Final Project Presentation

Group Project Work

M.Tech. FT-15

Group Members

Sl.No.	Reg. No.	Name of the Student	Department
1	15ETEC033002	Bharath Ranganath B K	ECE
2	**15ETEC033003**	**Dewang Shukla**	**ECE**
3	15ETEC033004	Gurubasarajaiah N M	ECE
4	15ETEC033006	Tousif Ahmed	ECE
5	15ETEC033008	Naresh Gowda M	ECE
6	15ETEC033010	Prakruthi U S	ECE
7	15ETEC033014	Shahmustafa Mujawar	ECE

- **Title of the Project:** FPGA Based Dust Monitoring System

- **Mentor:** Mr. Abdul Imran Rasheed

- **Place of Work:** Ramaiah University Of Applied Sciences

Outline

- Introduction
- Motivation
- Title and Aim
- Objectives
  - Methods and Methodology
- Problem Solving
  - Project Concept, Design, Development or Implementation, Testing
- Outcomes
  - Prototype
- Conclusion
- Division of Tasks
- Project Schedule
- References

Introduction

- Air pollution is regarded as the pollution that has the greatest impact on the environment
- Global warming phenomena that threaten our earth in recent years is closely related to the level of air pollution
- Particulate matter is defined by the United States Environment Protection Agency (EPA) has an air- suspended mixture of both solid and liquid particles
- Dust monitor is a dust concentration measurement system that makes use of the most modern technology available to transmit real time dust values of environment

Motivation

- The basic concept behind this project is to develop Dust Concentration Measurement System
- It utilizes IoT based platform to monitor real-time dust data by connecting with HTTP Server and Android App.
- The end user with the android app device will be able to observe dust concentration in a specific area or zone
- The requirement of this system is to have Nexys-4 Artix-7 FPGA, Digilent Pmod Wifi, Sharp Dust Sensor, Micro SD Card, Host Router, Android App Version 6

Motivation

- Air pollution is hazardous to human health which may cause lungs, respiration related diseases
- There is need for developing a system which is capable of monitoring particulate matter in the air to control air pollution
- In this project we developed a FPGA based dust monitoring system

Figure 1. Air pollution effects

Title and Aim

Title : FPGA Based Dust Monitoring System

Aim : The aim of this project is to develop IoT based dust density monitoring system using FPGA.

Objectives

- To perform literature survey on existing dust monitoring system, dust sensors, Wi-fi interface and memory interface with FPGA, Android Application interface, webpage development
- To develop and configure interface unit for Wi-fi module, dust sensor, ADC, SD card
- To develop and configure Android application for the proposed dust monitoring system
- To develop and configure webpage with http server
- To configure http server with FPGA, integrate and verify the proposed dust monitoring system

Methods and Methodology

- Literature survey on dust sensors, Wi-fi interface and memory interface with FPGA, Android Application interface, webpage development
- Specifications of the selected sensor, wi-fi module has been identified
- Functional verification of sensor has been performed and the signal conditioning circuit is built in simulation tool and verified
- Configuration and implementation of ADC, Wi-fi module, webpage with server
- SD Card and Android application configuration for the proposed system

Block Diagram of Proposed Dust Monitoring System

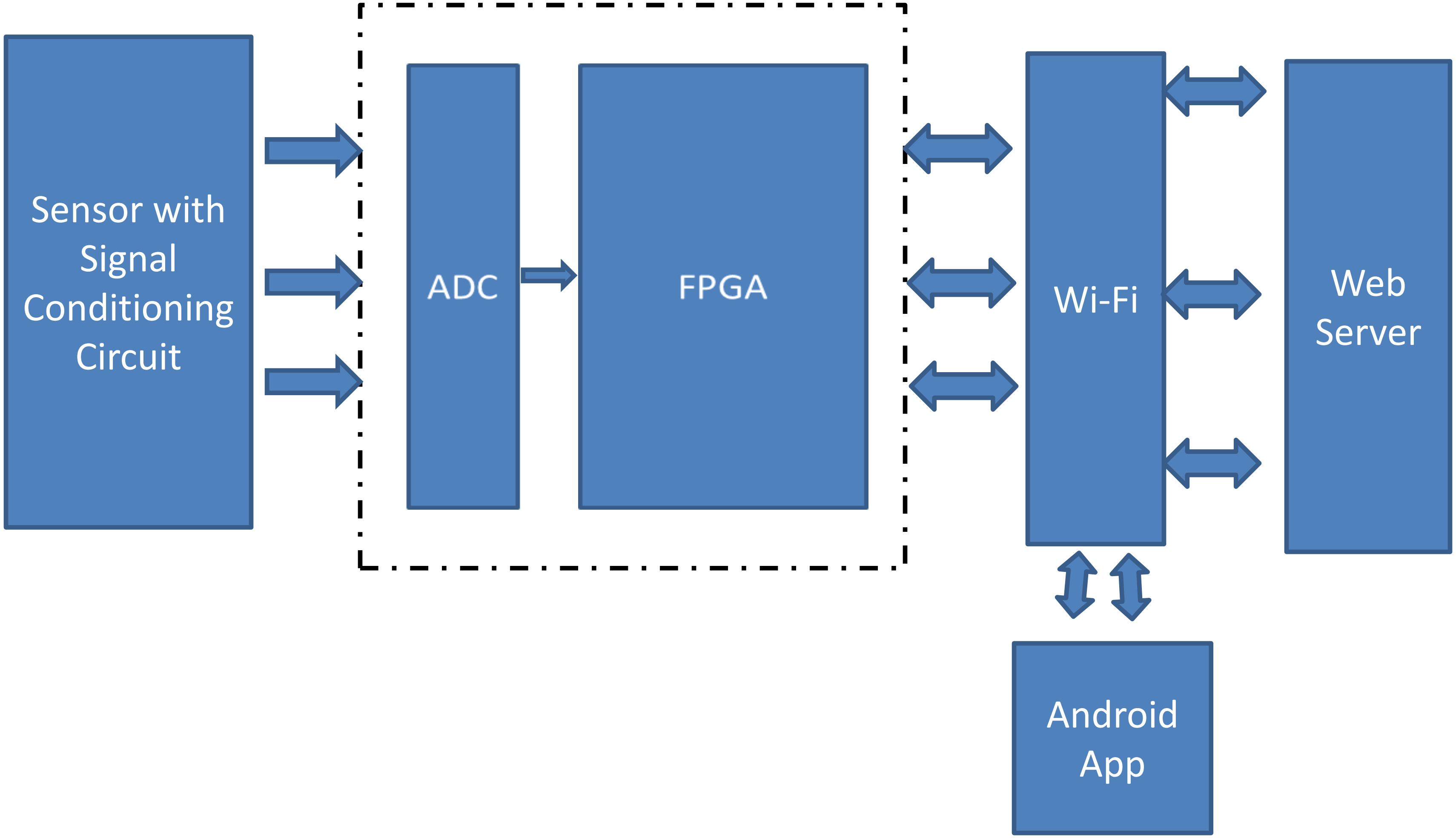


Figure 2. Block Diagram of proposed system

Dust Sensor

GP2Y1010AU0F is an optical air quality sensor, designed to sense dust particles

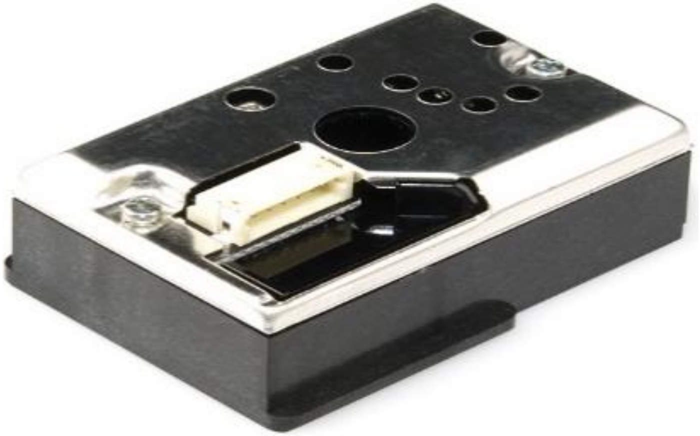


Figure 3. Dust Sensor

Specifications of dust sensor:

- Compact, thin package(46.0 × 30.0 × 17.6 mm)
- Low consumption current(20mA max, 11mA typical)
- sensitivity of 0.5V/0.1mg/m^3

Dust Sensor Working

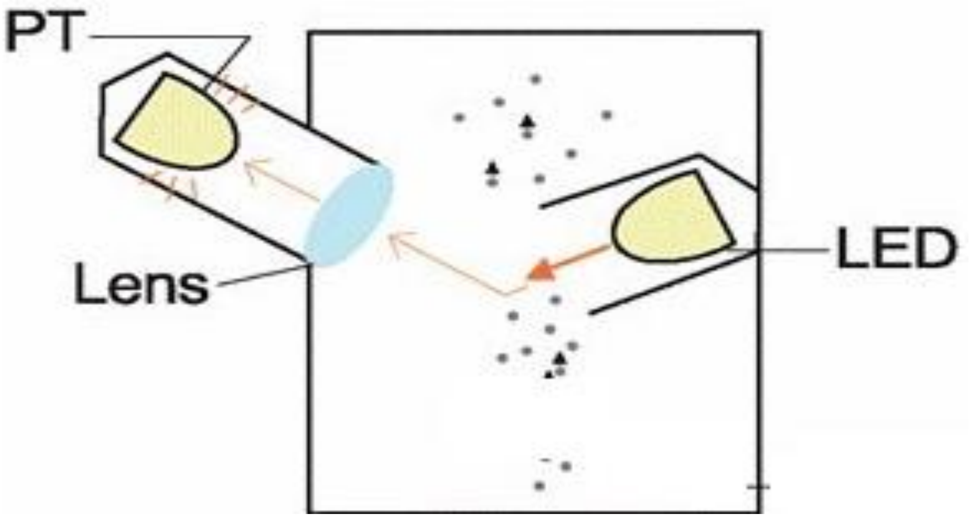


Figure 4. Dust Sensor Working Principle

- This sensor has two basic elements, one is infrared emitting diode and photo detecting transistor
- These elements are arranged in diagonally into this device, to allow it to detect the reflected light from dust particles

Dust Sensor Working

- The device makes voltage output even when dust is not being detected
- This output voltage at no dust condition can specified as Vnd, this no dust voltage is due to reflected light from case of the device and its some part is get detected by phototransistor

Behavioral Model of Dust Sensor

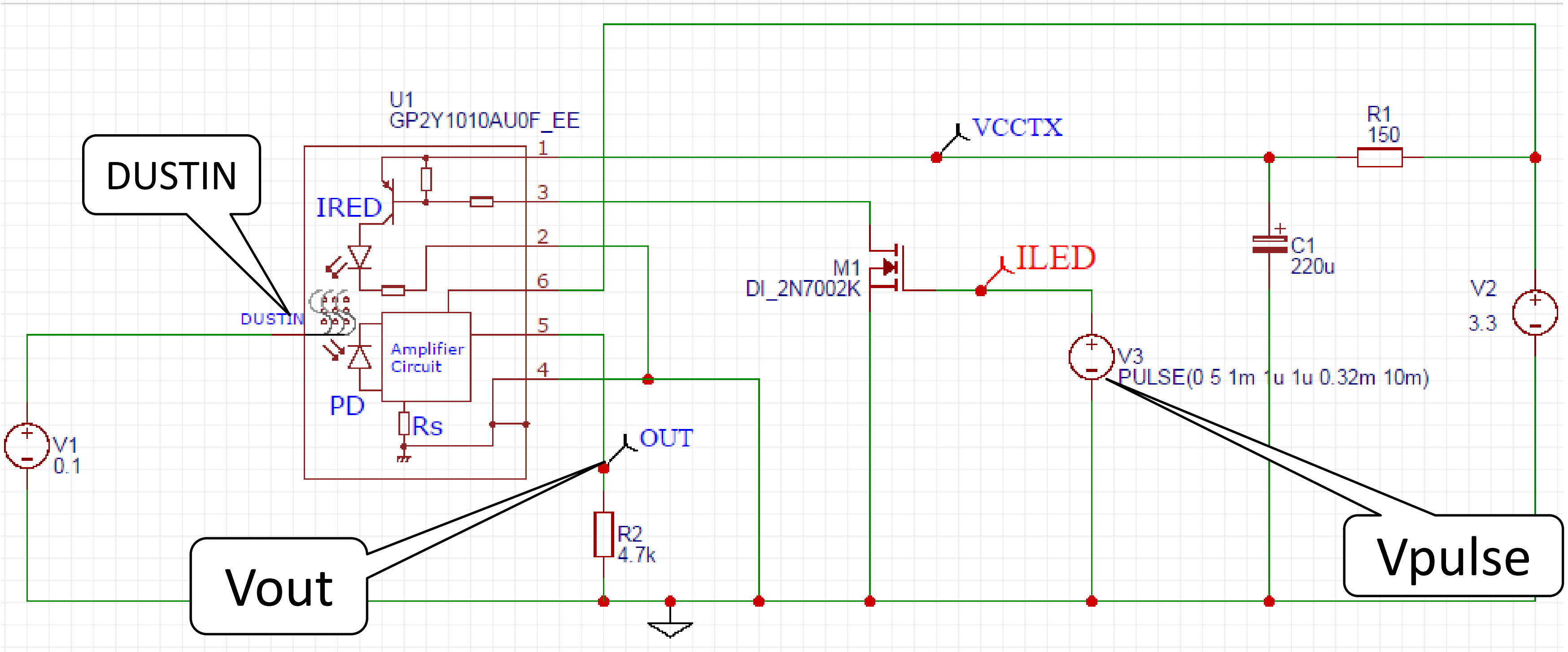
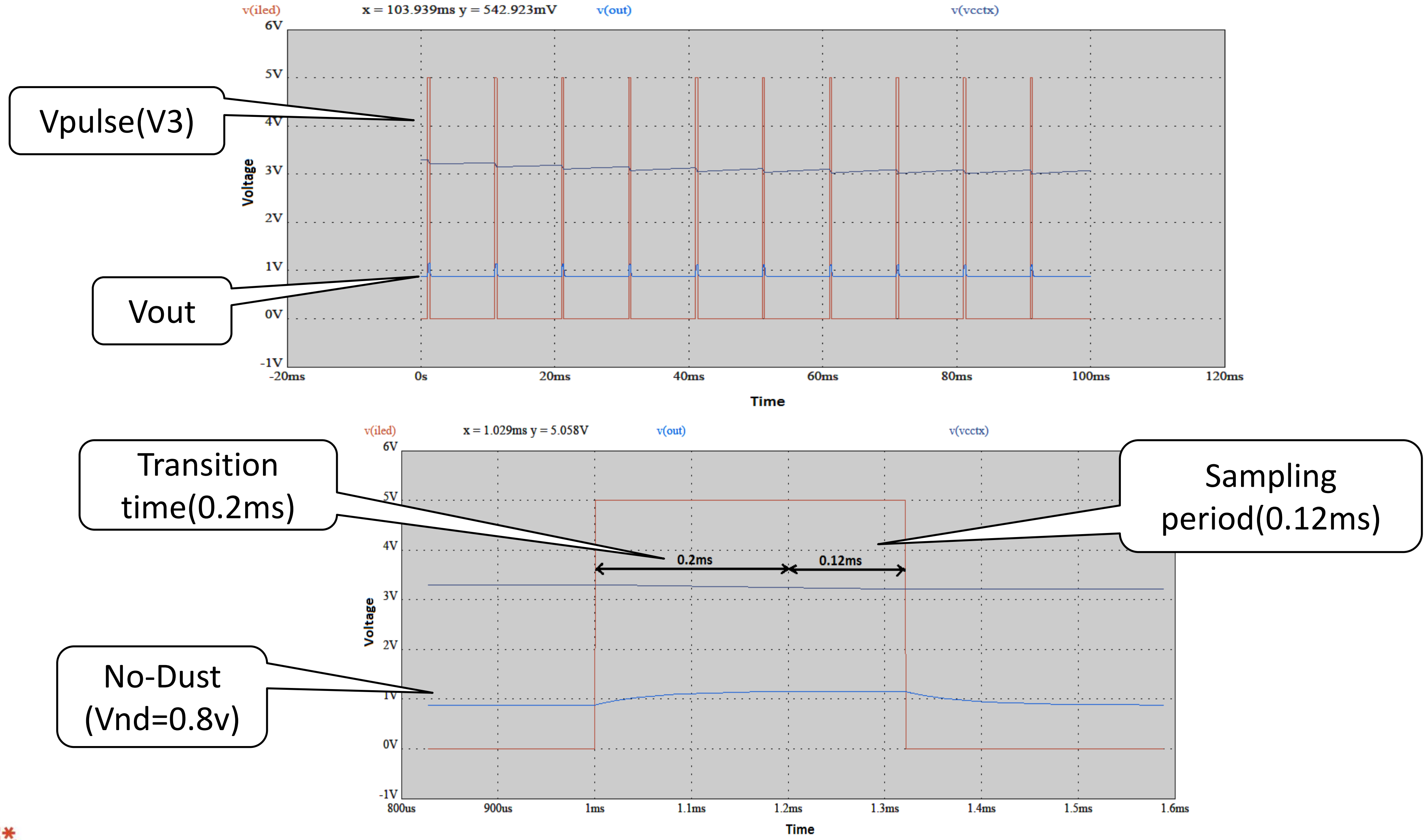


Figure 5. Schematic of Behavioral Circuit

- DUSTIN input scale is 0.1v represent 1mg/m^3
- The output voltage (Vout) of sensor is given by sum of output voltage when there is no dust (Vnd) and output corresponding to dust Vd.

  Vout = Vnd + Vd

Simulation of Dust Sensor Model



Vpulse(V3)

Vout

Transition time(0.2ms)

Sampling period(0.12ms)

No-Dust (Vnd=0.8v)

Figure 6. Simulation of Behavioral Circuit

Result of Simulation

- The Processor should give 0.2-0.3ms delay after that it should start to take samples of voltage from pin-5 of sensor(Vo)
- There is period of 0.12ms for sampling. When finished, set the pin-3(ILED) to low to disable the internal infrared emitting diode

Table 1. Dust Density Values

Practical Values		Simulation Values	
Voltage (V)	**Dust density(mg/m^3)**	**Voltage (V)**	**Dust density(mg/m^3)**
0.90	0.053	0.8	0.040
0.95	0.061	0.9	0.053
1.00	0.070	1.0	0.070
1.25	0.110	1.1	0.087
1.50	0.120	1.2	0.104

Air Pollution Index (API)

Table 2. Air Pollution Index

Air Pollution Index ($\mu g/m^3$)	Remark	Level of Health Concern
0 to 40	Good	Minimal impact
40 to 60	Moderate	Breathing discomfort to most people on prolonged exposure and causes lungs disease, asthma and heart diseases
Above 60	Severe	Affects healthy people and seriously impacts those with existing diseases and causes respiratory illness

Configuration of Pmod XADC

It is a 12 pin Pmod connector provides two 3.3V VCC signals (pins 6 and 12),two Ground signals (pins 5 and 11),eight logic signals. VCC and Ground pins provides a current of 1A.

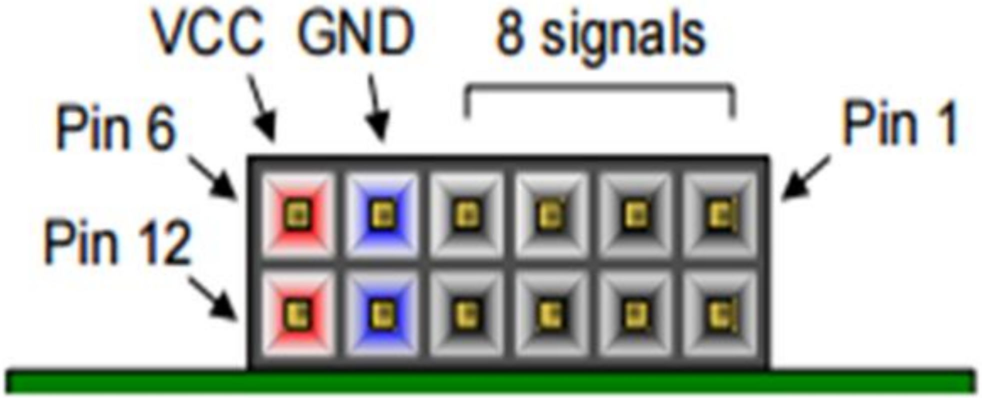
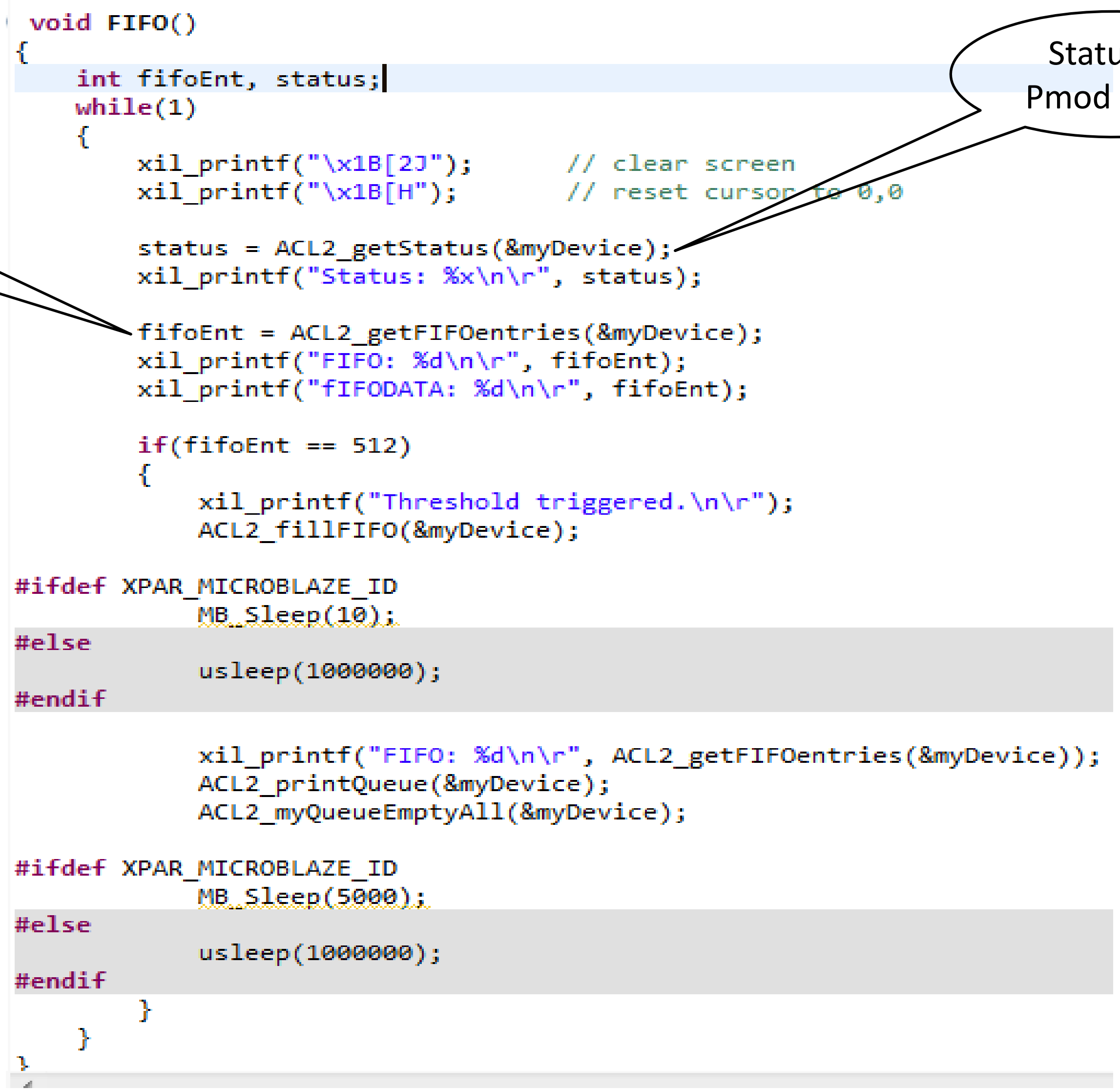


Figure 7. Pmod JXADC Pins

Continued……

- The Pmod ACL IP is a dual channel 12-bit analog-to-digital converter capable of operating at 1 Msps
- Either channel can be driven by any of the auxiliary analog input pairs connected to the JXADC header
- The XADC core is controlled and accessed from a user design via the Dynamic Reconfiguration Port (DRP)
- For communication over Wi-Fi the digital data coming from the Pmod ACL is stored in registers and migrated into SDK platform it should be read and transmit over Wi-Fi
- Its result appears on web page and android app

Sensor Data Configuration to Pmod XADC



```
void FIFO()
{
    int fifoEnt, status;
    while(1)
    {
        xil_printf("\x1B[2J");      // clear screen
        xil_printf("\x1B[H");       // reset cursor to 0,0

        status = ACL2_getStatus(&myDevice);
        xil_printf("Status: %x\n\r", status);

        fifoEnt = ACL2_getFIFOentries(&myDevice);
        xil_printf("FIFO: %d\n\r", fifoEnt);
        xil_printf("fIFODATA: %d\n\r", fifoEnt);

        if(fifoEnt == 512)
        {
            xil_printf("Threshold triggered.\n\r");
            ACL2_fillFIFO(&myDevice);

#ifdef XPAR_MICROBLAZE_ID
            MB_Sleep(10);
#else
            usleep(1000000);
#endif

            xil_printf("FIFO: %d\n\r", ACL2_getFIFOentries(&myDevice));
            ACL2_printQueue(&myDevice);
            ACL2_myQueueEmptyAll(&myDevice);

#ifdef XPAR_MICROBLAZE_ID
            MB_Sleep(5000);
#else
            usleep(1000000);
#endif
        }
    }
}
```

Status of Pmod XADC

Input data from sensor

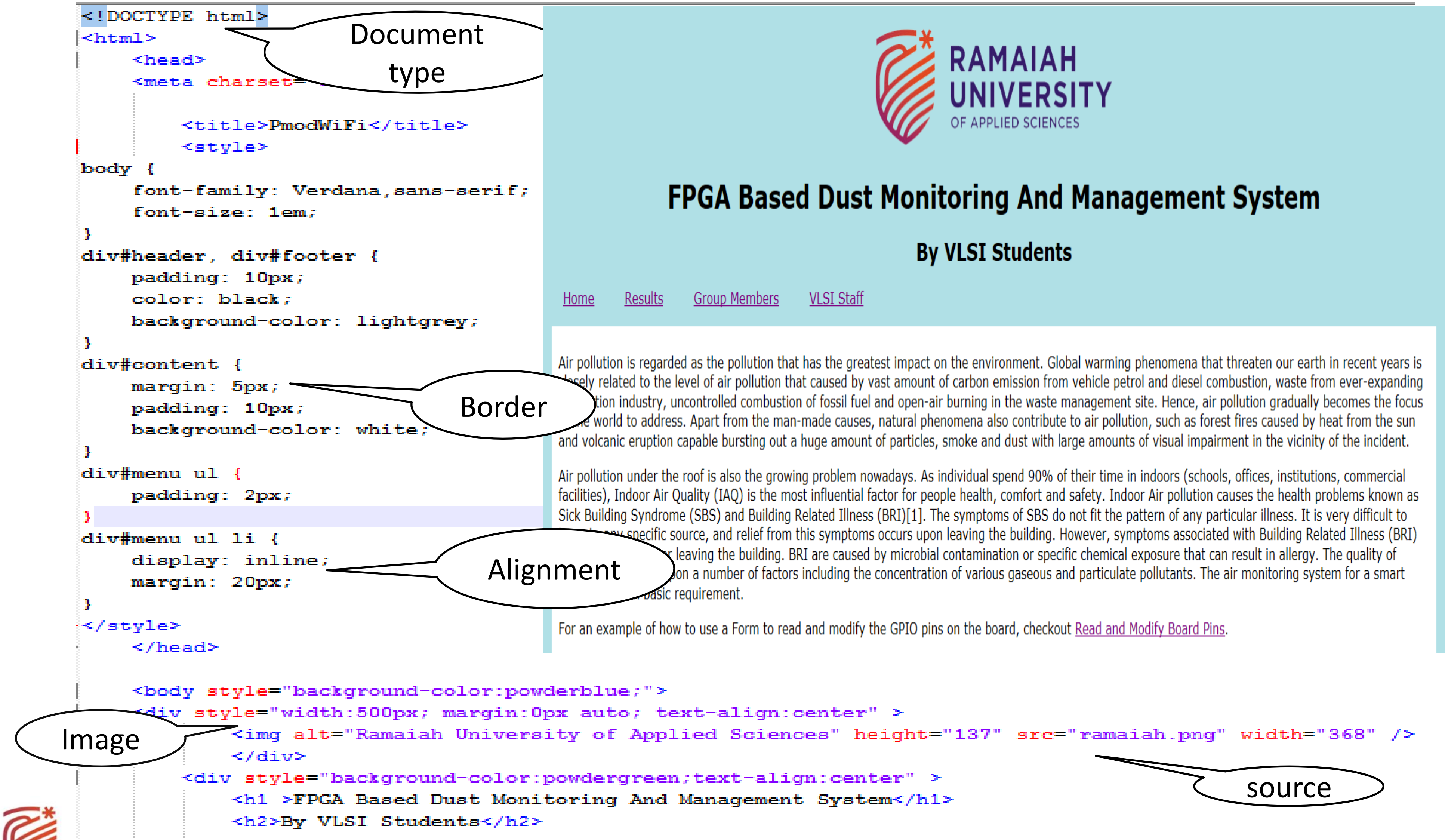
Web Page Development

- Web pages read HTML in order to display your website content. Saying "HTML page" is the same as saying "Web page“.
- In order to create a web page so it is properly read by the web browser, you will need 5 different HTML tags.

  1) <!DOCTYPE html /> 2) <html></html> 3) <title></title>

  4) <head></head> 5) <body></body>
- In this project HTML Page is created with images, contents, links between multiple HTML Page and Graphs

HTML Home Page



```
<!DOCTYPE html>
<html>
     <head>
     <meta charset=
          
          <title>PmodWiFi</title>
          <style>
body {
     font-family: Verdana,sans-serif;
     font-size: 1em;
}
div#header, div#footer {
     padding: 10px;
     color: black;
     background-color: lightgrey;
}
div#content {
     margin: 5px;
     padding: 10px;
     background-color: white;
}
div#menu ul {
     padding: 2px;
}
div#menu ul li {
     display: inline;
     margin: 20px;
}
</style>
     </head>

     <body style="background-color:powderblue;">
     <div style="width:500px; margin:0px auto; text-align:center" >
          <img alt="Ramaiah University of Applied Sciences" height="137" src="ramaiah.png" width="368" />
          </div>
     <div style="background-color:powdergreen;text-align:center" >
          <h1 >FPGA Based Dust Monitoring And Management System</h1>
          <h2>By VLSI Students</h2>
```

FPGA Based Dust Monitoring And Management System

By VLSI Students

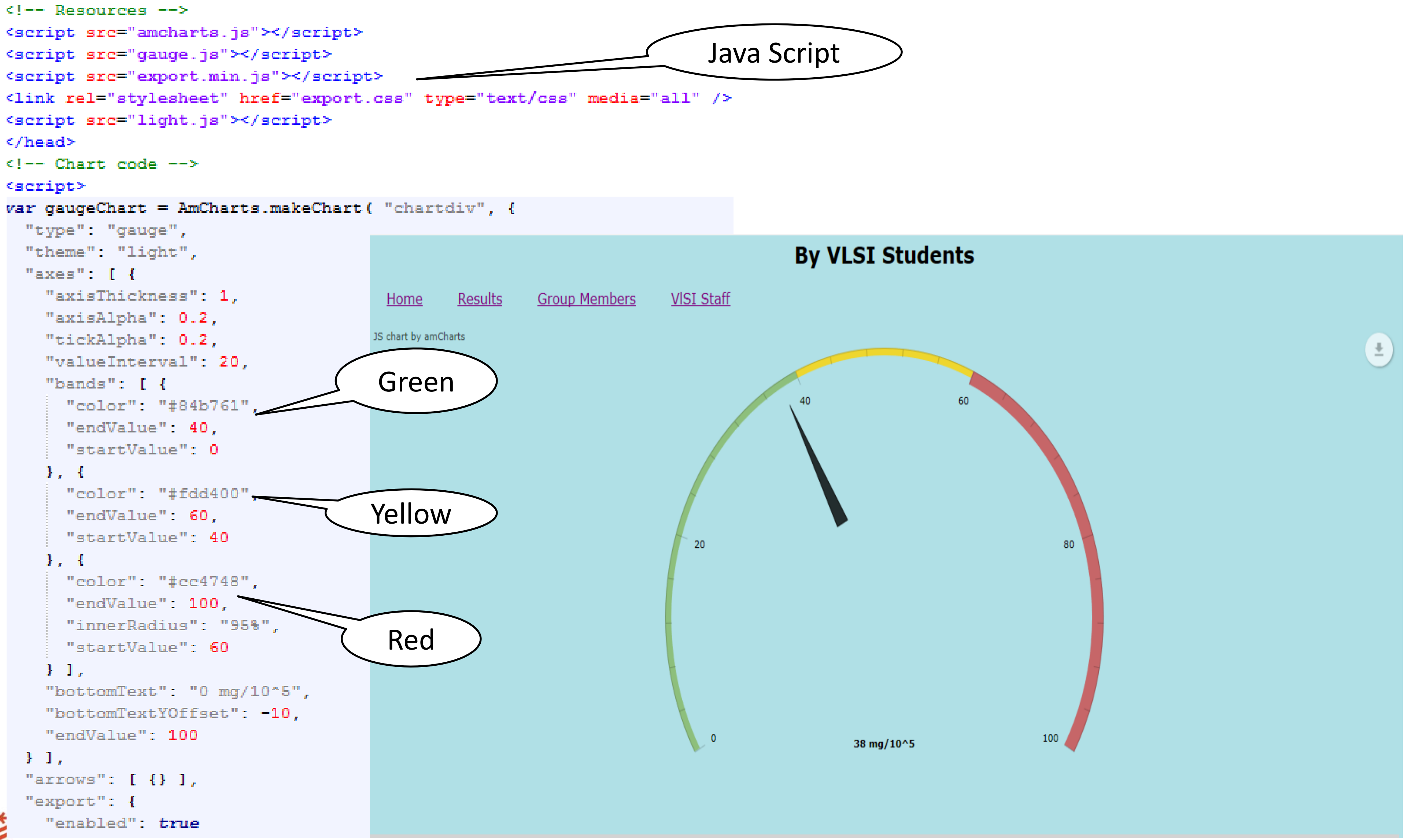
Home   Results   Group Members   VLSI Staff

Air pollution is regarded as the pollution that has the greatest impact on the environment. Global warming phenomena that threaten our earth in recent years is closely related to the level of air pollution that caused by vast amount of carbon emission from vehicle petrol and diesel combustion, waste from ever-expanding tion industry, uncontrolled combustion of fossil fuel and open-air burning in the waste management site. Hence, air pollution gradually becomes the focus e world to address. Apart from the man-made causes, natural phenomena also contribute to air pollution, such as forest fires caused by heat from the sun and volcanic eruption capable bursting out a huge amount of particles, smoke and dust with large amounts of visual impairment in the vicinity of the incident.

Air pollution under the roof is also the growing problem nowadays. As individual spend 90% of their time in indoors (schools, offices, institutions, commercial facilities), Indoor Air Quality (IAQ) is the most influential factor for people health, comfort and safety. Indoor Air pollution causes the health problems known as Sick Building Syndrome (SBS) and Building Related Illness (BRI)[1]. The symptoms of SBS do not fit the pattern of any particular illness. It is very difficult to any specific source, and relief from this symptoms occurs upon leaving the building. However, symptoms associated with Building Related Illness (BRI) r leaving the building. BRI are caused by microbial contamination or specific chemical exposure that can result in allergy. The quality of pon a number of factors including the concentration of various gaseous and particulate pollutants. The air monitoring system for a smart basic requirement.

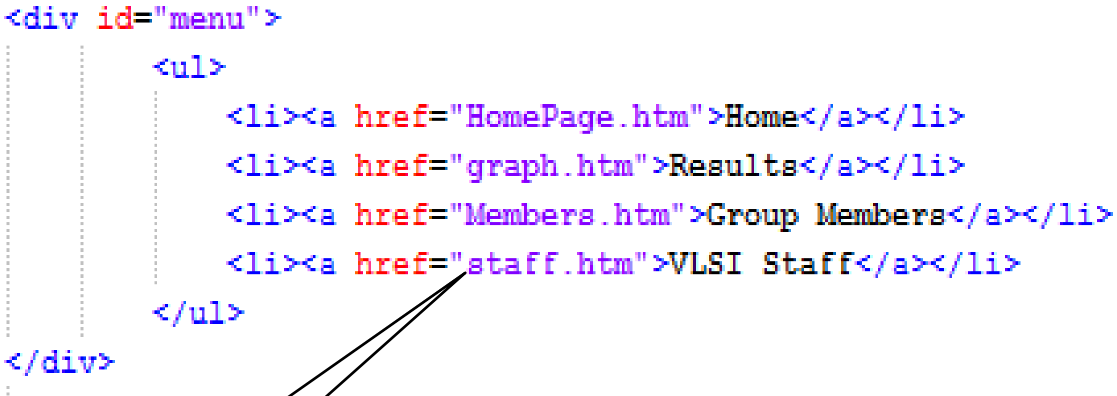
For an example of how to use a Form to read and modify the GPIO pins on the board, checkout Read and Modify Board Pins.

Dust Sensor Graph



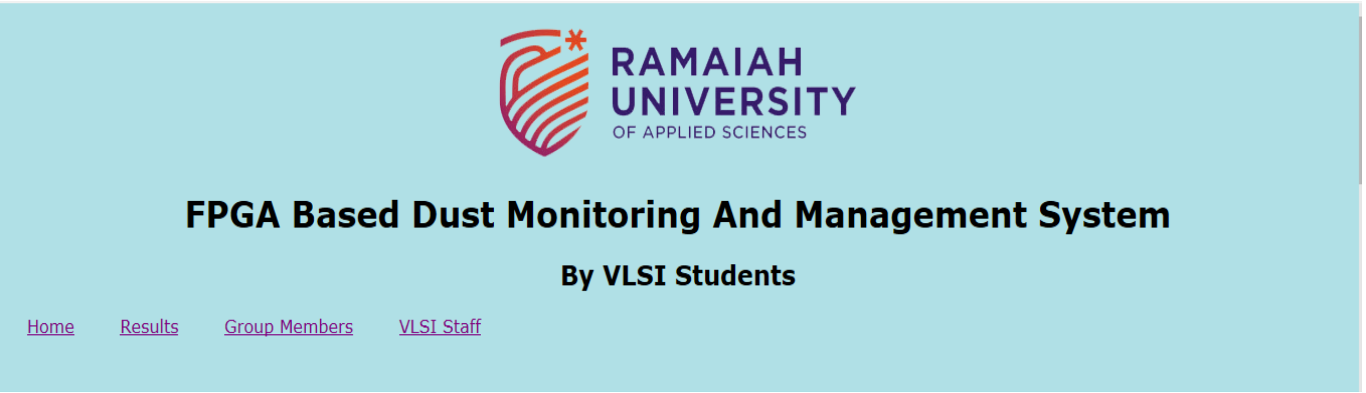
```
<!-- Resources -->
<script src="amcharts.js"></script>
<script src="gauge.js"></script>
<script src="export.min.js"></script>
<link rel="stylesheet" href="export.css" type="text/css" media="all" />
<script src="light.js"></script>
</head>
<!-- Chart code -->
<script>
var gaugeChart = AmCharts.makeChart( "chartdiv", {
  "type": "gauge",
  "theme": "light",
  "axes": [ {
    "axisThickness": 1,
    "axisAlpha": 0.2,
    "tickAlpha": 0.2,
    "valueInterval": 20,
    "bands": [ {
      "color": "#84b761",
      "endValue": 40,
      "startValue": 0
    }, {
      "color": "#fdd400",
      "endValue": 60,
      "startValue": 40
    }, {
      "color": "#cc4748",
      "endValue": 100,
      "innerRadius": "95%",
      "startValue": 60
    } ],
    "bottomText": "0 mg/10^5",
    "bottomTextYOffset": -10,
    "endValue": 100
  } ],
  "arrows": [ {} ],
  "export": {
    "enabled": true
```

Link of Web Pages



```
<div id="menu">
        <ul>
            <li><a href="HomePage.htm">Home</a></li>
            <li><a href="graph.htm">Results</a></li>
            <li><a href="Members.htm">Group Members</a></li>
            <li><a href="staff.htm">VLSI Staff</a></li>
        </ul>
</div>
```

Links



FPGA Based Dust Monitoring And Management System

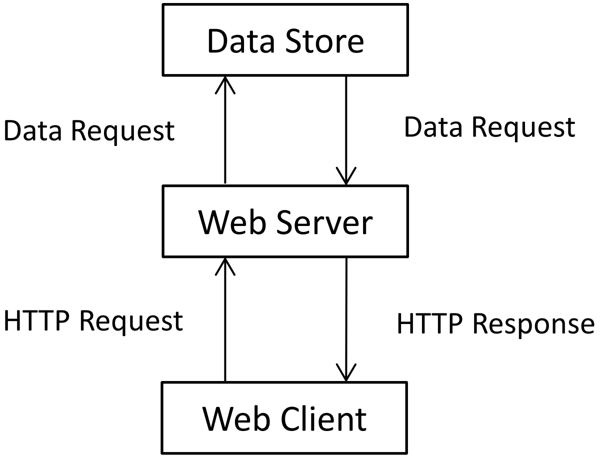
By VLSI Students

HTTP Server Configuration with FPGA

- HTTP (Hyper Text Transfer Protocol) is a protocol, that many web servers use to facilitate the transfer of html or webpages.
- If (request is found) else

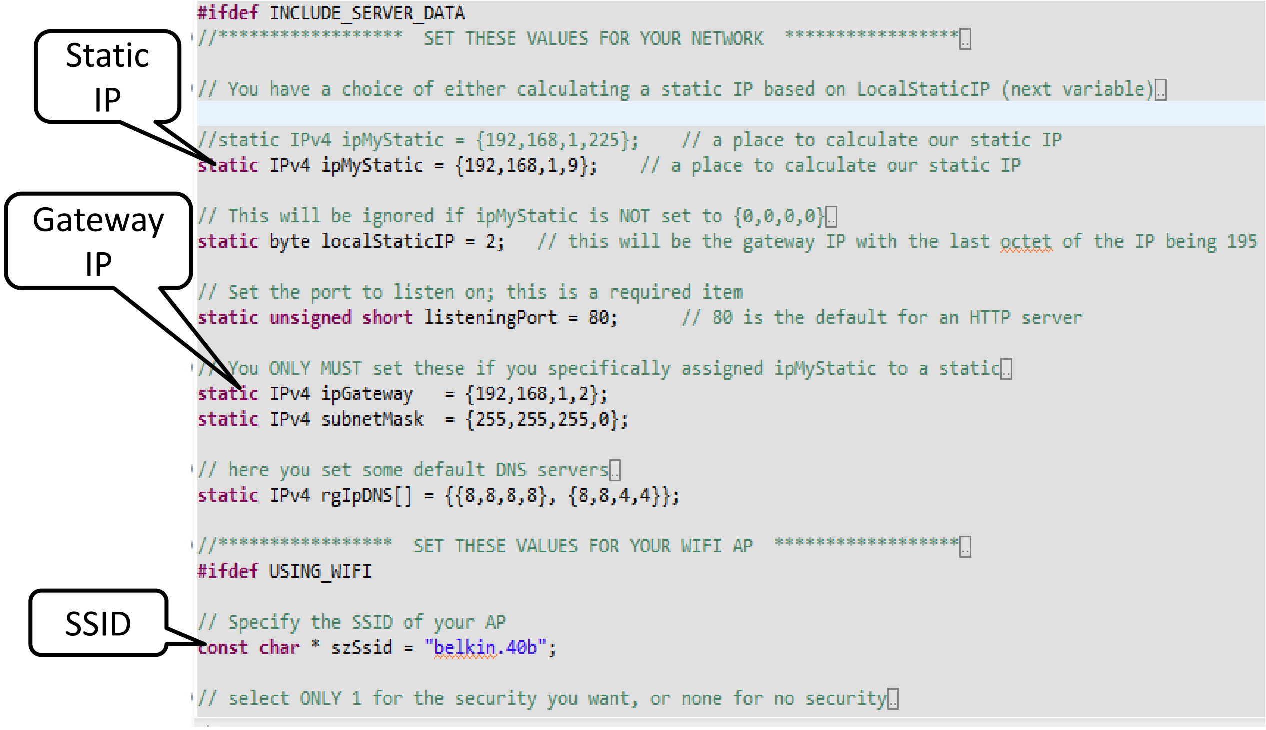
{

HTTP Response = Error page not found



}

HTTP Server Configuration with FPGA



```
#ifdef INCLUDE_SERVER_DATA
//******************  SET THESE VALUES FOR YOUR NETWORK  *****************

// You have a choice of either calculating a static IP based on LocalStaticIP (next variable)

//static IPv4 ipMyStatic = {192,168,1,225};    // a place to calculate our static IP
static IPv4 ipMyStatic = {192,168,1,9};    // a place to calculate our static IP

// This will be ignored if ipMyStatic is NOT set to {0,0,0,0}
static byte localStaticIP = 2;   // this will be the gateway IP with the last octet of the IP being 195

// Set the port to listen on; this is a required item
static unsigned short listeningPort = 80;      // 80 is the default for an HTTP server

// You ONLY MUST set these if you specifically assigned ipMyStatic to a static
static IPv4 ipGateway   = {192,168,1,2};
static IPv4 subnetMask  = {255,255,255,0};

// here you set some default DNS servers
static IPv4 rgIpDNS[] = {{8,8,8,8}, {8,8,4,4}};

//*****************  SET THESE VALUES FOR YOUR WIFI AP  ******************
#ifdef USING_WIFI

// Specify the SSID of your AP
const char * szSsid = "belkin.40b";

// select ONLY 1 for the security you want, or none for no security
```

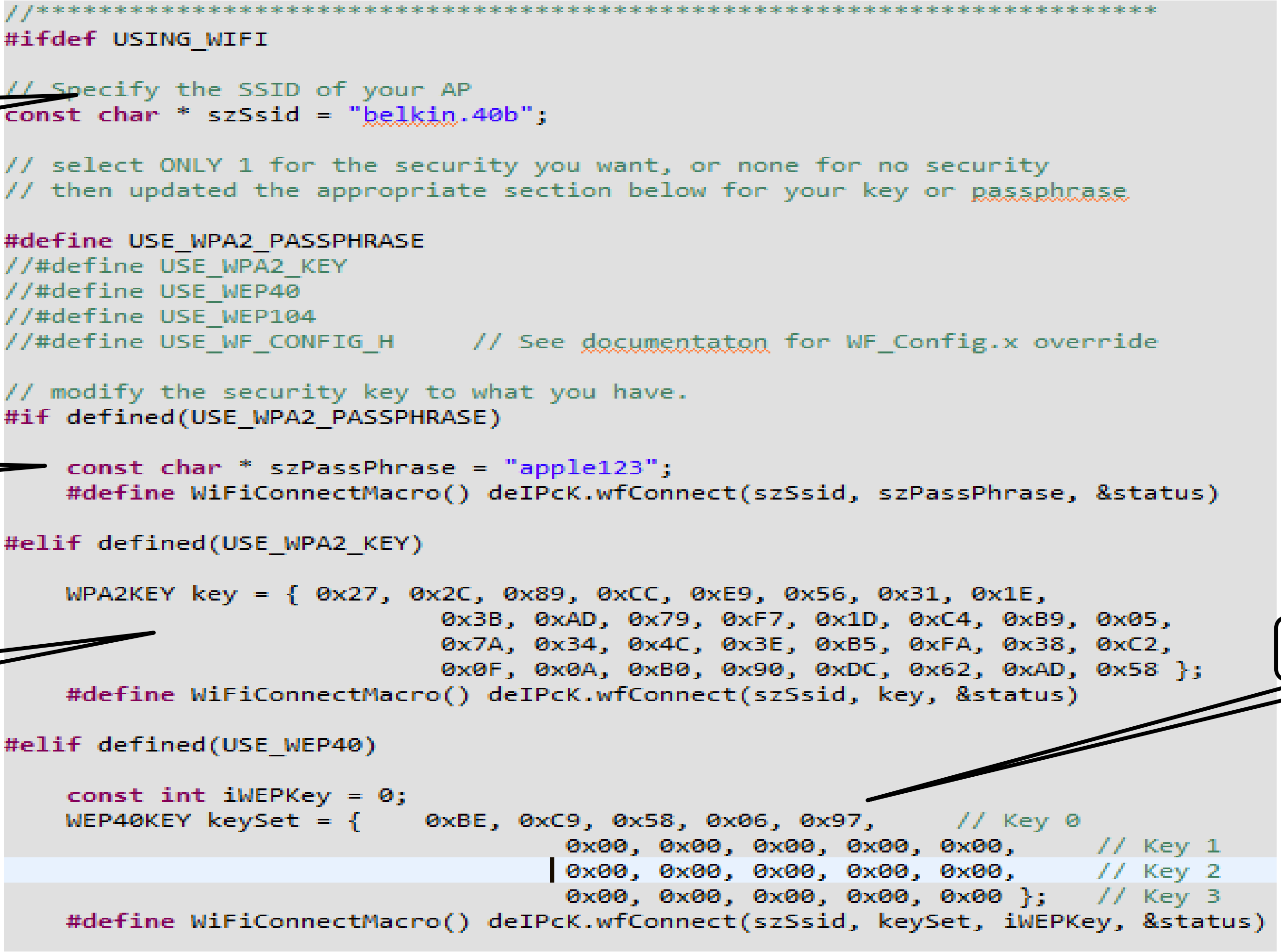
Static IP

Gateway IP

SSID

- Configure with IP
- Set local static IP
- Set Gateway IP
- Set Subnet mask
- Provide ssid, password

Continued……



```
//************************************************************************
#ifdef USING_WIFI

// Specify the SSID of your AP
const char * szSsid = "belkin.40b";

// select ONLY 1 for the security you want, or none for no security
// then updated the appropriate section below for your key or passphrase

#define USE_WPA2_PASSPHRASE
//#define USE_WPA2_KEY
//#define USE_WEP40
//#define USE_WEP104
//#define USE_WF_CONFIG_H     // See documentaton for WF_Config.x override

// modify the security key to what you have.
#if defined(USE_WPA2_PASSPHRASE)

    const char * szPassPhrase = "apple123";
    #define WiFiConnectMacro() deIPcK.wfConnect(szSsid, szPassPhrase, &status)

#elif defined(USE_WPA2_KEY)

    WPA2KEY key = { 0x27, 0x2C, 0x89, 0xCC, 0xE9, 0x56, 0x31, 0x1E,
                            0x3B, 0xAD, 0x79, 0xF7, 0x1D, 0xC4, 0xB9, 0x05,
                            0x7A, 0x34, 0x4C, 0x3E, 0xB5, 0xFA, 0x38, 0xC2,
                            0x0F, 0x0A, 0xB0, 0x90, 0xDC, 0x62, 0xAD, 0x58 };
    #define WiFiConnectMacro() deIPcK.wfConnect(szSsid, key, &status)

#elif defined(USE_WEP40)

    const int iWEPKey = 0;
    WEP40KEY keySet = {    0xBE, 0xC9, 0x58, 0x06, 0x97,     // Key 0
                                   0x00, 0x00, 0x00, 0x00, 0x00,     // Key 1
                                   0x00, 0x00, 0x00, 0x00, 0x00,     // Key 2
                                   0x00, 0x00, 0x00, 0x00, 0x00 };   // Key 3
    #define WiFiConnectMacro() deIPcK.wfConnect(szSsid, keySet, iWEPKey, &status)
```

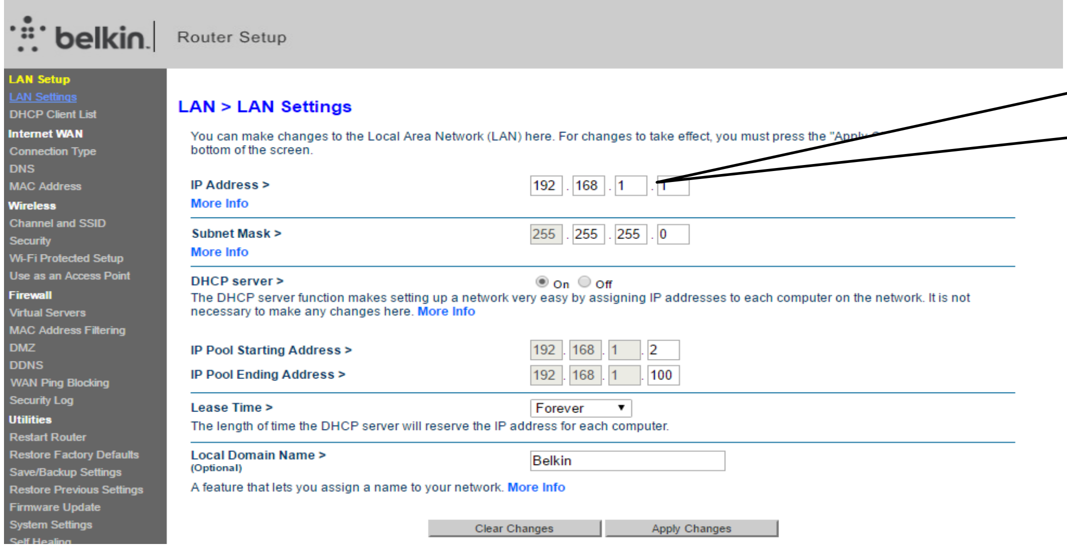
SSID of Router

Password

WPA2 Key

WEP Key

Router Configuration



LAN > LAN Settings

You can make changes to the Local Area Network (LAN) here. For changes to take effect, you must press the "Apply
bottom of the screen.

IP Address > 192 . 168 . 1 . 1
More Info

Subnet Mask > 255 . 255 . 255 . 0
More Info

DHCP server > ● On ○ Off
The DHCP server function makes setting up a network very easy by assigning IP addresses to each computer on the network. It is not necessary to make any changes here. More Info

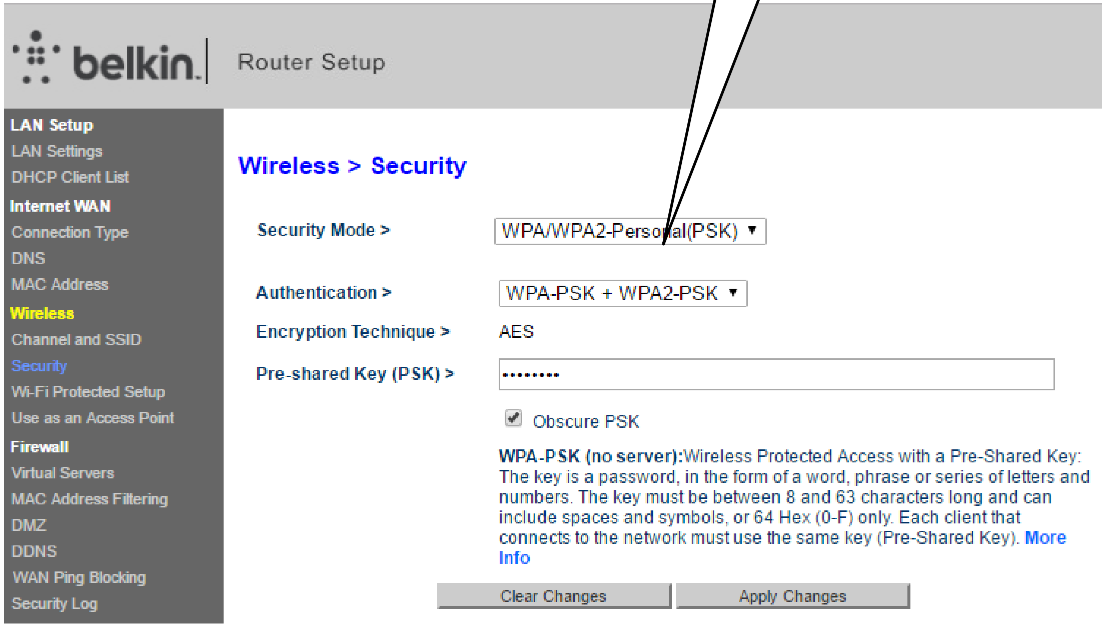
IP Pool Starting Address > 192 . 168 . 1 . 2
IP Pool Ending Address > 192 . 168 . 1 . 100

Lease Time > Forever ▼
The length of time the DHCP server will reserve the IP address for each computer.

Local Domain Name > Belkin
(Optional)
A feature that lets you assign a name to your network. More Info

Router IP Gateway For Pmod Wi-Fi
192.168.1.2

Security Mode
WPA/WPA2



Wireless > Security

Security Mode > WPA/WPA2-Personal(PSK) ▼

Authentication > WPA-PSK + WPA2-PSK ▼

Encryption Technique > AES

Pre-shared Key (PSK) > ••••••••

☑ Obscure PSK

WPA-PSK (no server): Wireless Protected Access with a Pre-Shared Key: The key is a password, in the form of a word, phrase or series of letters and numbers. The key must be between 8 and 63 characters long and can include spaces and symbols, or 64 Hex (0-F) only. Each client that connects to the network must use the same key (Pre-Shared Key). More Info

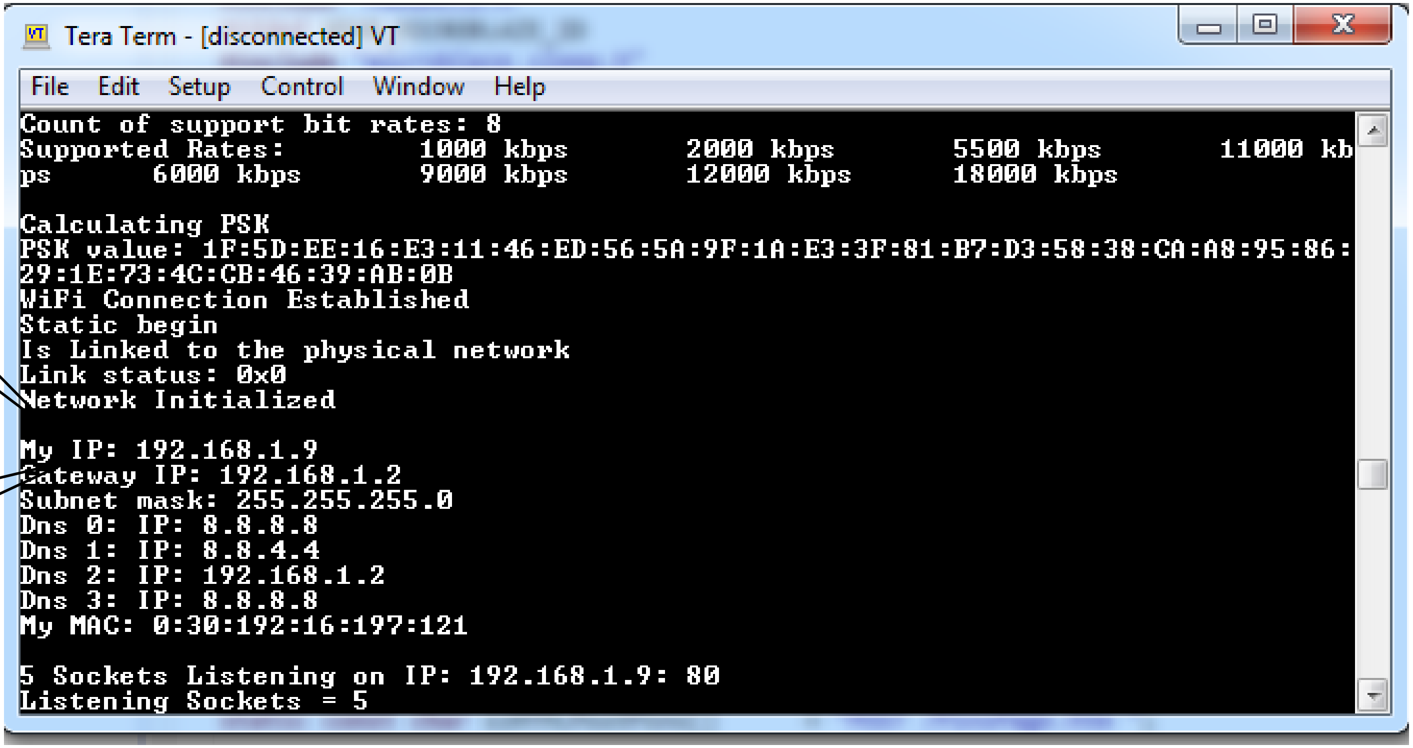
Continued……

- The router we are using will provide the network. The Wi-Fi module will utilise the network and facilitate the data transmission from FPGA to end user through wifi provided by the router. We need to configure the router also with IP address, Gateway IP
- The router IP address will be the Pmod wifi gateway and the router, wifi module should be in the same network

Status of Pmod Wifi Connection with Network

Static IP 192.168.1.9

Gateway IP 192.168.1.2


Tera Term - [disconnected] VT
File Edit Setup Control Window Help

```
Count of support bit rates: 8
Supported Rates:        1000 kbps       2000 kbps       5500 kbps       11000 kb
ps      6000 kbps       9000 kbps       12000 kbps      18000 kbps

Calculating PSK
PSK value: 1F:5D:EE:16:E3:11:46:ED:56:5A:9F:1A:E3:3F:81:B7:D3:58:38:CA:A8:95:86:
29:1E:73:4C:CB:46:39:AB:0B
WiFi Connection Established
Static begin
Is Linked to the physical network
Link status: 0x0
Network Initialized

My IP: 192.168.1.9
Gateway IP: 192.168.1.2
Subnet mask: 255.255.255.0
Dns 0: IP: 8.8.8.8
Dns 1: IP: 8.8.4.4
Dns 2: IP: 192.168.1.2
Dns 3: IP: 8.8.8.8
My MAC: 0:30:192:16:197:121

5 Sockets Listening on IP: 192.168.1.9: 80
Listening Sockets = 5
```

Figure 8. Tera Term Serial Terminal Results

Pmod Wifi Configuration

- The Pmod Wi-Fi provides Wi-Fi access through the Microchip MRF24WG0MA Wi-Fi radio transceiver module. Users can communicate with the IEEE 802.11g compliant chip through SPI and achieve data rates up to 54 Mbps
- For configuring with MicroBlaze Design the DeIPck library should be configured in application layer

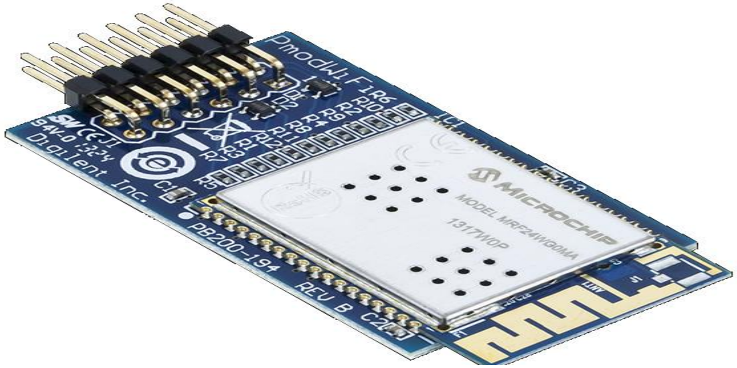
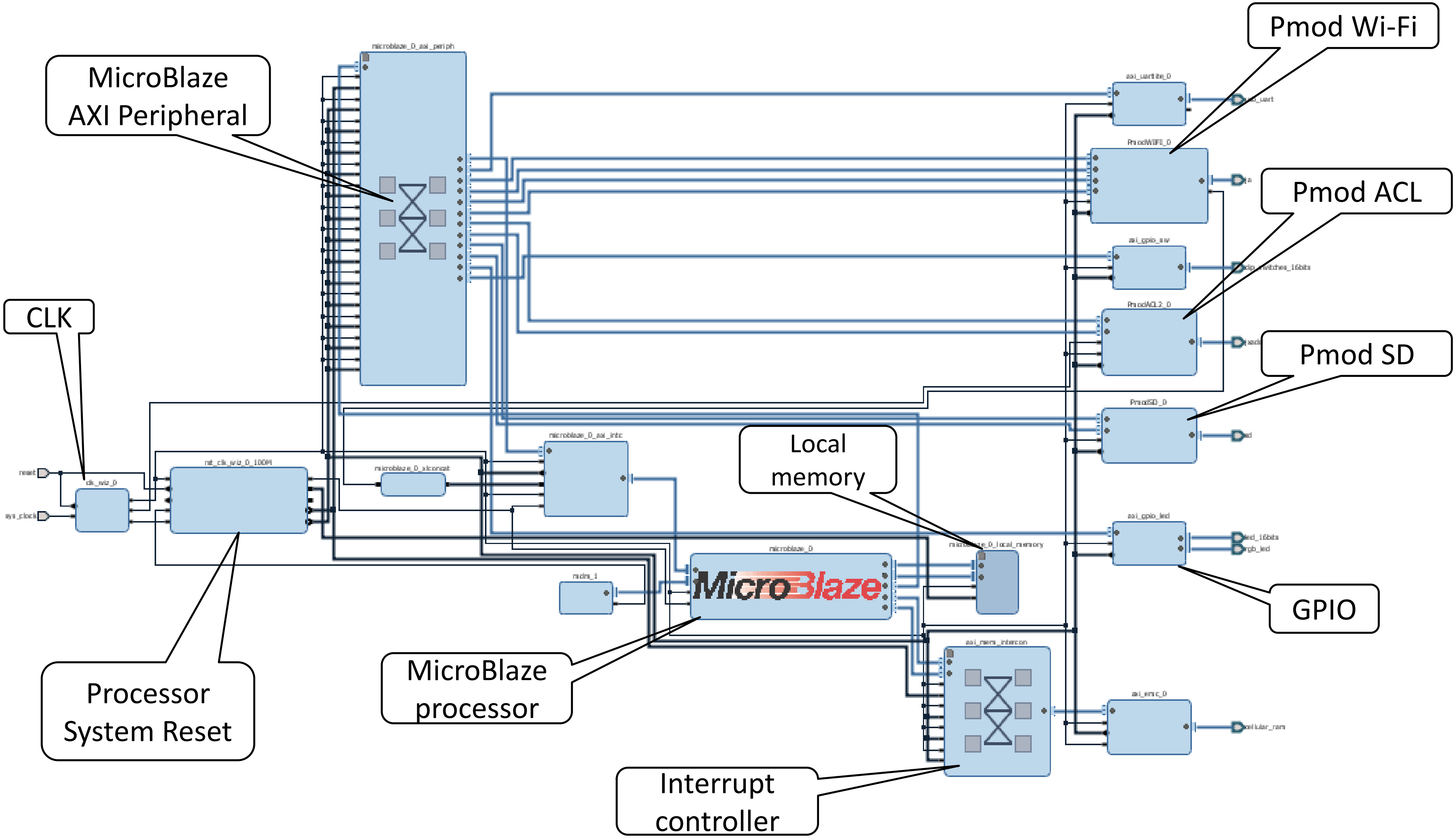


Figure 9.Pmod Wi-Fi Device

Architecture of Interface Control Unit



MicroBlaze AXI Peripheral

Pmod Wi-Fi

Pmod ACL

Pmod SD

CLK

Local memory

GPIO

Processor System Reset

MicroBlaze processor

Interrupt controller

Figure 10. Interface Control Unit

Continued……

- The schematic diagram represents the design and development of dust monitoring system.
- Which consist of,

    -MicroBlaze 32-bit embedded processor

    -Pmod Wifi

    -Pmod ACL

    -Clock

    -Block Ram

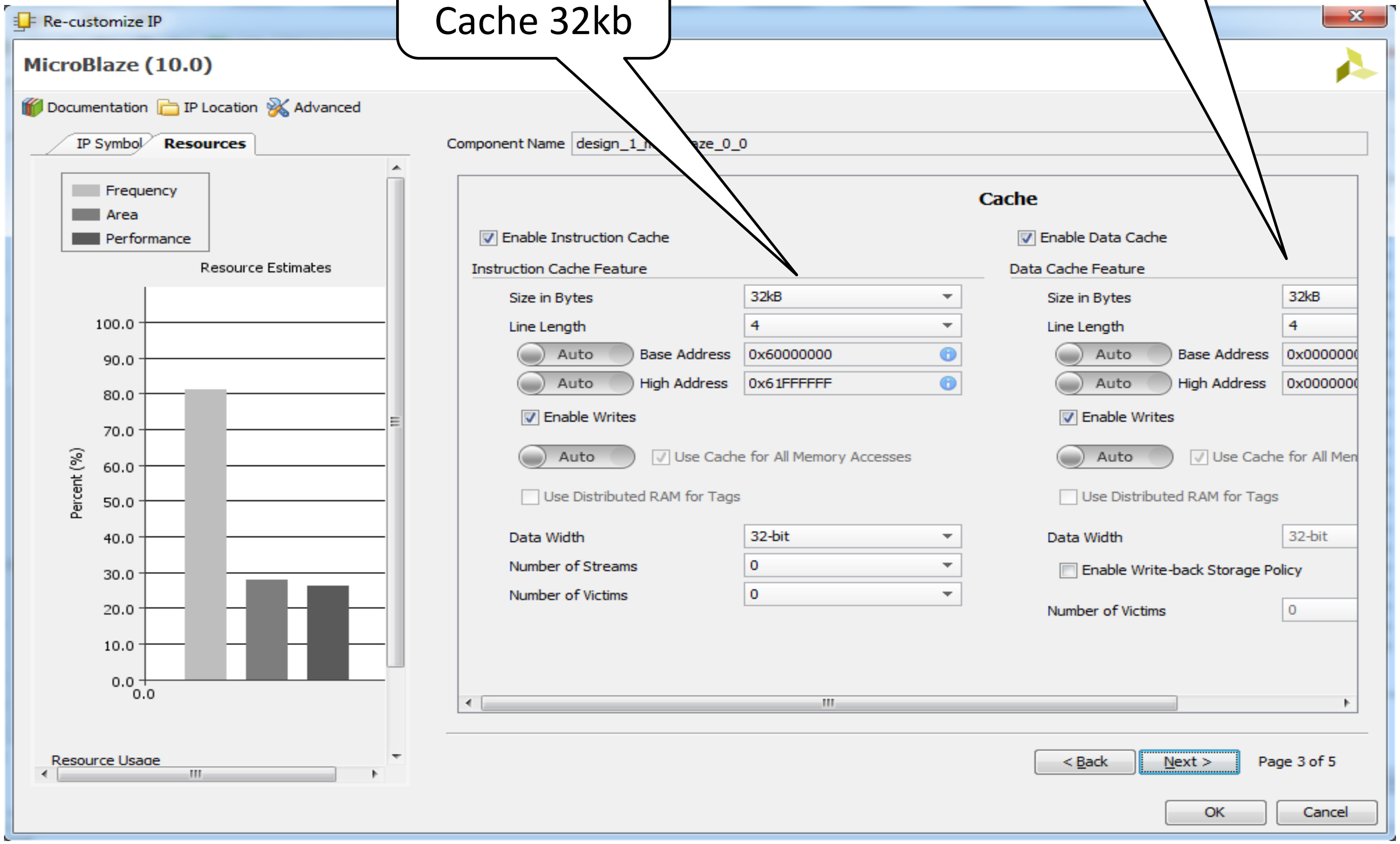
    -UART

    -GPIO

MicroBlaze Configuration

Instruction Cache 32kb

Data Cache 32kb



Component Name design_1_microblaze_0_0

Cache

☑ Enable Instruction Cache

Instruction Cache Feature

- Size in Bytes: 32kB
- Line Length: 4
- Auto Base Address: 0x60000000
- Auto High Address: 0x61FFFFFF
- ☑ Enable Writes
- Auto ☑ Use Cache for All Memory Accesses
- □ Use Distributed RAM for Tags
- Data Width: 32-bit
- Number of Streams: 0
- Number of Victims: 0

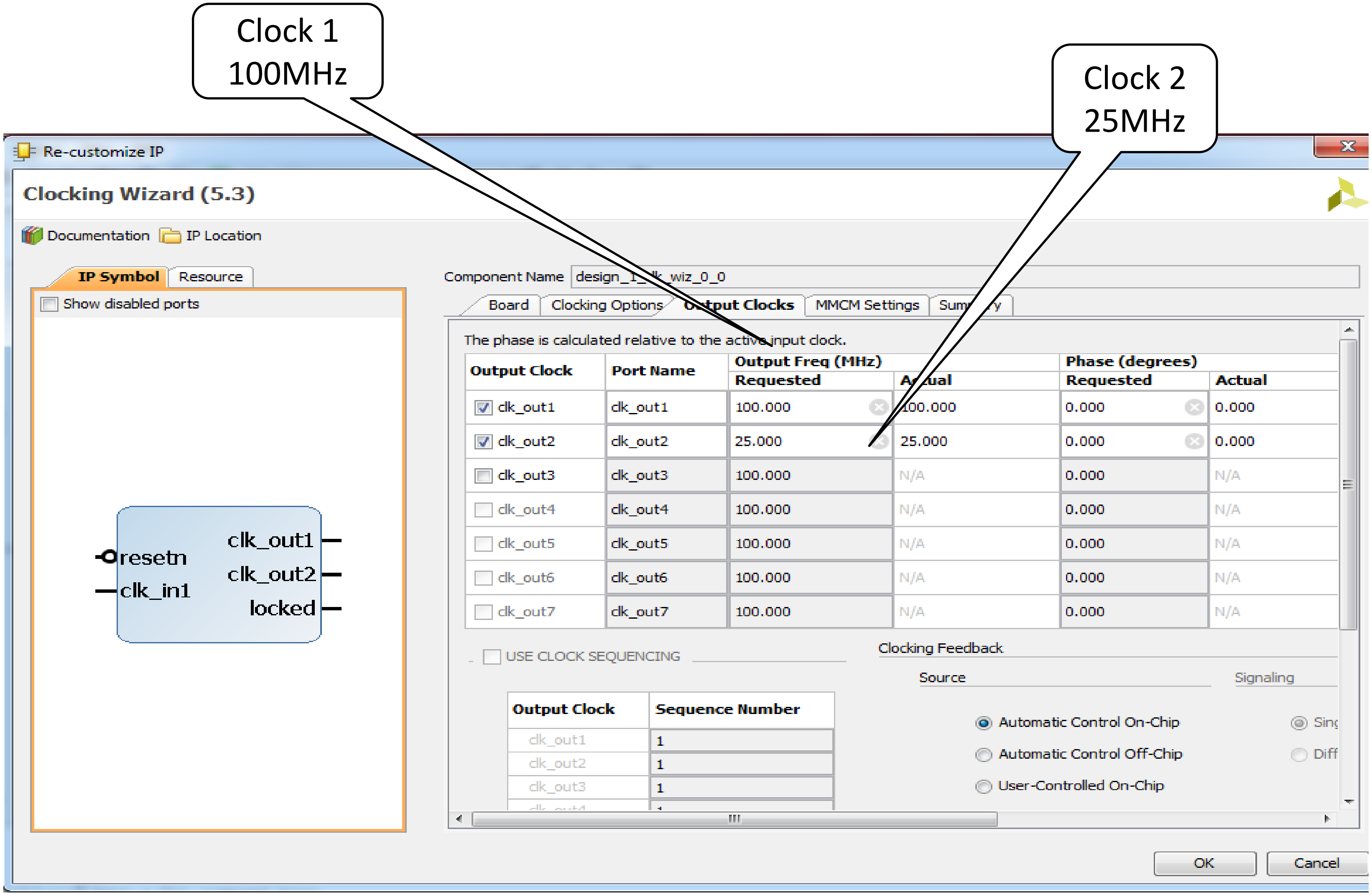
☑ Enable Data Cache

Data Cache Feature

- Size in Bytes: 32kB
- Line Length: 4
- Auto Base Address: 0x0000000
- Auto High Address: 0x0000000
- ☑ Enable Writes
- Auto ☑ Use Cache for All Men
- □ Use Distributed RAM for Tags
- Data Width: 32-bit
- □ Enable Write-back Storage Policy
- Number of Victims: 0

Page 3 of 5

Clock Configuration



Clock 1
100MHz

Clock 2
25MHz

Re-customize IP

Clocking Wizard (5.3)

Component Name design_1_clk_wiz_0_0

The phase is calculated relative to the active input clock.

Output Clock	Port Name	Output Freq (MHz)		Phase (degrees)	
		Requested	Actual	Requested	Actual
☑ clk_out1	clk_out1	100.000	100.000	0.000	0.000
☑ clk_out2	clk_out2	25.000	25.000	0.000	0.000
□ clk_out3	clk_out3	100.000	N/A	0.000	N/A
□ clk_out4	clk_out4	100.000	N/A	0.000	N/A
□ clk_out5	clk_out5	100.000	N/A	0.000	N/A
□ clk_out6	clk_out6	100.000	N/A	0.000	N/A
□ clk_out7	clk_out7	100.000	N/A	0.000	N/A

□ USE CLOCK SEQUENCING

Output Clock	Sequence Number
clk_out1	1
clk_out2	1
clk_out3	1
clk_out4	1

Clocking Feedback

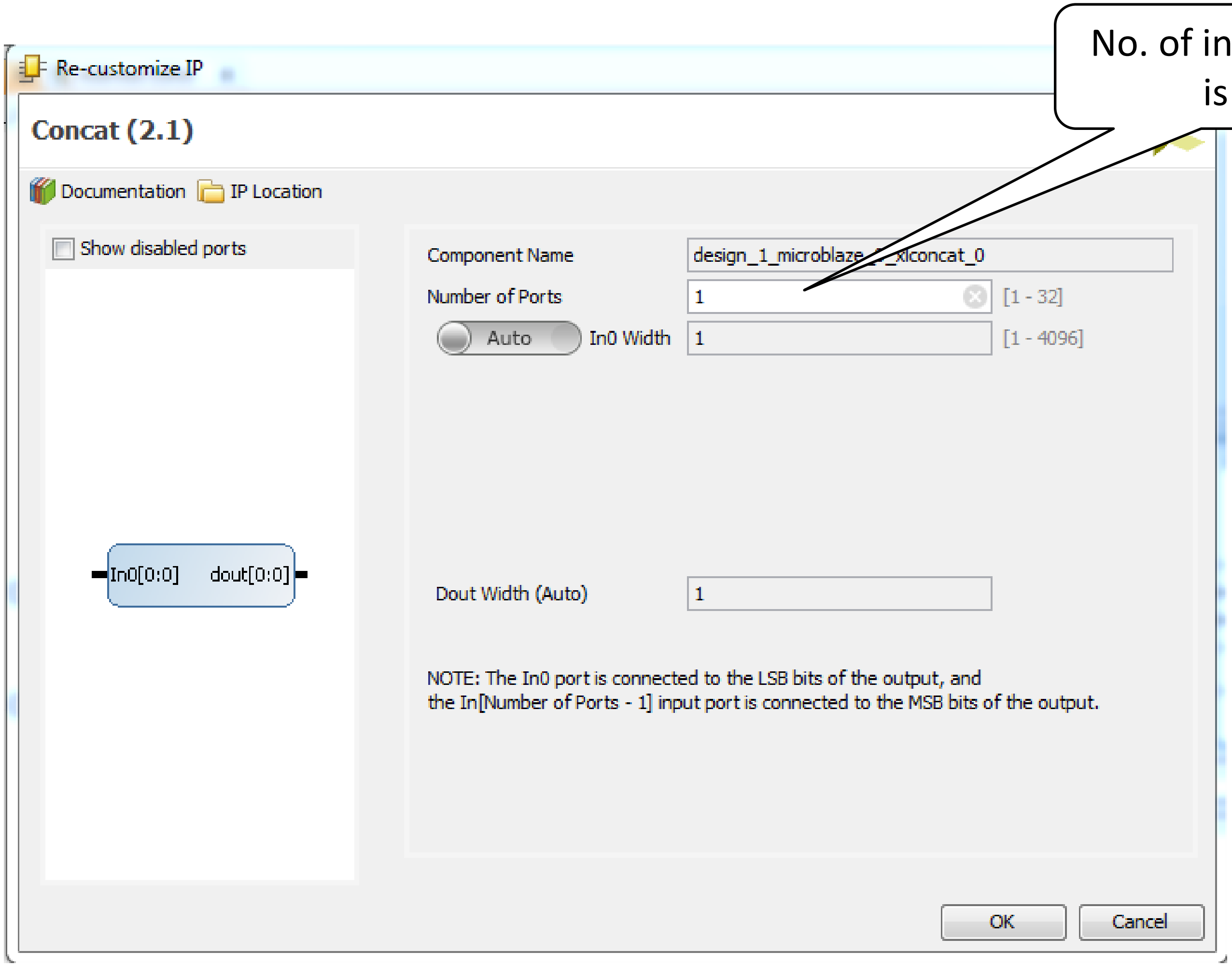
Source

● Automatic Control On-Chip
○ Automatic Control Off-Chip
○ User-Controlled On-Chip

Signaling

● Sing
○ Diff

Concat Block Configuration



Concat (2.1)

□ Show disabled ports

Component Name: design_1_microblaze_0_xlconcat_0

Number of Ports: 1 [1 - 32]

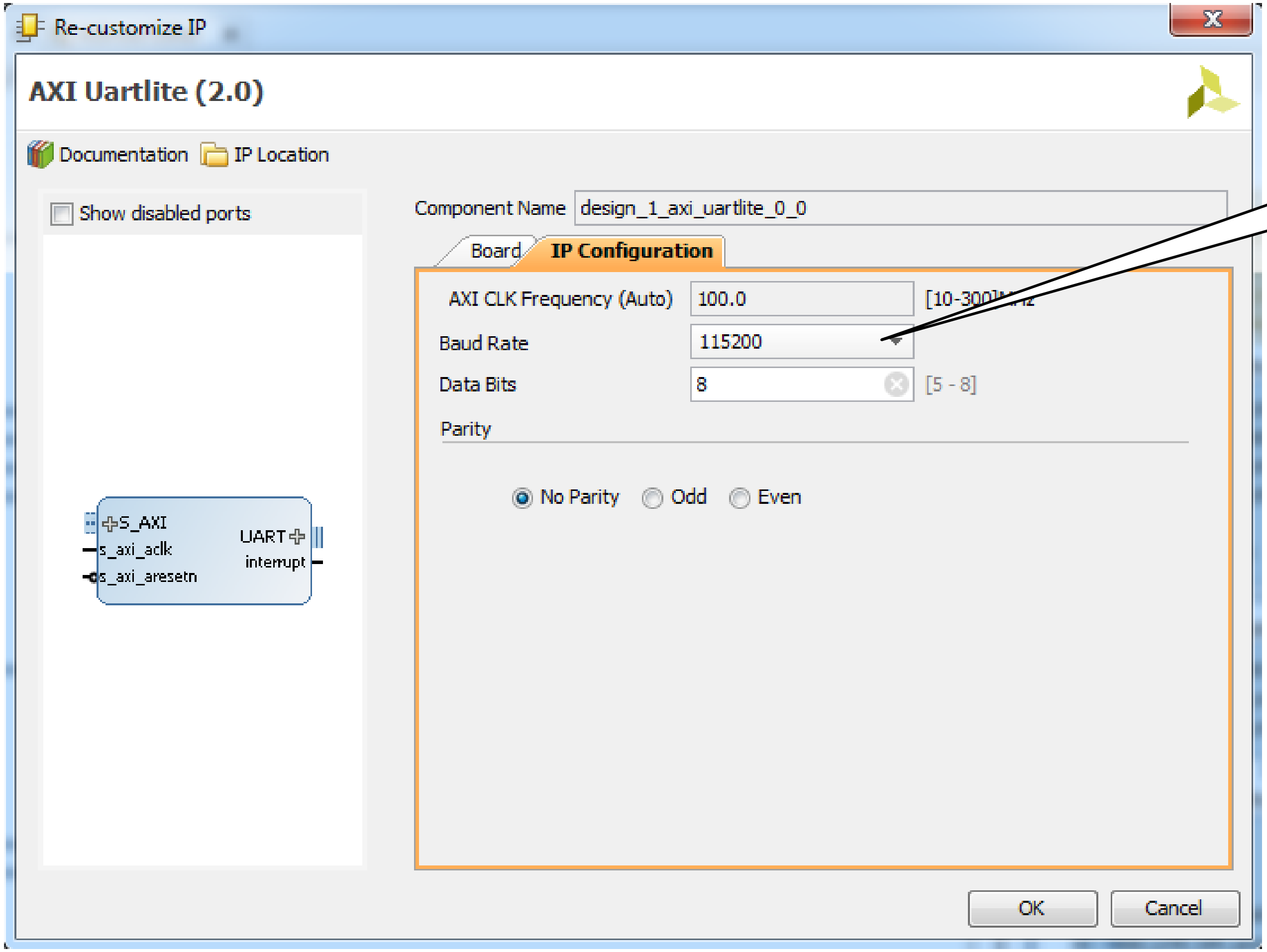
Auto In0 Width: 1 [1 - 4096]

Dout Width (Auto): 1

NOTE: The In0 port is connected to the LSB bits of the output, and the In[Number of Ports - 1] input port is connected to the MSB bits of the output.

No. of input port is 1

UART Configuration



Re-customize IP

AXI Uartlite (2.0)

Component Name: design_1_axi_uartlite_0_0

IP Configuration

AXI CLK Frequency (Auto): 100.0 [10-300] MHz

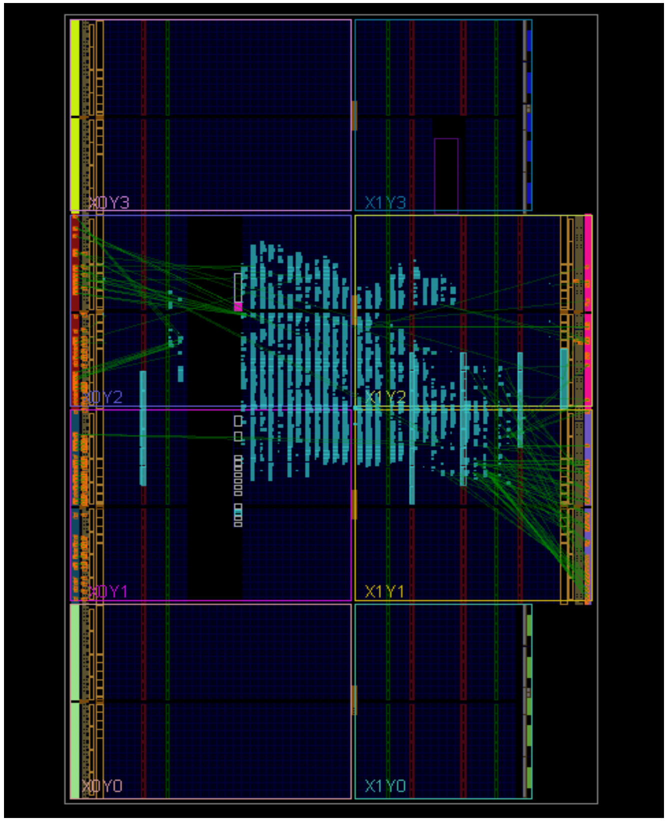
Baud Rate: 115200

Data Bits: 8 [5 - 8]

Parity: ● No Parity ○ Odd ○ Even

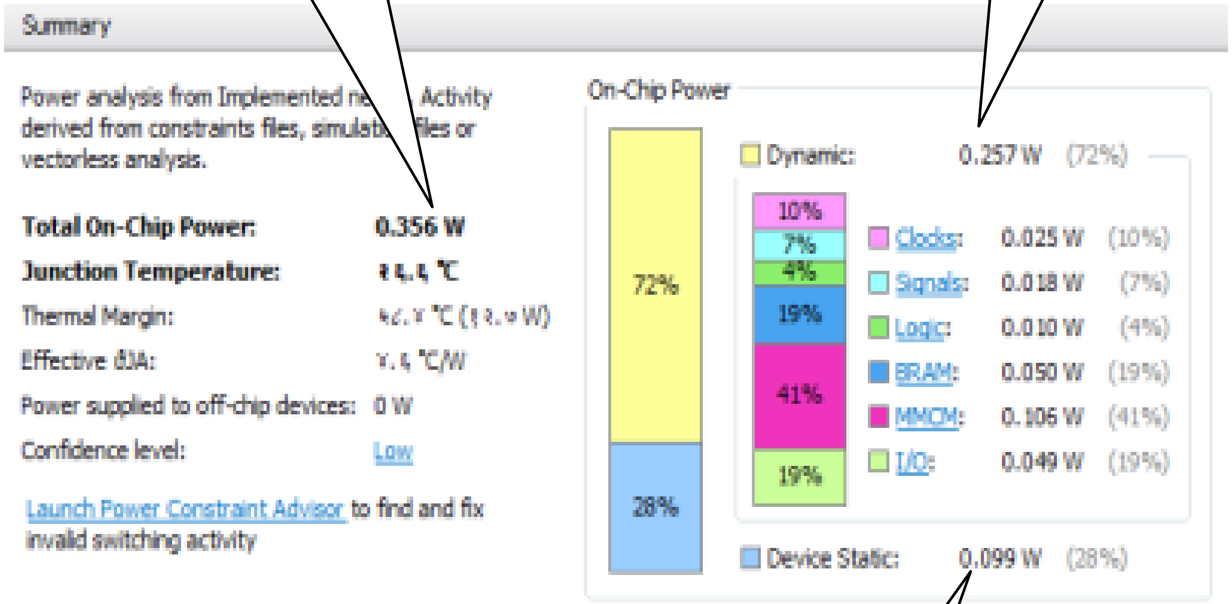
Baud Rate is 115200

On-Chip Power Summary



Total On-Chip Power
0.356W

Dynamic Power
0.257W



Summary

Power analysis from Implemented netlist. Activity derived from constraints files, simulation files or vectorless analysis.

Total On-Chip Power: **0.356 W**

Junction Temperature: **۲۴.۴ ℃**

Thermal Margin: ۶۲.۷ ℃ (۱۲.۰ W)

Effective ϑJA: ۲.۴ ℃/W

Power supplied to off-chip devices: 0 W

Confidence level: Low

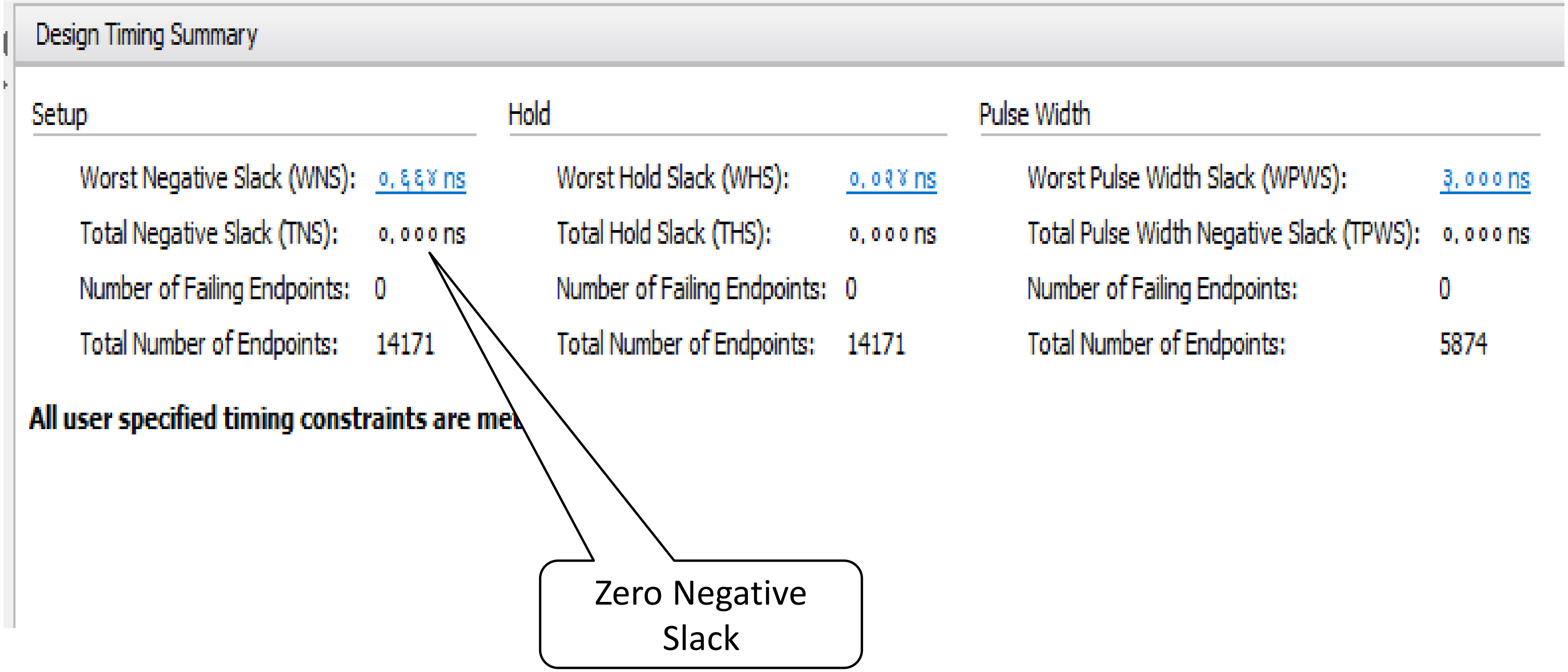
Launch Power Constraint Advisor to find and fix invalid switching activity

On-Chip Power

□ Dynamic:	0.257 W	(72%)
□ Clocks:	0.025 W	(10%)
□ Signals:	0.018 W	(7%)
□ Logic:	0.010 W	(4%)
□ BRAM:	0.050 W	(19%)
□ MMCM:	0.106 W	(41%)
□ I/O:	0.049 W	(19%)
□ Device Static:	0.099 W	(28%)

Static Power
0.099W

Design Timing Summary



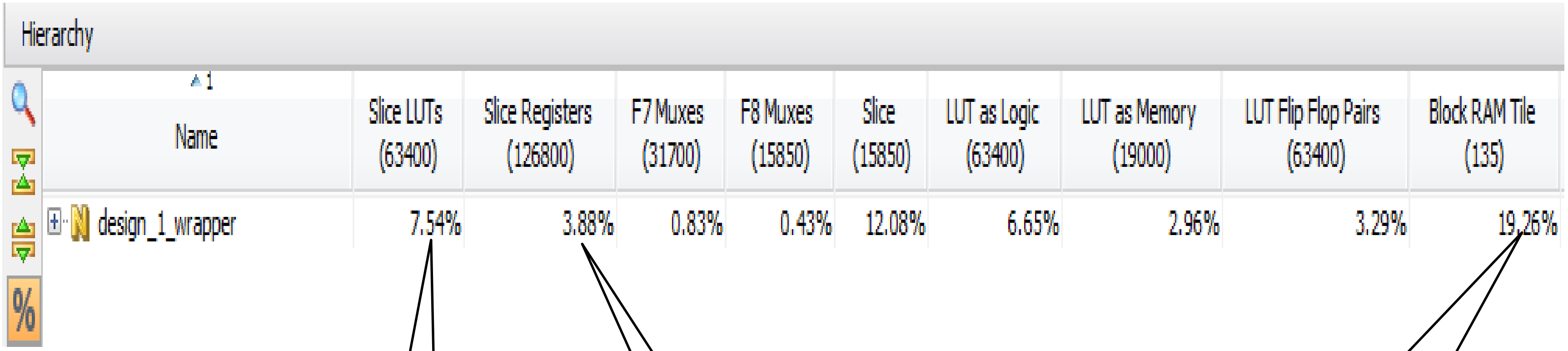
Design Timing Summary

Setup		Hold		Pulse Width	
Worst Negative Slack (WNS):	०,६६४ ns	Worst Hold Slack (WHS):	०,०२४ ns	Worst Pulse Width Slack (WPWS):	३,००० ns
Total Negative Slack (TNS):	०,००० ns	Total Hold Slack (THS):	०,००० ns	Total Pulse Width Negative Slack (TPWS):	०,००० ns
Number of Failing Endpoints:	0	Number of Failing Endpoints:	0	Number of Failing Endpoints:	0
Total Number of Endpoints:	14171	Total Number of Endpoints:	14171	Total Number of Endpoints:	5874

All user specified timing constraints are met

Zero Negative Slack

Design Utilizations



Hierarchy

Name	Slice LUTs (63400)	Slice Registers (126800)	F7 Muxes (31700)	F8 Muxes (15850)	Slice (15850)	LUT as Logic (63400)	LUT as Memory (19000)	LUT Flip Flop Pairs (63400)	Block RAM Tile (135)
design_1_wrapper	7.54%	3.88%	0.83%	0.43%	12.08%	6.65%	2.96%	3.29%	19.26%

LUTs 4780

Registers 4917

Block RAM 26

Specifications of Developed System

Table 3. Specifications of Proposed System

Parameters	Values
Pmod Wifi Range	400m
Sensor Sensitivity	0.5V/0.1mg/m3
Operating Voltage	3.3V
System Operating Frequency	100MHz
Number of access points	253

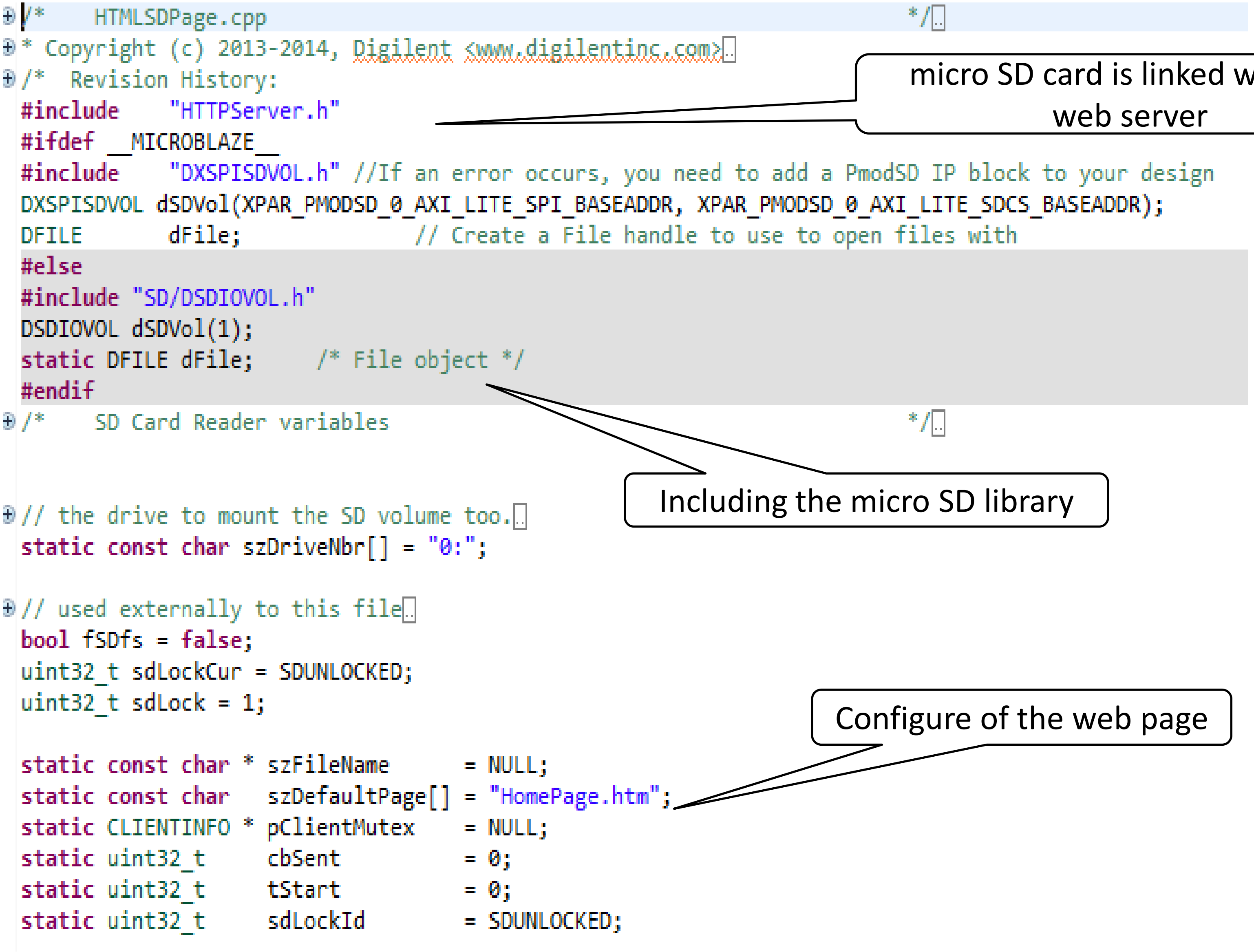
SD card Configuration

- The micro SD card is linked with the web server in which the file related to the web page is stored
- Micro SD card integration with MicroBlaze processor via the AXI bus coming from the AXI interconnect, with MicroBlaze as the master
- All of the SD pins on the FPGA are wired to support full SD speeds in native interface mode
- All html files ,photos graph, icon is linked with the web pages are stored in the SD card

Continued…

- The micro SD slot IP is configured with the HTTP Server which will allows to access data of SD card to web page and android application
- When the end user try to access information of the web page then the through the web server over the Wi-Fi network the stored data in the SD card is linked with it

Configure SD Slot IP with Web Page



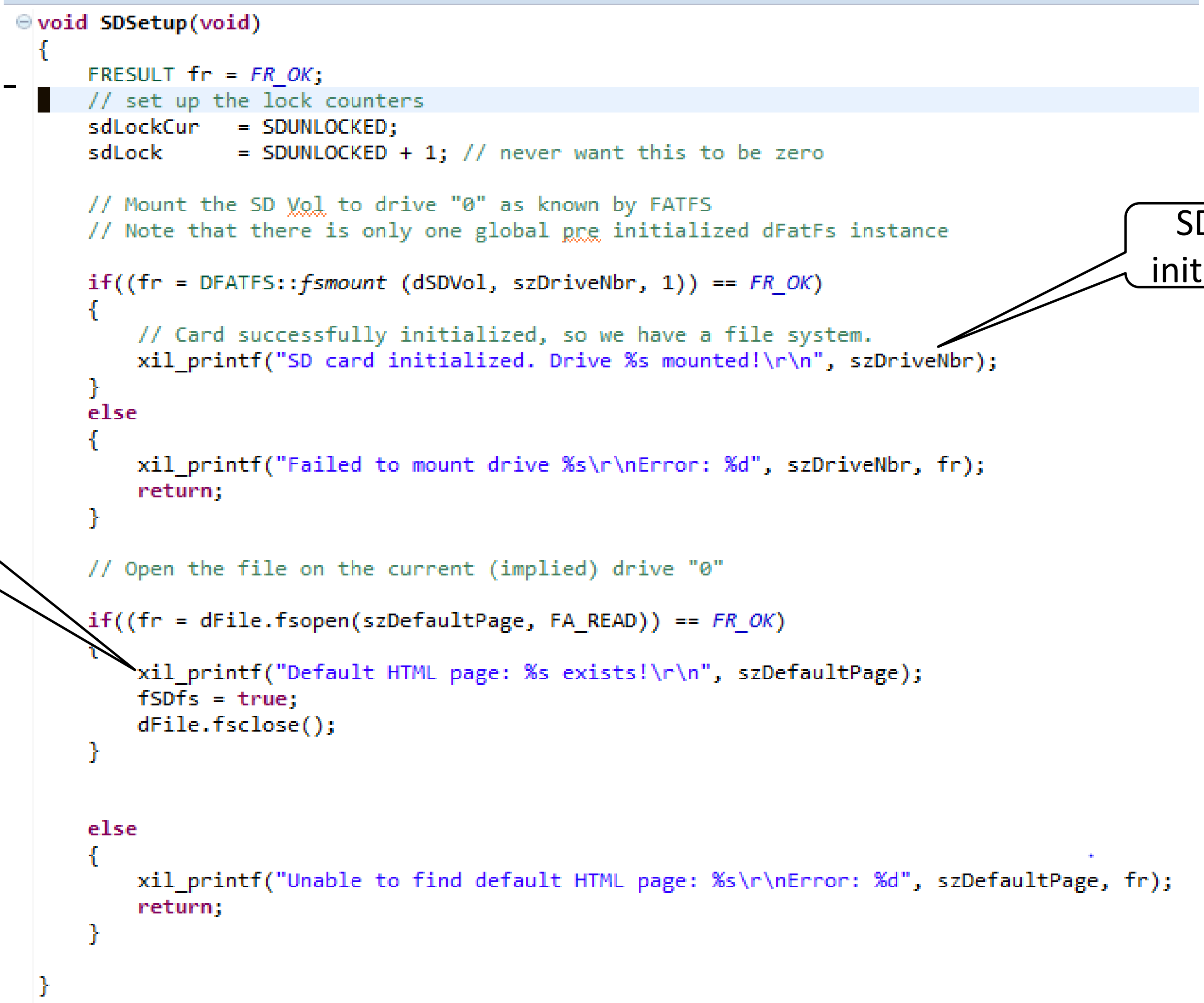
```
/*    HTMLSDPage.cpp                                                   */
* Copyright (c) 2013-2014, Digilent <www.digilentinc.com>
/*  Revision History:
#include    "HTTPServer.h"
#ifdef __MICROBLAZE__
#include    "DXSPISDVOL.h" //If an error occurs, you need to add a PmodSD IP block to your design
DXSPISDVOL dSDVol(XPAR_PMODSD_0_AXI_LITE_SPI_BASEADDR, XPAR_PMODSD_0_AXI_LITE_SDCS_BASEADDR);
DFILE       dFile;              // Create a File handle to use to open files with
#else
#include "SD/DSDIOVOL.h"
DSDIOVOL dSDVol(1);
static DFILE dFile;     /* File object */
#endif
/*    SD Card Reader variables                                         */


// the drive to mount the SD volume too.
static const char szDriveNbr[] = "0:";

// used externally to this file
bool fSDfs = false;
uint32_t sdLockCur = SDUNLOCKED;
uint32_t sdLock = 1;

static const char * szFileName      = NULL;
static const char   szDefaultPage[] = "HomePage.htm";
static CLIENTINFO * pClientMutex    = NULL;
static uint32_t     cbSent          = 0;
static uint32_t     tStart          = 0;
static uint32_t     sdLockId        = SDUNLOCKED;
```

micro SD card is linked with the web server

Including the micro SD library

Configure of the web page

SD Card Interfacing with HTTP Server



```
void SDSetup(void)
{
    FRESULT fr = FR_OK;
    // set up the lock counters
    sdLockCur    = SDUNLOCKED;
    sdLock       = SDUNLOCKED + 1; // never want this to be zero

    // Mount the SD Vol to drive "0" as known by FATFS
    // Note that there is only one global pre initialized dFatFs instance

    if((fr = DFATFS::fsmount (dSDVol, szDriveNbr, 1)) == FR_OK)
    {
        // Card successfully initialized, so we have a file system.
        xil_printf("SD card initialized. Drive %s mounted!\r\n", szDriveNbr);
    }
    else
    {
        xil_printf("Failed to mount drive %s\r\nError: %d", szDriveNbr, fr);
        return;
    }

    // Open the file on the current (implied) drive "0"

    if((fr = dFile.fsopen(szDefaultPage, FA_READ)) == FR_OK)
    {
        xil_printf("Default HTML page: %s exists!\r\n", szDefaultPage);
        fSDfs = true;
        dFile.fsclose();
    }


    else
    {
        xil_printf("Unable to find default HTML page: %s\r\nError: %d", szDefaultPage, fr);
        return;
    }

}
```

SD Cad initializing

Link data to HTML page

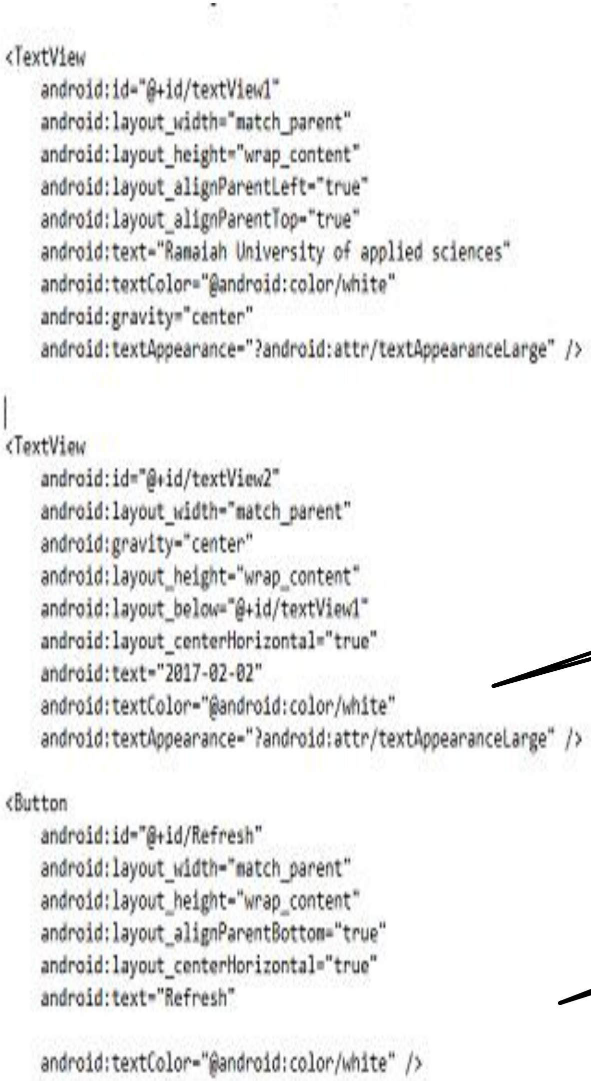
Android Application Development

- Android Eclipse is used to develop the Android application development, the Package name must be unique for every android application development
- Android SDK manager is required for different versions to be installed. Android Virtual device manager is enabled with java development kit for new virtual devices

Android Application Development

- Android operating system is widely used on hand held devices, hence monitoring of dust values will be easy when connected to internet
- The speedometer is designed for displaying the values of the dust by fetching the current dust values from the server using the IP address
- For Android 4.0 and above the application is feasible and is developed using Eclipse development kit

Android Application for Dust Monitoring System

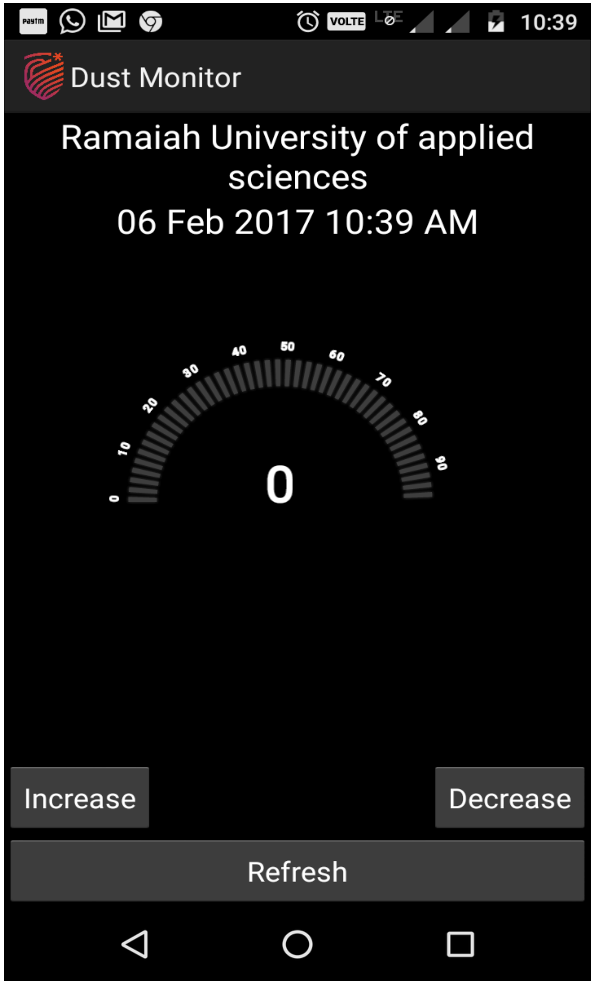


```
<TextView
    android:id="@+id/textView1"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_alignParentLeft="true"
    android:layout_alignParentTop="true"
    android:text="Ramaiah University of applied sciences"
    android:textColor="@android:color/white"
    android:gravity="center"
    android:textAppearance="?android:attr/textAppearanceLarge" />

|
<TextView
    android:id="@+id/textView2"
    android:layout_width="match_parent"
    android:gravity="center"
    android:layout_height="wrap_content"
    android:layout_below="@+id/textView1"
    android:layout_centerHorizontal="true"
    android:text="2017-02-02"
    android:textColor="@android:color/white"
    android:textAppearance="?android:attr/textAppearanceLarge" />

<Button
    android:id="@+id/Refresh"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_alignParentBottom="true"
    android:layout_centerHorizontal="true"
    android:text="Refresh"

    android:textColor="@android:color/white" />
```

College name

For System Date and Time

Refresh button



Ramaiah University of applied sciences

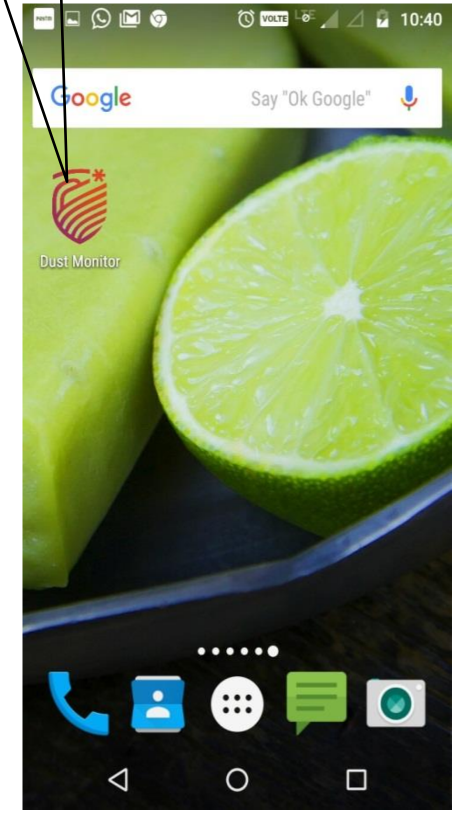
06 Feb 2017 10:39 AM

Steps for Generating .apk File

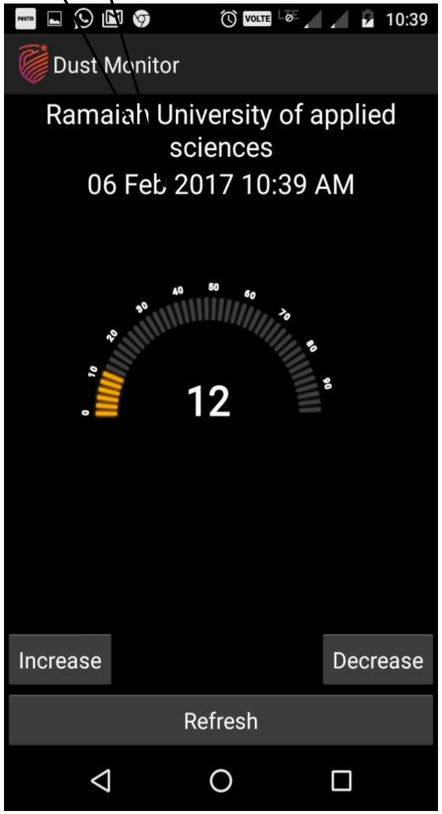
- Creating the android application package file(.apk) for signed apk by generating the code key
- The IP address used for the application is Class C static IP, which makes easier for the Private application
- For developing the application the phone has to be in developer mode
- http.h file is used to ping the IP address and to get the response from the server

Android Application for Different Dust Density

Android application on smartphone

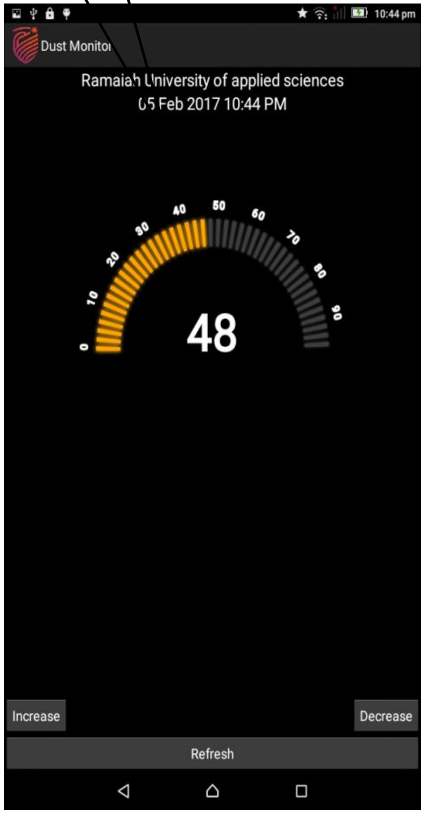


Good dust density value



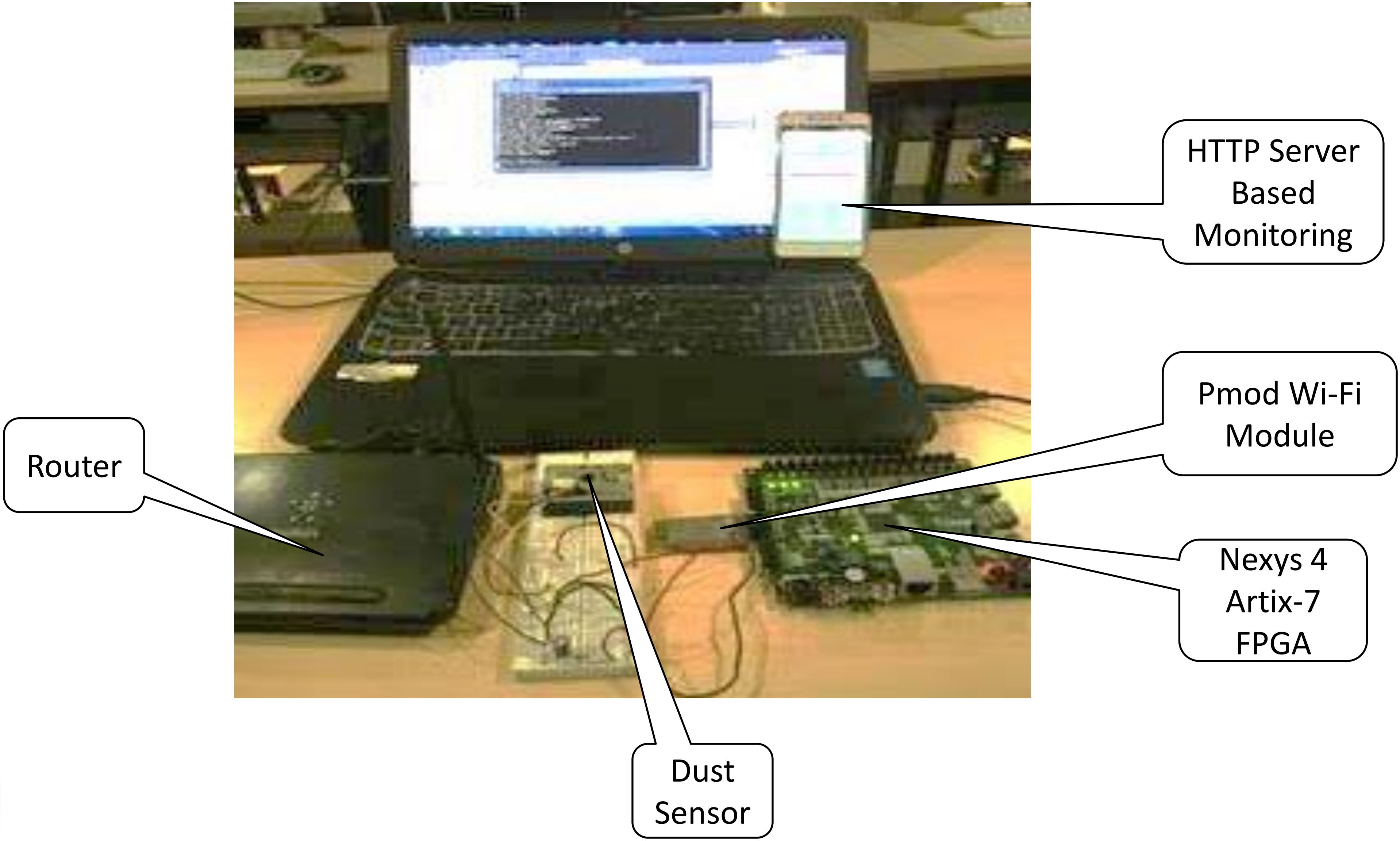
Ramaiah University of applied sciences
06 Feb 2017 10:39 AM

Moderate dust density value



Ramaiah University of applied sciences
05 Feb 2017 10:44 PM

Prototype


HTTP Server Based Monitoring
Pmod Wi-Fi Module
Router
Nexys 4 Artix-7 FPGA
Dust Sensor

Conclusion

- Dust in the environment is increasing everyday hence dust monitoring system is necessary to monitor the dust
- FPGA based dust monitoring system is successfully implemented on EDK and SDK platform
- The entire FPGA based dust monitoring system is prototyped on IoT platform by interfacing with web page and android app device
- Android application and web page is developed to display the dust density on the smartphone
- The number of users that can be connected are 253 users, Wi-fi range is up to 400 meters, the total power consumed by the device is 0.356 Watt (dynamic power=0.257 Watt, static power=0.099 Watt). The total cost involved for the Prototype development is approximately 12,000 rupees

Division of Tasks

Group Members	Task Division	Problem Solving
Bharath Ranganath B K	• Micro SD Card IP Integration with Micro Blaze • Micro SD IP interfacing with HTTP Server and Webpage in application level	• Interfacing Micro SD IP with MicroBlaze Design • Configuration of SD Card with Webpage
Dewang Shukla	• Configuration and Implementation of interface unit for MicroBlaze design • Pmod Wi-Fi interface with MicroBlaze design in application level	• Configuration of FPGA peripherals with MicroBlaze Processor • Configuration of Pmod Wi-Fi for MicroBlaze Design in application level

Division of Tasks

Group Members	Task Division	Problem Solving
Gurubasarajaiah N M	- Webpage development	- Implementation of real time graph in webpage - Linking the HTTP server data to webpage
Tousif Ahmed	- Android Application Development for the dust density monitoring	- Linking the static IP address of the server with the application
Naresh Gowda M	- Pmod XADC IP Configuration in application level	- Pmod XADC IP block Configuration - Sending data from XADC to webpage

Division of Tasks

Group Members	Task Division	Problem Solving
Prakruthi U S	• HTTP Server configuration and implementation with FPGA • Configuration of router	• Network level configuration • Interfacing server with webpage
Shahmustafa Mujawar	• Sensor verification in EDA Tool	• Analyzing behavioral model of dust sensor circuitry and its parameters

Project Schedule

Duration of Weeks	1	2	3	4	5	6	7	8	9	10
Pre-Project Presentation	🟦									
Literature Review	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	
Concept Exploration	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨		
Design Specification			🟨	🟨	🟨	🟨				
Software Simulation				🟨	🟨	🟨	🟨	🟨	🟨	
Hardware Implementation					🟨	🟨	🟨	🟨	🟨	
Final Design and Testing								🟨	🟨	🟨
Final Presentation										🟦

🟦 Presentation

🟨 To be completed

References

Digilent, (2013) Nexys4™ FPGA Board Reference Manual, Henley Court Pullman, WA.

Frost, R. (2013) *Applied Kinesiology, Revised Edition: A Training Manual and Reference Book of Basic Principles and Practices*. North Atlantic Books.

Norhafizah, R. (2015) Air Pollution Index Real Time Monitoring System, Master Thesis, University Tun Hussein Onn Malaysia.

Ricardo J., Helena S. (2016) A Wireless Biosignal Measurement System using a SoC FPGA and Bluetooth Low Energy, International Conference on Consumer Electronics-Berlin, pp. 36-40.

Sha-Sha Shelyna , S. (2015) Real Time Air Quality Reporting System, Master Thesis, University Tun Hussein Onn Malaysia.

Smyth, N. (2016) *Android Studio 2 Development Essentials*. eBookFrenzy.

Xilinx FPGA Editor, http://toolbox.xilinx.com/docsan/xilinx6/books/manuals.pdf

Xilinx Nexys-4 Pro User Guide, http://direct.xilinx.com/bvdocs/publications/ds083.pdf

Demonstration of Dust Monitoring System

Thank You