Remote programmable FPGA - LED cube

With eDiViDe

About me

- Sam Thys
- Student at Thomas More Kempen
  - Campus Geel
- Electronics-ICT
  - Professional bachelor Electronics
  - 3th year

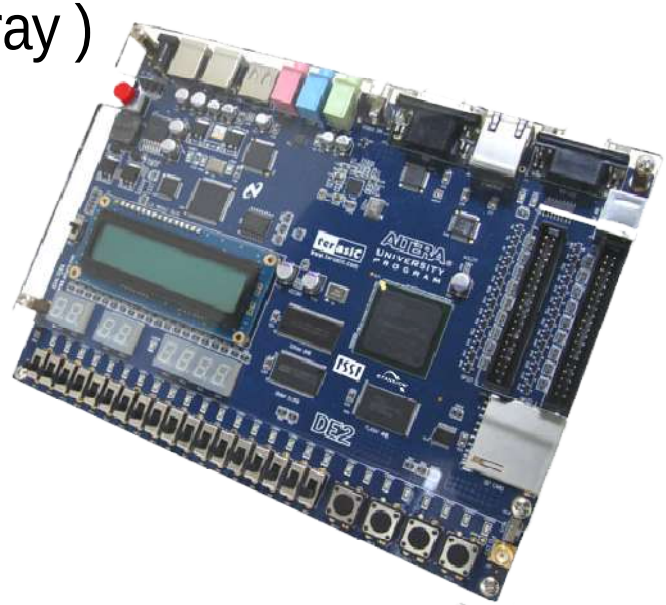
Internship

- Internship at Thomas More Kempen
  - Campus Geel
    - Electronics Lab

Problem

Problem - Background

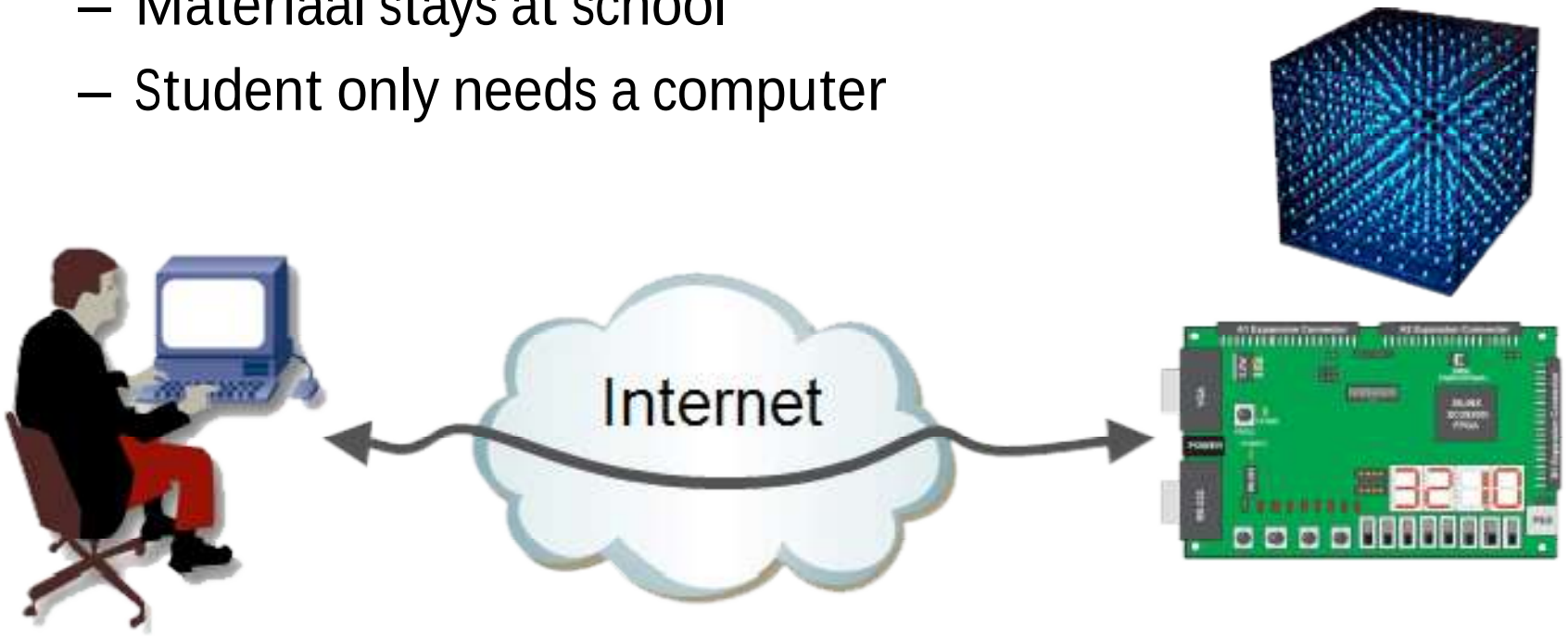
- Programmable systems
  - FPGA (Field Programmable Gate Array)
  - VHDL (VHSIC Hardware Description Language)



- No material to exercise at home
  - To expensive
  - Lend material
    - Material can break

Problem - Goal

- A remote programmable FPGA
  - LED cube
  - Materiaal stays at school
  - Student only needs a computer



Solution

eDiViDe

eDiViDe

- European Digital Virtual Design Lab
  - www.edivide.eu
- Website
  - Program
  - Control
  - Monitor
- Developed at KHLim, Belgium
  - Nele Mertens
  - Jochen Vandorpe
- Erasmus
  - Lifelong Learning Programme (LLP)

eDiViDe

- Different setups
  - Multiple colleges and universities
  - Europa
    - Belgium
    - Germany
    - Norway
    - Slovakia

eDiViDe - Usage

- VHDL Template

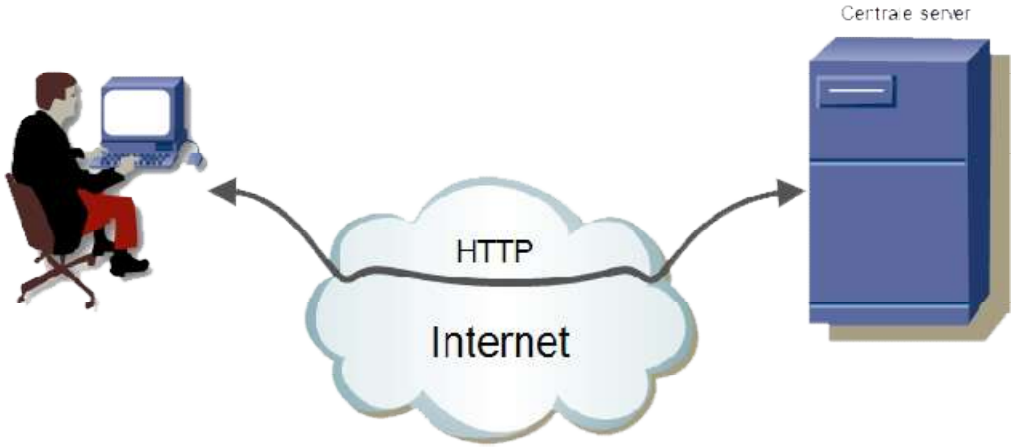
```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.STD_LOGIC_ARITH.ALL;
use IEEE.STD_LOGIC_UNSIGNED.ALL;
library WORK;
use WORK.all;

entity LedCube_Exercise_01 is Port (
Clk : in STD_LOGIC; -- 50 MHz system Clock
enable : in STD_LOGIC; -- Switch Input
reset : in STD_LOGIC; -- Button Input, High Active -> '1' when pressed
colorSel : in STD_LOGIC_VECTOR (1 downto 0); -- Switch vector Input
colorOut : out STD_LOGIC_VECTOR ( 1 downto 0);
x : out STD_LOGIC_VECTOR ( 2 downto 0); -- X Coordinate, value 0 - 7
y : out STD_LOGIC_VECTOR ( 2 downto 0); -- Y Coordinate, value 0 - 7
z : out STD_LOGIC_VECTOR ( 2 downto 0); -- Z Coordinate, value 0 - 7
FreeIn : in STD_LOGIC_VECTOR ( 8 downto 0); -- Free to use Input
FreeOut : out STD_LOGIC_VECTOR ( 8 downto 0)); -- Free to use Output
end LedCube_Exercise_01;

architecture Structural of LedCube_Exercise_01 is begin


end Structural;
```

eDiViDe - Usage

- Webinterface

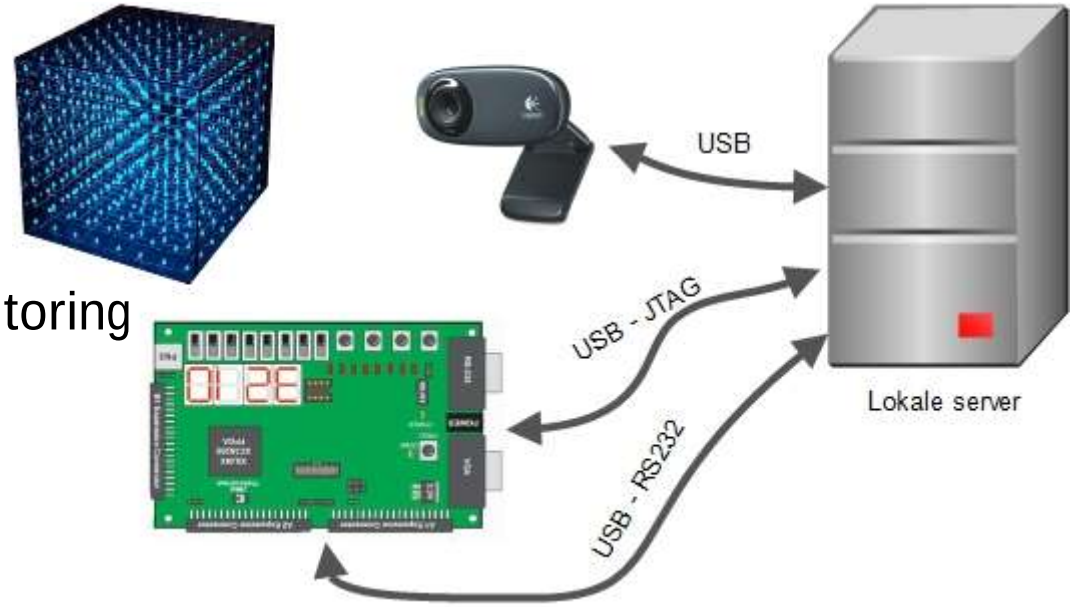
eDiViDe - Operation

- Central server
  - KHLIM
  - Webserver
    - Interface
    - database
  - Scripts
    - Communication with local sever



eDiViDe - Operation

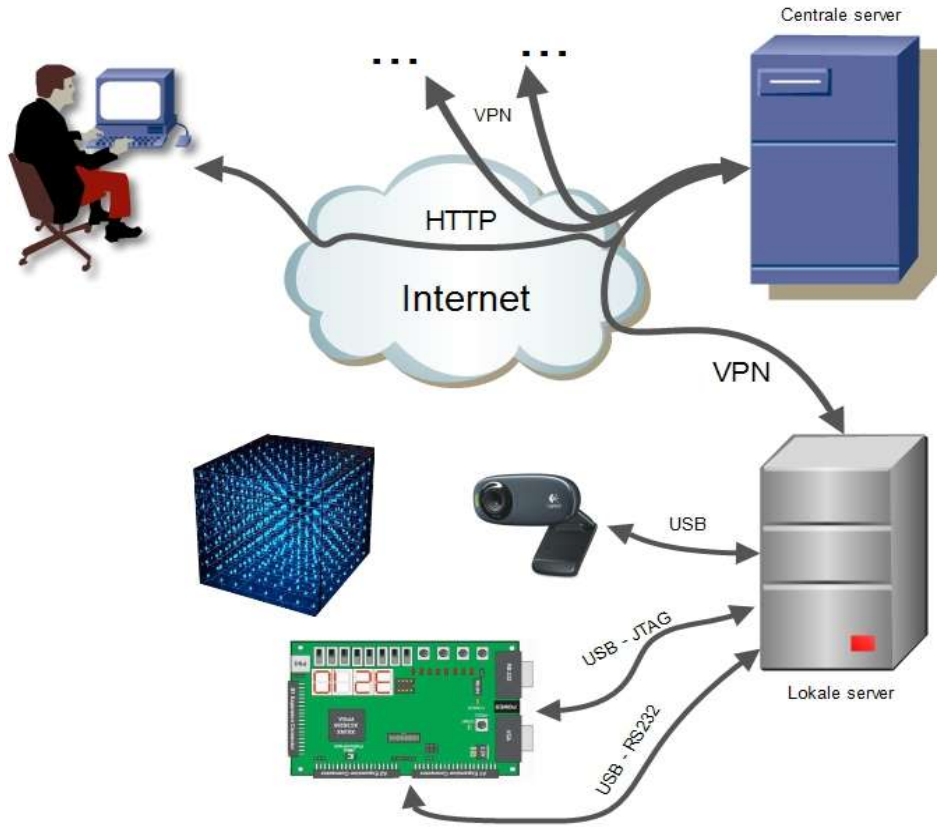
- Local Setup
  - FPGA development board
  - Webcam
  - USB – RS232
    - Control and monitoring
  - USB – JTAG
    - Programming
  - LED Cube



eDiViDe - Operation

- Communication
  - Virtual Private Network

KHLIM

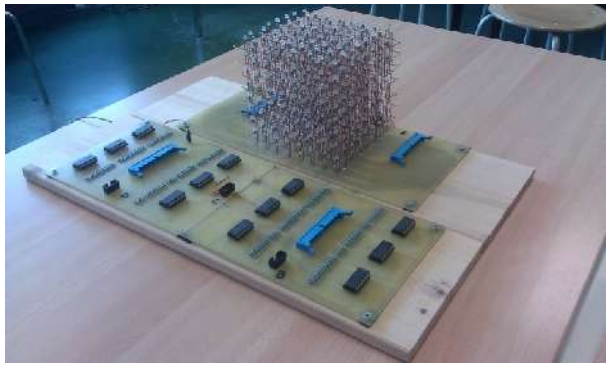


Thomas More Geel

Local Setup

Local setup

- LED cube (RGB 8 x 8 x 8)
  - Greduation project 2010
  - Original control
    - Microcontroller
- Digilent Spartan 3
  - Xilinx Spartan x3s200
  - eDiViDe Compatible



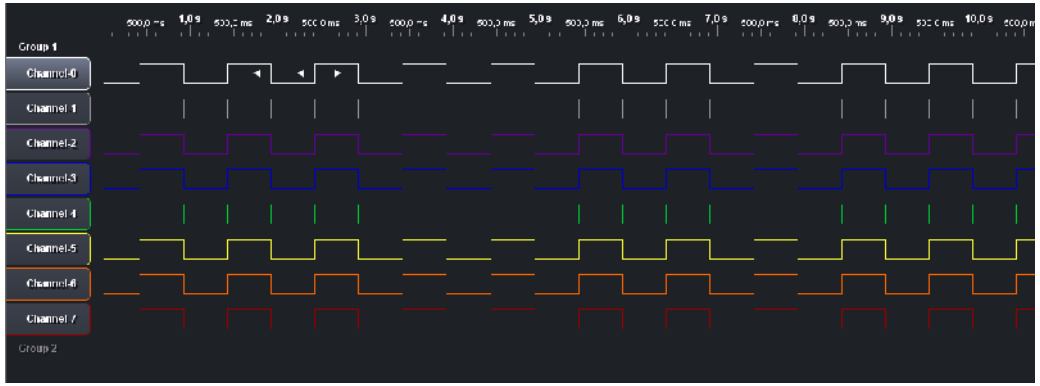
THOMAS MORE EDIVIDE

Lokale Setup

- LED Cube
  - 32 Inputs
    - 24 Horizontal selection
    - 8 Vertical selection
  - Works on 5V

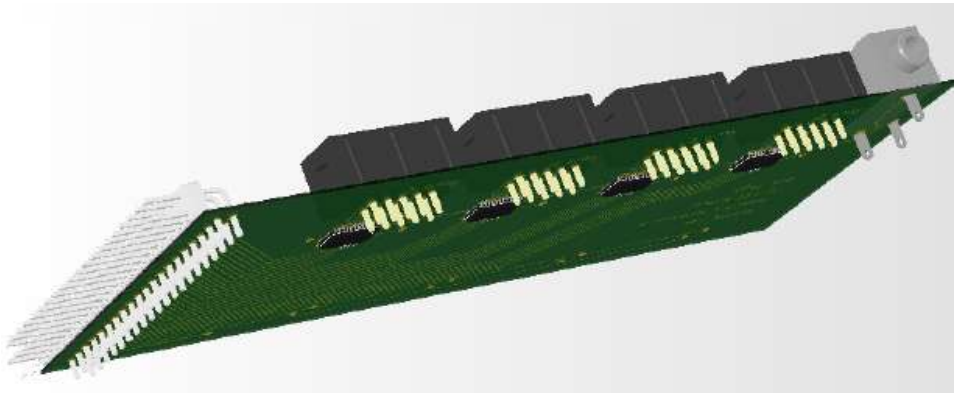
Lokale opstelling

- Digilent Spartan 3 ontwikkelbord
  - 32 Connections
    - Looked up in datasheet
      - Shared with onboard components
    - Tested with a logic analyzer
  - Works on 3,3V



Lokale Local setup

- Connection FGPA to LED cube
  - Voltage translation 3,3V -> 5V
  - Current buffer
  - Isolation
  - 4 x 74LVX4245



VHDL Design

VHDL Design

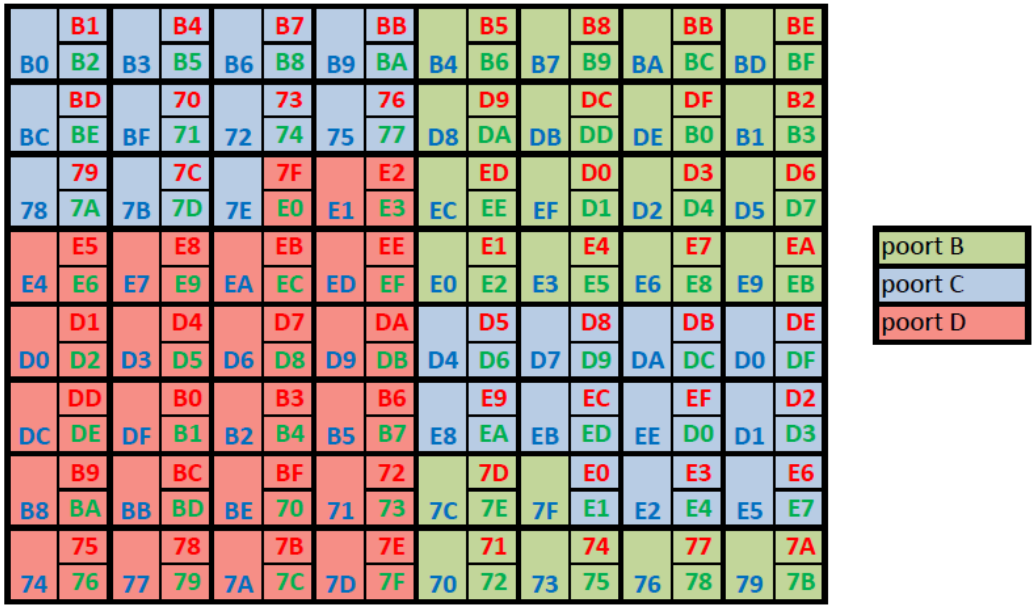
- Exercises
  - Start simple
  - Systematically get harder

  - Xilinx ISE Design Suite 14
    - Xilinx Spartan3 – XC3S200
    - Compilation
  - Modelsim PE
    - Simulatiion

THOMAS MORE EDIVIDE

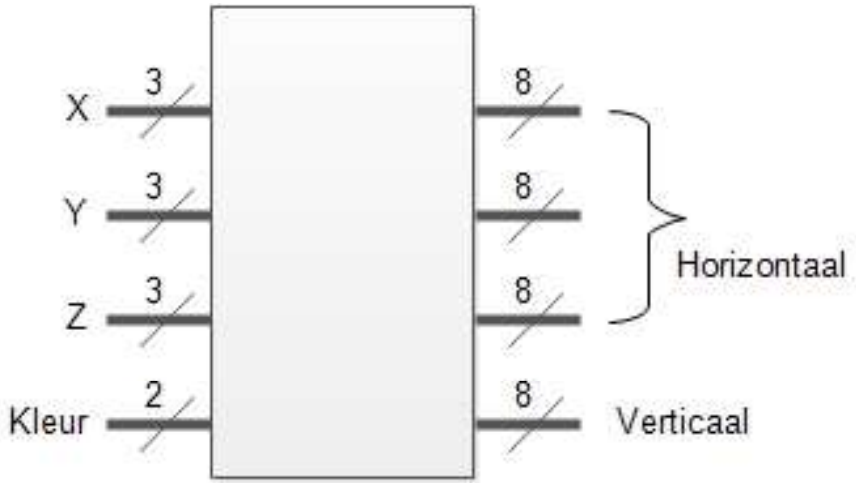
VHDL Design

- Control of the LED cube
  - Original system is complex and not very logical
    - Not educational



VHDL Design

- Conversion table
  - Inputs
    - Coordinates
      - X,Y,Z
      - 0 to 7
    - color
      - Blue, Green, Red, Nothing
      - 0 to 3
  - Outputs
    - Data for the LED cube

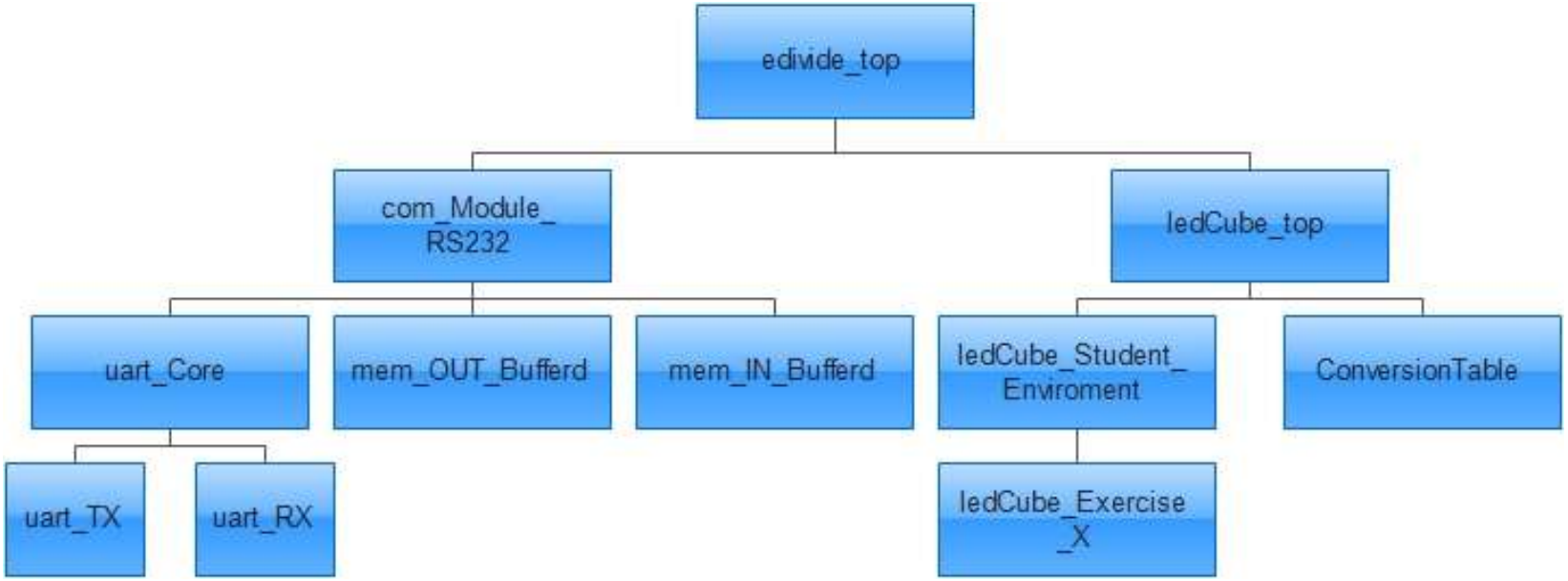


VHDL Design

- 6 Exercises
  - Running Light
  - Running plane
  - Running Plane in 3 directions with color change
  - Dimmable Cube
  - Ambiant cube
  - 3D Symbol

VHDL Design

- Conversion to eDiViDe VHDL structure

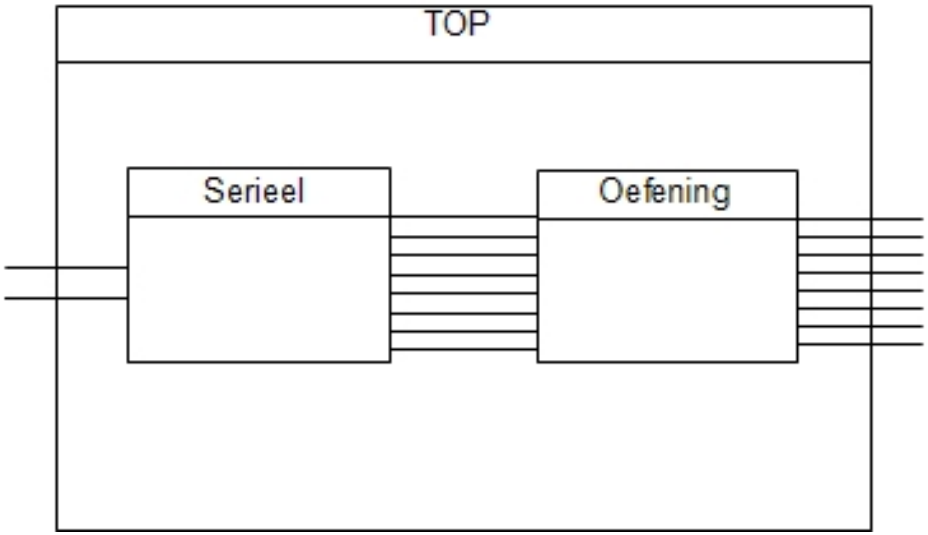


VHDL ontwerp

- Conversion to eDiViDe VHDL structure
  - LedCube Exercise X
    - Exercise of the student
  - LedCube Student Enviroment
    - Filtering of inputs and outputs
    - Exercise dependant modules
  - LedCube Top
    - Setup dependant modules
      - Conversion table
  - Edivide Top
    - Inputs and outputs to serial communication
    - Inputs and outputs to physical connection

VHDL Design

- USB – RS232
  - Control IO
  - 128 Bits (16 bytes)
  - Internal conversion
    - Module in the FPGA



Webinterface

Webinterface

- Webinterface
  - Control elements
    - Switches
    - Pushbuttons
    - Digital potentiometer
  - Display elementen
    - Video
    - Leds

Webinterface

- Control
  - Javascript-Function
    - Adjusts data in variables
    - Collects all variables variabelen
  - Passes on to a PHP-script
    - Write data to a file on the centrale server

Webinterface

- Dislay of status
  - Refresh every 500ms
  - PHP-script
    - Reads data from a file on the centrale server
  - Javascript function
    - Writes data to the corresponding variables
    - Adjusts the display elements

Webinterface

- Video display
  - Javascript
    - JWPlayer
  - RTMP Stream
    - Real Time Messaging Protocol
    - Audio, Video
    - Flash Player
    - TCP

Webinterface

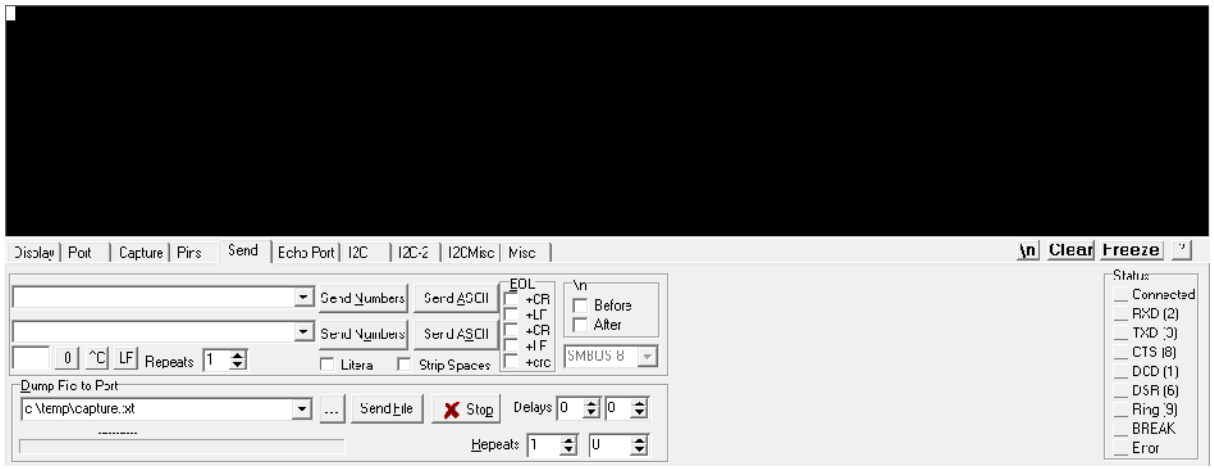
- Local test of the interface
  - Host with Wamp Server
    - Apache
    - PHP
      - Possebility to execute PHP
    - MySQL

Webinterface

- Local testing of the interface
  - Mozilla Firefox
    - FireBug Plugin
      - Check of functions get called
      - Checking data and variables

Webinterface

- Local testing of the interface
  - Realterm
    - Serial terminal
    - Write file to serial port



Local server

Local Server

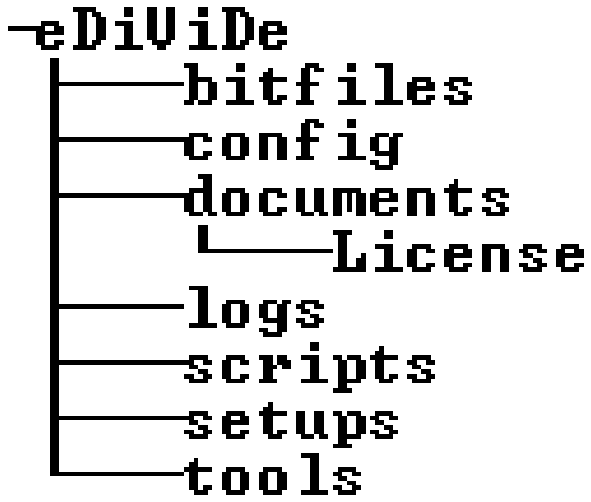
- Hardware
  - Chosen because of reliable operation
    - Server
      - Dell PowerEdge T110 II
    - Webcam
      - Logitech C310
    - USB-Serial cable
      - FTDI

Local Server

- Software
  - Linux operating system
    - Debian 6.0.5 amd64
  - Xilinx Tools
    - Xilinx ISE embedded
      - Compilation of VHDL code
    - IMPACT
      - Programming Xilinx FPGA
  - Adept
    - Programmeren with the Digilent JTAG programmer

Local Server

- Folder structure
  - User : eDiViDe-khlim
    - Home folder

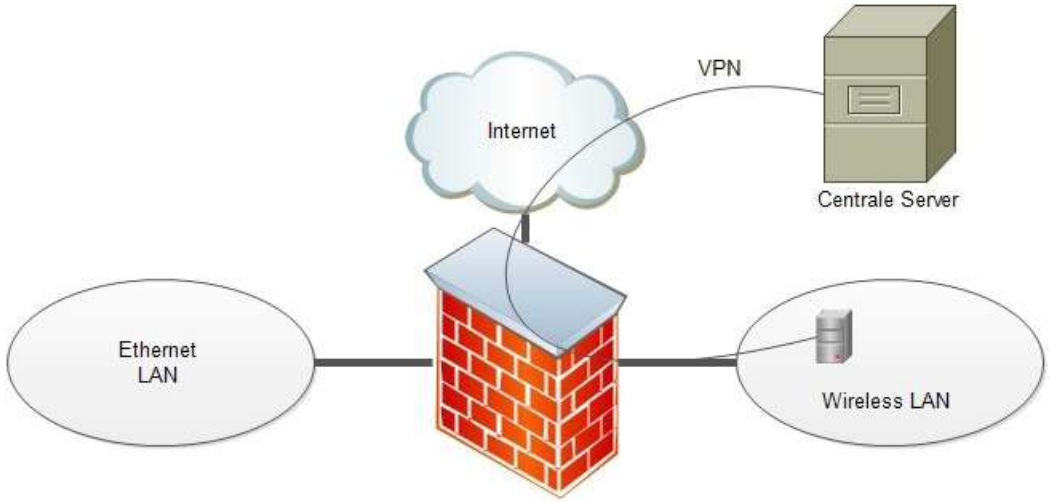


Local Server

- Configuration
  - USB rules for device management (UDEV)
    - USB-JTAG programmer
    - FTDI USB-Serial cable
  - Configuration files
    - contain serialnumbers of the devices
      - USB-JTAG
      - Webcam
      - FTDI

Local Server

- VPN connection
  - OpenVPN
  - Server on Wireless Lan



Local Server

- Additional software
  - Installed by KHlim
    - VideoStream
      - Webcam
    - ComPortServer
      - Serial communication

Final Testing

- setting up the webinterface
  - Video
    - Starting the video software
    - Live videostream
  - Control
    - Sending data (mozilla firefox, firebug plugin)
    - Manual control from local server
    - Starting the ComPort server

Implementation

- Explanation for on the website
  - General explanation about the setup
  - explanation about the exercises
  - Downloading VHDL templates

Website

3D LED Cube

The "3D LED Cube" setup is developed and hosted at Thomas More Kempen, Campus Geel in Belgium. The setup is built upon a Digilent Spartan 3 Development board which contains a Xilinx X3S200 FPGA.

The LED cube is made of 512 RGB LEDs in an 8x8x8 cube. The purpose of this setup is to enable users to develop basic 3D animations in the LED cube. The cube is controlled with 12 decoders which are connected to 32 pins of the A2 expansion header on the development board. The data sent to the decoders is generated automatically, based on XYZ coordinates and a color output.

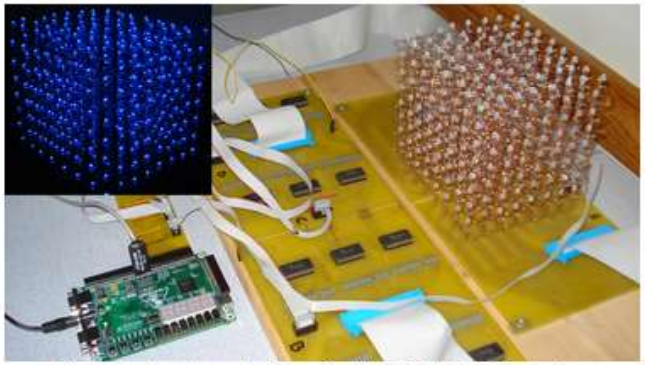


Figure 1: Hardware platform for the "3D LED Cube" setup

The coordinates are arranged so that the origin is in the lower left corner of the cube.
By increasing the X-coordinate the LED moves to the right.
When the Y-coordinate increases the LED moves away in the depth.
And by increasing the Z-coordinate, the LED moves up.

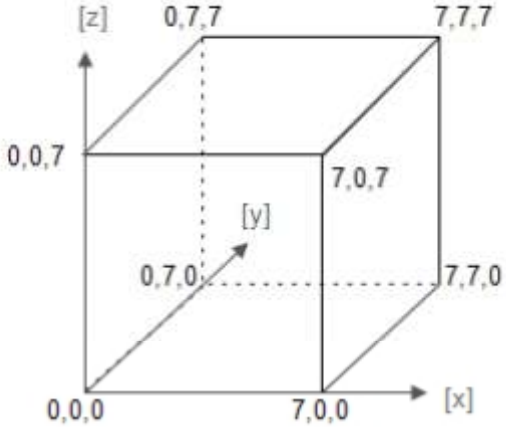


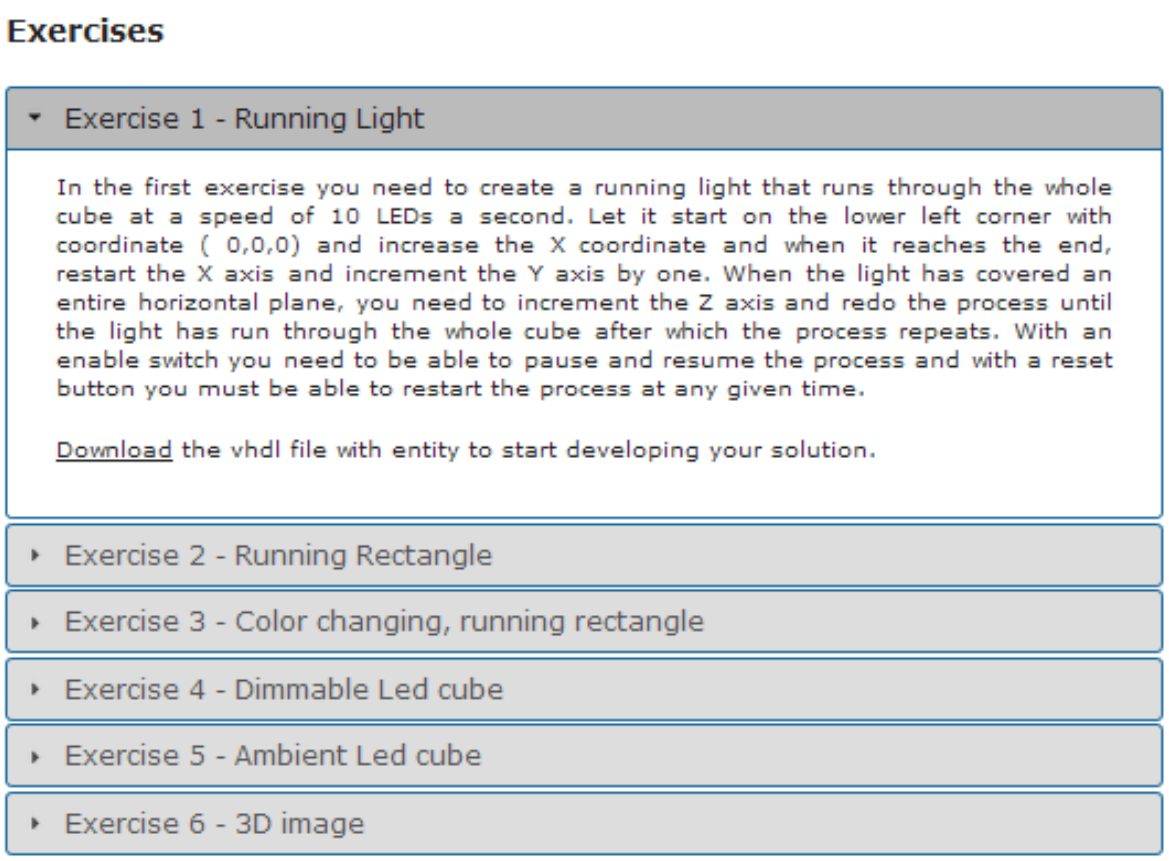
Figure 2: Coordinates for the 3D LED Cube

Since the cube consists of RBG LEDs, the color of the LEDs can be changed.
The color of the LED is dependent on the data at the "colorOut"-output.

	ColorOut 1	ColorOut 2
Blue	0	0
Red	0	1
Green	1	0
Off	1	1

THOMAS MORE EDIVIDE

Website


Exercises
Exercise 1 - Running Light
In the first exercise you need to create a running light that runs through the whole cube at a speed of 10 LEDs a second. Let it start on the lower left corner with coordinate (0,0,0) and increase the X coordinate and when it reaches the end, restart the X axis and increment the Y axis by one. When the light has covered an entire horizontal plane, you need to increment the Z axis and redo the process until the light has run through the whole cube after which the process repeats. With an enable switch you need to be able to pause and resume the process and with a reset button you must be able to restart the process at any given time.
Download the vhdl file with entity to start developing your solution.
Exercise 2 - Running Rectangle
Exercise 3 - Color changing, running rectangle
Exercise 4 - Dimmable Led cube
Exercise 5 - Ambient Led cube
Exercise 6 - 3D image

Summary

- Goal :
  - A remote programmable FPGA-LedCube
- Solution :
  - Collaboration with eDiViDe

THOMAS MORE EDIVIDE

Summary

- Research eDiViDe operation and functioning
- Reseach LedCube
- Connecting the LedCube and FPGA
- Writing VHDL code
- Developing a Webinterface
- Configure the Local server
- Final Testing

Thank you for your Attention