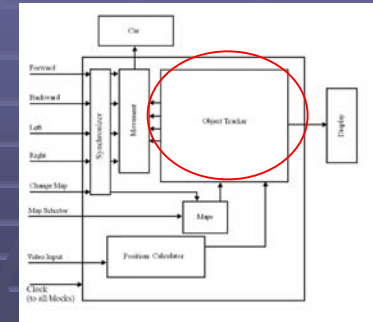
- contains xvga module and modules for each terrain
- provides feedback for car controller and screen display signals

Inputs:

- 65 MHz clock
- Map_Param [1:0] (from *Maps*)
- front_x_coord [9:0],
front_y_coord [9:0],
back_x_coord [9:0],
back_y_coord [9:0] (from *Position Calculator*)

Outputs:

- to *Movement*: Forward_Halt, Backward_Halt, Right_Halt, Left_Halt
- to *Display*: Vsync, Hsync, Blank, Pixel [2:0]



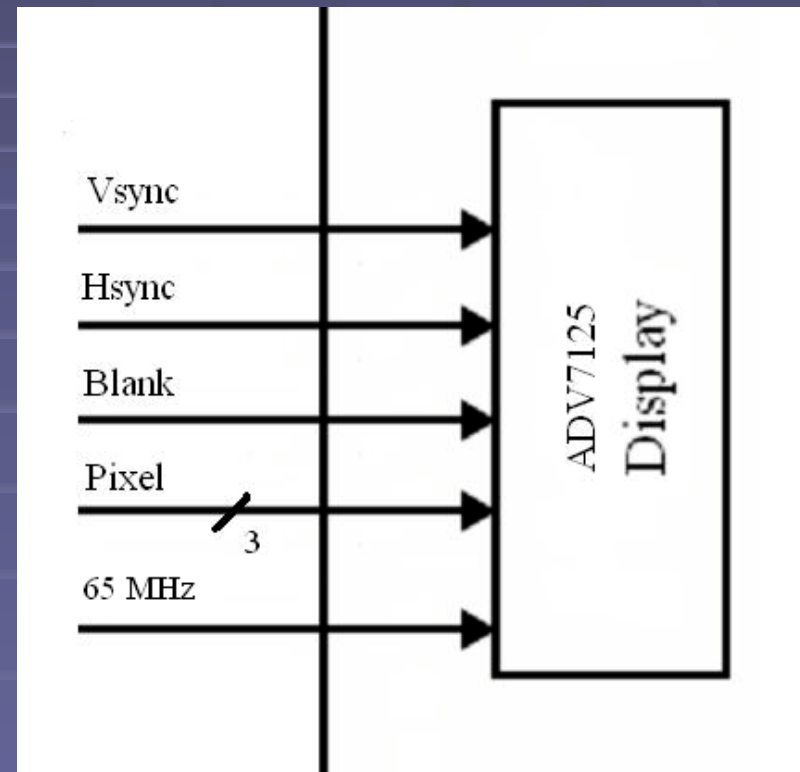
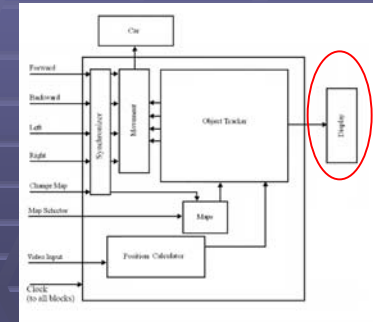
Display

- ADV7125 Triple 8-bit high speed video DAC
- produce correct analog signals based on inputs

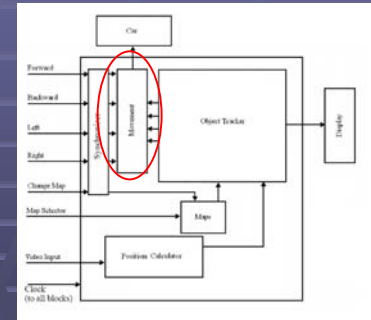
Inputs:

- Vsync (from *Object Tracker*)
- Hsync (from *Object Tracker*)
- Blank (from *Object Tracker*)
- Pixel [2:0] (from *Object Tracker*)
- 65 MHz clock

Outputs analog signals to monitor



Movement



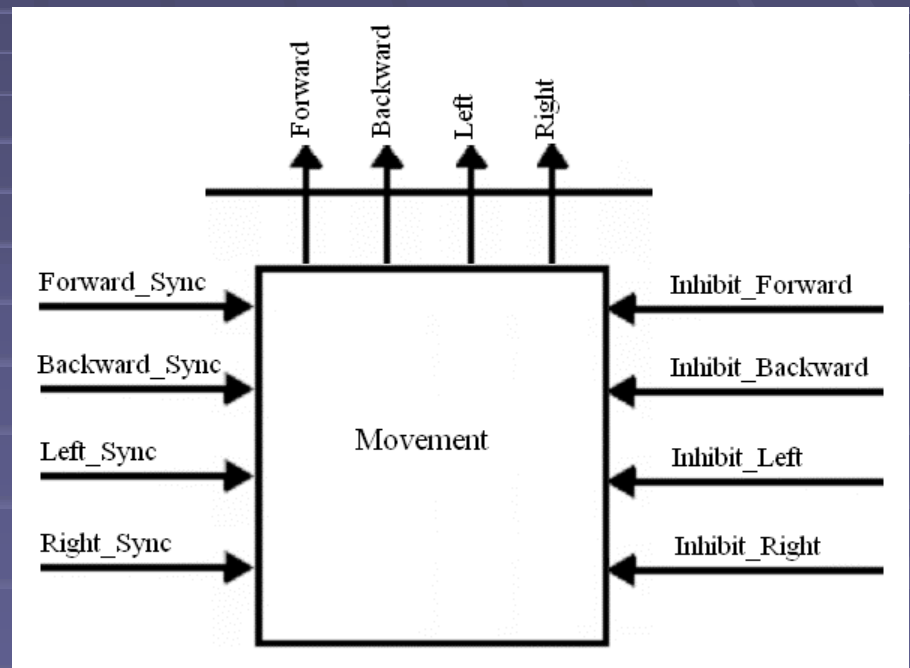
- Controls the movement of the car

Inputs:

- Forward_Sync (from *Synchronizer*)
- Backward_Sync (from *Synchronizer*)
- Left_Sync (from *Synchronizer*)
- Right_Sync (from *Synchronizer*)
- Inhibit_Left (from *Object Tracker*)
- Inhibit_Right (from *Object Tracker*)
- Inhibit_Forward (from *Object Tracker*)
- Inhibit_Backward (from *Object Tracker*)

Outputs:

- Forward (To *Car*)
- Backward (To *Car*)
- Left (To *Car*)
- Right (To *Car*)



Project Goals

- Accurately sense and calculate the position and orientation of the car
- Accurately display the car and terrain elements on screen
- Functional terrain selection and display
- Proper boundary checking and feedback to controller