**VINAY AGGARWAL**

206/5, Block**-**3, Khurbura, Dehradun, Uttarakhand, India, 248001

Mobile: 7906460037, Email: vinay\_va12@yahoo.com, vinayagg125@gmail.com

Aadhaar Number: 425566707942, PAN : AHBPA7495D

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**EXPERIENCE SUMMARY**

I have worked for companies like **Tata Consultancy Services** and **Cognizant Technology Solutions**.

I have an experience in **Finance, Banking, Telecom, Geographic, Web and Performance Engineering** covering **Testing** (Unit Testing, Integrated Testing, System Testing, Functional Testing, Performance Testing), **Development** (C, C++, Microsoft Visual Basic, Structured Query Language (SQL)**,** Perl, Hypertext Markup Language (HTML), DSP (Digital Signal Processing**)**, Assembly Language for INTEL 8051 Microcontroller, Assembly Language for INTEL 8085 Microprocessor) and **Support** (Fidessa (Financial Trading System) servers).

**EDUCATIONAL QUALIFICATION**

* Passed **B.E (Electronics and Communication Engineering)** from Dr. K. N. Modi Institute of Engineering and Technology, Modinagar with **71 %** marks in **2000**.
* Passed **XIIth (C.B.S.E. Board)** from **S.G.R.R Public School, Dehradun** with **88% (PCM)** and **79 % (Agg)** marks in **1994**.

Mathematics: 91%

Chemistry: 90%

Physics: 83%

* Passed **Xth (C.B.S.E. Board)** from **S.G.R.R Public School, Dehradun** with **63 % (Agg)** marks in **1992**.

Mathematics: 83%

Science: 81%

**EXPERIENCE**

* Assistant System Engineer with **Tata Consultancy Services (TCS)** from Sept 2004 to Dec 2007.
* Associate Projects with **Cognizant Technology Solutions (CTS)** from January 2008 to May 2008.
* GIS Developer with **GIS Technology Centre, Survey of India** from January 2010 to June 2010.

**TRAINING**

* Got training on **Investment Banking** (Fidessa (Financial Trading System Software))inCoE (Centre of Excellence) Fidessa, Tata Consultancy Services (TCS) Pune, India.
* Got training on **Performance Testing (**Mercury Load Runner, IBM Rational Performance Tester (RPT))in Tata Consultancy Services (TCS) Mumbai, India.
* Got training on **DSP (Digital Signal Processing)** in CoE (Centre of Excellence) Embedded Systems, Tata Consultancy Services (TCS) Kolkata, India.
* Got training on **GIS (Geographic Information System)** in GIS Technology Centre, Survey of India, Dehradun, India.
* Got training on **Computer Hardware and LAN** in Oil and Natural Gas Corporation (O.N.G.C.) Dehradun, India.

**CERTIFICATION**

* **TCS** CertifiedPerformance Test Engineer. (Mercury Load Runner)
* **Cognizant** CertifiedProfessional on Performance Testing. (Mercury Load Runner)
* Load Runner **Cognizant** CertifiedProduct Consultant.

**SKILLS PROFILE**

|  |  |
| --- | --- |
| **Experience**  **(Application / Areas)** | * **Performance Testing -** Using Mercury Load Runner (LR) and IBM Rational Performance Tester (RPT). * Mercury Load Runner (LR) **-** Test Scripting (Created Virtual User Script, Selected Protocol, Selected Recording Mode (HTML based script), Data Correlation), Test Execution (Created Scenario, Executed Load test and Stress test) and Test Reports. * IBM Rational Performance Tester (RPT) **-** Test Scripting (Created Script, Selected Protocol, Data Correlation), Test Execution (Used Loops, Tester Agents, Data Pool, Executed Load test, Stress test, Endurance test and Mix Load test) and Test Reports. * **Investment Banking -** UsingRoyal Blue Fidessa (Fidessa provides Trading System, Market Data and Global Connectivity) * Royal Blue Fidessa **-** Understanding of Processes and Tables of subsystems of Fidessa (EMMA, FTS and OMAR). Hands**-**on on the Fidessa Trader’s Workbench (For Testing the behavior of different order types on stock exchange during continuous trading). Used Minder Process for Server Monitoring, Process Monitoring and for generating Summary Reports. Performed Unit Testing, Integrated Testing, System Testing and Functional Testing of Fidessa Query Tool. * **Banking -** Core Banking System Migration (Code Conversion) **-** To Review * **DSP (Digital Signal Processing) -** Using ADSP**-**21535 (Blackfin Processor), ADMC326**-**YR**-**SKD**-**1F and TMS320C6202 DSP. (DSP platform coding was done in Assembly Language and C Language) * TMS320C6202 DSP **-** For Sample Logging. * Getting basic knowledge of TMS320C6202 DSP, CCS (Code Composer Studio) and MATLAB. * Getting basic knowledge of Sample Log, Data Block, Memory Allocation, FIFO (First Input First Output), DMA and Interrupts. * ADSP**-**21535 (Blackfin Processor) **-** For Network Camera. * Understanding of Datasheet (Technical Data) of ADSP**-**21535. (Instruction Set, Pin Config, VisualDSP++ Development Environment) * ADMC326**-**YR**-**SKD**-**1F (Universal embedded controller) **-** For Inverter * Understanding of Datasheet (Technical Data) of ADMC326**-**YR**-**SKD**-**1F. (Instruction Set and Pin Configuration) * Performed ROM Programming (Serial ROM I2C-24C16). * **INTEL 8051 Microcontroller -** Development of code using Assembly Language of INTEL 8051 Microcontroller. * **GIS (Geographic Information System) -** Using Esri ArcGIS. * Esri ArcGIS **-** Scanned the map for obtaining geographical data on the system. Digitized geographical data. Annotated geographical data. * **Web Page Development -** Developed Web Pages using HTML. |
|  |  |
| **Operating System** | Windows, DOS, UNIX |
|  |  |
| **Languages** | C, C++, Microsoft Visual Basic, Structured Query Language (SQL)**,** Perl, HTML, DSP (Digital Signal Processing**)**, Assembly Language for INTEL 8051 Microcontroller, Assembly Language for INTEL 8085 Microprocessor |
|  |  |
| **Tools** | Mercury Load Runner, IBM Rational Performance Tester (RPT), Mercury Test Director, Mercury Win Runner, Esri ArcGIS **(**Geographic Information System) |

# PROJECT DESCRIPTION

|  |  |
| --- | --- |
| **Title** | Geographical Data Digitization |
|  |  |
| **Domain** | **Geographic** |
|  |  |
| **Responsibilities** | * Used Scanner for scanning the map for obtaining geographical data on the system. Scanning converts paper maps into digital format by capturing features. * Used **Esri ArcGIS** for digitization of geographical data. (ArcGIS is a geographic information system (GIS) for working with Maps and Geographic information. (A geographic information system (GIS) is a system designed to capture, store, manipulate, analyze, manage and present all types of geographical data.)) * To bring the scanned file on the **Esri ArcMap** for digitization of geographical data. (ArcMap is used to view,edit,create and analyze geospatial data) * To Add the Style File. * To digitize the Geographical Data. (Clicked the Editor button in the Editor Toolbar. Chosen Start Editing) * Digitized the Ward Boundary Line. * Digitized the EB (Enumeration Block) Line. * Digitized the House, Important Building (IMPBLD) and Other Building (OTHERBLD). * Digitized the Roads, Rivers, Park, Johars, Stairs and Bridge. * Used **Esri ArcCatalog** (Database for ArcGIS) for annotation of geographical data. Used Microsoft Office Access Application for Table Generation (.mdb File). (The ArcCatalog application provides a catalog window that is used to organize and manage various types of geographic information for ArcGIS for desktop) * Annotation of House, Important Building (IMPBLD) and Other Building (OTHERBLD). * Annotation of Roads, Rivers, Johars, Stairs and Bridge. * Annotation of EB (Enumeration Block). * Used Text Tool from Toolbar for annotation in the outside of the Ward Boundary. * To draw Grid and to adjust Digitized File in the Grid. * To check the following in the Digitized Map. * Check for the Label of House, Other Building (OTHERBLD) and Important Building (IMPBLD). * Check for the Label of Roads, Rivers, Johars, Stairs and Bridge. * Check for the Label of EB (Enumeration Block). Check for the Legend. * Check for Ward Boundary and EB (Enumeration Block) Boundary. * Check for Inside and Outside annotation of Ward Boundary. * Check for Top Annotation. * To check the Grid in print preview. To take Print Out of the Digitized Map. * To check the Print Out of the Digitized Map. * To get the basic knowledge of Scanning and Printing a Map. * Prepared Documents for the whole system. (ArcGIS **(**Geographic Information System)) * Scanner use, ArcMap Digitization, Whole Ward Layout, Separate EB Layout, Joining ID’s and Merger. |
|  |  |
| **Project** | To Digitize the geographical data. (Digitizing in geographic information system (GIS) is the process of converting geographic data either from a hardcopy or a scanned image into vector data by tracing the features) |
|  |  |
| **Work Location** | GIS Technology Centre, **Survey of India**, Dehradun, India |

|  |  |
| --- | --- |
| **Title** | Wyndham WVO QA FB Performance Testing |
|  |  |
| **Client** | **Wyndham Hotel Group** |
|  |  |
| **Domain** | **Performance Engineering** |
|  |  |
| **Responsibilities** | * To understand the application. The main interfaces which needed to be tested were * Logging and Logout, Creating and Modifying User, Sending News and Alerts. * Browsing of links like Reports, Training, Finance, Forms and Tools. * Used **Mercury** **Load Runner** for doing Performance Testing of application. * **Test Scripting -** Used **Mercury Virtual User Generator (Vugen)**. * For creating Virtual User (Vuser) Script. Vugen records actions into Vuser Scripts. Each Vuser script has three default Functions * Vuser\_init (Used to record login to a server), Action (Used to record user activity), Vuser\_end (Used to record logoff procedure). * For replaying of Virtual User Script for one Virtual User only to ensure that script is recorded correctly. * For selecting Protocol**.** (For client server communication) * For selecting Recording Mode **-** Selected the ‘HTML based script’ mode. * For Data Correlation **-** Selected the ‘Enable correlation during recording’. * For Think Time **-** Selected the ‘Record think**-**time’ option. * **Test Execution -** Used **Mercury Load Runner Controller**. * For selecting a Scenario **-** Selected the ‘Manual Scenario. * For adding Vuser script in scenario. (Group Name column shows script name) * For using Load generators to create load on the application by running Vusers. * For Executing the Scenario for **Load Testing** (Number of concurrent users a system can handle) on Virtual User (Vuser) Scripts. * Runtime Settings (Script created for only 1 Vuser, Iteration any number) * Run Logic (Number of Iterations) **-** 10, 20, 30 * Pacing **-** Start new iteration at fixed intervals, every 10 Seconds. * Think Time **-** Selected the Replay think time, As recorded option. * **Mercury Load Runner Controller Schedule Builder** Settings (For configuration of Schedule **-** Schedule byScenario) * Ramp Up Tab **-** Start 1 Vuser every 10 Seconds. * Duration Tab **-** Run for 3 minutes after ramp up has been completed. * **Test Reports -** Used **Mercury Analysis** for generating Reports and Graphs which are used to identify the bottlenecks in application and to determine what changes need to be made in system to improve its performance. * Summary Report, Running Vusers, Hits per Second, Throughput, Average Transaction Response Time, Transaction Summary and Window Resources. * Running Vusers Graph **-** Displays the number of Vusers that executed Vuser scripts and their status, during each second of a load test. * Hits Per Second Graph **-** Displays the number of hits made on the Web server by Vusers during each second of the load test. * Transaction Summary Graph **-** Displays the number of transactions that passed, failed, stopped, or ended with errors. |
|  |  |
| **Project** | The objective of project was to measure the performance of application, identify limitations and suggest corrective measures before deployed into production. |
|  |  |
| **Work Location** | **Cognizant Technology Solutions (CTS)**,Kolkata, India |

|  |  |
| --- | --- |
| **Title** | TRGGER **-** CVMS Interface |
|  |  |
| **Client** | **TRG (Title Resource Group)** |
|  |  |
| **Domain** | **Performance Engineering** |
|  |  |
| **Responsibilities** | * To understand the application. The main interfaces which needed to be tested were * Orders created in TRGGER by PHH and CVMS, Orders updated from ICD to TRGGER, Order response from TRGGER to CVMS and PHH. * Used **Mercury** **Load Runner** for doing Performance Testing of application. * **Test Scripting -** Used **Mercury Virtual User Generator (Vugen)**. * For creating Virtual User Script **-** 4 Scripts (Web Services Protocol), 15 Scripts (Web (HTTP/HTML) Protocol) and 11 Scripts (ODBC Protocol). * For selecting Protocol.(For client server communication) * For replaying of Virtual User Script for one Virtual User only to ensure that script is recorded correctly. * For selecting Recording Mode **-** Selected ‘HTML based script’. (HTML based mode records script for every user action that is performed during recording) * For Data Correlation **-** Selected the ‘Enable correlation during recording’. (The Build**-**in correlation detects and correlates dynamic data) * For Think Time **-** Selected the ‘Record think**-**time’ option. * **Test Execution -** Used **Mercury Load Runner Controller**. * For selecting a Scenario **-** Selected the ‘Manual Scenario. (Manage Load test by specifying the number of Virtual Users to run) * For adding Vuser script in scenario. (Group Name column shows script name) * For using Load generators to create load on the application by running Vusers. * For Executing the Scenario for **Load Testing** on Vuser Scripts for 12 users. * Runtime Settings (Script created for only 1 Vuser, Iteration any number) * Run Logic (Number of Iterations) **-** 5, 10, 12. * Pacing **-** Start new iteration at fixed intervals, every 10 Seconds. * Think Time **-** Selected the Replay think time, As recorded option. * **Mercury Load Runner Controller Schedule Builder** Settings (For configuration of Schedule **-** Schedule byScenario) * Ramp Up Tab **-** Start 1 Vuser every 10 Seconds. * Duration Tab **-** Run for 3 minutes after ramp up has been completed. * Data Access Method for parameterization. (Parameter in Vugen is a container that contain a recorded value that is replaced for various users) * Select next row **-** Sequential * Update value on **-** Each Iteration, Each Occurrence and Once. * For Executing the Scenario for **Stress Testing** on Virtual User Scripts. (Executed the Load Test with more than the maximum number of users) * **Test Reports -** Used **Mercury Analysis** for generating Performance Test Reports. * Summary Report, Running Vusers, Hits per Second, Throughput, Average Transaction Response Time, Transaction Summary and Window Resources. |
|  |  |
| **Project** | The scope of the performance testing on the TRGGER application was to simulate load on the system for up to 12 active users and to evaluate the application performance including the interface activities to the system by defining the Scalability, Stability and Reliability of an application. |
|  |  |
| **Work Location** | **Cognizant Technology Solutions (CTS)**, Kolkata, India |

|  |  |
| --- | --- |
| **Title** | Ultimatix **-** GLOBAL SPEED |
|  |  |
| **Domain** | **Performance Engineering** |
|  |  |
| **Responsibilities** | * To understand the application. (The Functional Flow of application) * This is a complete system for goal setting and employee performance review during quarterly, project end, half yearly and year end appraisal processes. * To Prepare the Performance Test Plan for the project. * Used **IBM Rational Performance Tester (RPT)** for doing Performance Testing. * **Test Scripting -** Used **IBM Rational Performance Tester Recorder**. * For creating Performance Test Scripts. (Scripts are created by recording the user actions which are to be tested) * For selecting the Recorder Type **-** Selected the ‘HTTP Recording’ option. (Creates a recording of the HTTP Protocol) * For Data Correlation**-**In application each request depends on information from a previous response. Internal linking of response & request data is correlation. * Checking of correlation. Modification of script. (Checking of session ID ) * For creating Data Pool which was used for creating set of Test Data that was used by script at run time. (If there are 20 records in data pool for one script, that script will be executed 20 times for a single user if data pool is linked to that script) * **Test Execution -** Used **Performance Schedule**. * For creating a Schedule. * For adding a Performance Test Script in a Schedule. * For adding a Loop for the Performance Test Script. Used Loop for repeating the Test Execution at least 3 times to get the exact data. * For using **IBM Rational Performance Tester Agents**. * For Generating Load on the application. * For Distributing the Load. (Load is divided equally between Agents) * For Executing **Load Test** (Number of concurrent users a system can handle) for critical business transactions of application. Schedule Element details were * Number of users **-** 1, 50, 100, 250, 500, 750, 1000, 1250, 1500, 2000. * Delay between starting each user **-** 100 Milliseconds. * Think Time **-** Used the ‘Use the recorded think time’ option. * For Executing the **Stress Test** for critical business transactions of application. (Executed Load Test with more than the maximum number of users) * For Executing the **Endurance Test** for the critical business transactions of the application. (Executed the Load Test for an extended period of time) * **Test Reports -** Overall, Summary (Run Summary, Page Element Summary, Page Summary), Page Performance, Response Vs Time Summary, Response Vs Time Detail, Page Throughput, Server Health Summary and Server Health Detail. * To get the **Glance Report** for server side Performance Metrics. * CPU Utilization and Memory Utilization * To Perform Performance Testing in **Staging Environment**. (Performance Testing is to check performance of a new application before a new application goes to production) * Staging Environment is nearly exact replica of a production environment. |
|  |  |
| **Project** | The project was to do the Performance Testing of the critical business transactions of the application. |
|  |  |
| **Work Location** | **Tata Consultancy Services (TCS)**, Gurgaon, India |

|  |  |
| --- | --- |
| **Title** | Fidessa Query Tool |
|  |  |
| **Client** | **Lehman Brothers** |
|  |  |
| **Domain** | **Finance (Investment Banking)** |
|  |  |
| **Responsibilities** | * To understand the application. (The Functional Flow of application) * Used **Microsoft Visual Studio** for designing of Tool. * Used **Microsoft Visual Basic** Form for designing of front end. (Used Standard Exe to create a standard executable) * Added controls on form by simply double clicking on control in Toolbox.  Toolbox contains all controls available to be added to a form. (Control such as Label) * Clicked the control on the Form and dragged it to the desired location. * Used Tab Order button on layout Toolbar for configuring the Tab Order in a form. (Tab Order is implemented for sensible sequence of moves between controls) * Used layout Toolbar for aligning groups and sizing groups of controls in a form. (For example, all the controls can be left or right aligned and resized to the same size, width or height) * Used Grid for Positioning and Sizing Controls **-** Activated a grid for Positioning and Sizing Controls. When the grid is active, controls snap to the nearest grid position when added to a form. * Used **Microsoft Visual Studio** for development of code for the Tool. * Used **Structured Query Language (SQL)** for Fetching data (Order and Trades) from Sybase Database. (Data is displayed in simple Grids on Visual Basic Form) * Used **Microsoft Visual Basic** for developing code for other actions. * Performed **Unit Testing** (Testing of single module or unit of code), **Integrated Testing** (Individual units of a program are combined and tested as a group), **System Testing** (Testing of complete and integrated software) and **Functional Testing** (To verify that the system meets the Functional requirements) of application. (Testing is basically done to ensure that the system meets the specified requirements) * Prepared Test Cases for performing Unit Testing, Integrated Testing, System Testing and Functional Testing of application. * Execution of Test Cases during Unit Testing, Integrated Testing, System Testing and Functional Testing of application. * To get the knowledge of Trade. The Trade contains the following key information. * Book **-** All trades are entered against a trading book. When the trade is done, the positions in the book are affected. * Instrument **-** All trades are result of trading on a simple instrument. Instrument may be available on a single book or multiple books. When trade is done, the position in instrument as well as trading book is affected. (Instrument like Equities) |
|  |  |
| **Project** | To Develop and Test a standalone Visual Basic application that would allow users to retrieve information from the Sybase database.  This tool will help in retrieving old audit trail data on orders and trades. The data will be retrieved from the database and displayed in the simple grids on Visual Basic form.  Input would be either Order ID (When the order is received by the exchange, the exchange order id is issued) or Trade ID (When the order gets executed at exchange, a unique trade set id is allocated) or combination of the Book, Instrument, User, Counterparty (With which trade has been done) and Date Fields (It is the trade date). |
|  |  |
| **Work Location** | **Tata Consultancy Services (TCS)**, Bangalore, India |

|  |  |
| --- | --- |
| **Title** | Fidessa (Financial Trading System Software) |
|  |  |
| **Client** | **Morgan Stanley** |
|  |  |
| **Domain** | **Finance (Investment Banking)** |
|  |  |
| **Responsibilities** | * Understanding of **Subsystems of Fidessa**. (EMMA, FTS, OMAR) * **EMMA** (European Multi Market Access) **-** It provides trading access to stock exchanges. It also provides full market depth which includes the list of stocks available in the market, bids and offers available, last traded price and open price. * **OMAR** (Order Management and Routing) **-** It manages order flows and route them to various execution services. * **FTS** (Fidessa Trading System) **-** It is the central repository of all the trading activities in the system. It captures trade information, market price information in real time and calculates the position, average price, profit and loss. * To Test the behavior of different order types on stock exchange during continuous trading. (Each order sent to the exchange is of particular type, based on the behavior of which, exchange will execute the order against the orders already present in order book) * Selected the Order Sides **-** Buy/Sell. (On the market depth, double click on the bid for putting a sell order or double click on the offer for putting a buy order) * Selected the Order Type **-** At best, at market, Concealed, Execute and Eliminate, Hidden and Limit. * Placed the Ordered Volume and Ordered Price. Checked the Expected Result. * Done support for Fidessa servers **-** By executing Unix commands in **Windows Command Prompt**.(Server start/close commands, Process start/restart commands) * All subsystems reside on **UNIX** server. Each subsystem has a pair of servers running. One is primary server and other is the Standby Server. The primary server is always running and in case of failure of primary, standby will start. Each server has a real time database and a number of processes. * Used Minder Process for Server Monitoring, Process Monitoring and for generating Summary Reports **-** By executing Unix commands in **Windows Command Prompt**. (Gives the List of Fidessa Processes, Start time, Command line parameters, summary reports for primary/standby server) * The Minder Process always runs in background, even when the server is closed. It checks health of server & also of individual processes & provides summary report. * To get the basic knowledge of Processes and Tables of subsystems of Fidessa. * To get the basic knowledge of Fidessa Databases, Interface Processes and Protocol. * **RTD (Real Time Database) -** The Database used in all Fidessa servers is a Memory based database called Fidessa RTD. (A real time database works on live & constantly changing data. RTD which is build for speed resides in real memory.) * **Sybase Database -** This database is referred as **Fidessa Data Architecture (FDA)** which holds standing data. (The Sybase Database resides on Disk) * **FidToSyb Process -** Process sends data from Fidessa RTD to Sybase (FDA). * **Open Access Protocol -** This is the protocol used for communicating between the processes on a server or processes between two servers. |
|  |  |
| **Project** | To Understand the subsystems of Fidessa (EMMA, FTS, OMAR) **-** Functional Workflow & Technical Architecture. Hands**-**on on the FTW (Fidessa Trader’s Workbench).  To Test behavior of order types on stock exchange during continuous trading. |
|  |  |
| **Work Location** | CoE Fidessa, **Tata Consultancy Services (TCS)**, Pune, India |

|  |  |
| --- | --- |
| **Title** | CBS Migration |
|  |  |
| **Client** | **Lloyds TSB Bank** |
|  |  |
| **Domain** | **Banking** |
|  |  |
| **Responsibilities** | * Used the following tools for Reviewing: * StanCheck **-** It is a tool for checking C coding standards. It can run on multiple Modules in a single run. * RunCheck **-** It is atool to compare the C and MASM results. It works in two phases. First it generates the runstreams then from that runstreams the C and MASM files are created in mainframe which are then FTP (File Transfer Protocol) to Desktop for comparing (To find out the mismatches). It saves lot of time as manually comparing the C and MASM results takes lot of time. It can be applied on any number of Modules. * MissingVariable **-** It is a tool for checking the missing variable. It saves time as manually finding the missing variable takes lot of time. It can be applied on any number of Modules. * PackageVerify **-** It is a tool to verify that Package has no missing Element. With this tool any missing or extra element can be found in all the contents present in Module. This saves time as manually verifying the missing or extra element in all the contents present in Module takes a lot of time. It can be applied on any number of Modules. * Preparation of Reviewing Report. * To get the basic knowledge of **Perl**. * Perl is programming languages specially designed for text editing. * Programs written in Perl are called Perl scripts. * Perl is an interpreted programming language with fast execution time as there is no need to compile a Perl script. * Perl is used for developing a variety of software applications. Perl is simple and easy to program and understand. * Perl is used for extracting information from a text file and printing out a report or for converting a text file into another file. * Perl is the best language for File Handling, Text Processing and Output Reporting. * To get the basic knowledge of **Mainframe**. * A Mainframe computer is a very large computer capable of handling and processing a very large amount of data quickly. * A Mainframe computer is used when high speed is required. * The high Stability and Reliability of a Mainframe computer enable these machines to run uninterrupted for a very long period of time. * FTP (File Transfer Protocol) is used to transfer files from and to mainframe computers. (FTP is the TCP/IP application for moving files between computers) |
|  |  |
| **Project** | The project is the Conversion of the **Core Banking System (CBS)** Software Environment from Basic Mode to Extended Mode. The Core Banking System (CBS) of Lloyds TSB Bank is a retail Banking application.  The project involves conversion of legacy system of Lloyds TSB Bank written in Assembly Language into C. Over 490 modules containing 238,247 lines of code constituted part of the system to be converted. |
|  |  |
| **Work Location** | **Tata Consultancy Services (TCS)**, Gurgaon, India |

|  |  |
| --- | --- |
| **Title** | Sample Logging DSP Enhancements |
|  |  |
| **Client** | **Aeroflex** |
|  |  |
| **Domain** | **Telecom (Embedded Systems)** |
|  |  |
| **Responsibilities** | * To get the basic knowledge of Sample Logging. * Sample Logging is a method of characterizing and detecting errors in signal arriving at Rx DSP (Receiver DSP) or generated at Tx DSP (Transmitter DSP). * To get the basic knowledge of **Sample Logging DSP (SL-DSP).** * Sample Logging DSP captures real time baseband samples. * The input signal to the SL**-**DSP is typically a baseband Tx output/ Rx input. * To get the basic knowledge of Sample Log/Data Block **-** The first step in the processing is to convert the number of words in the data block to the number of samples. * To get the basic knowledge of TMS320C6202 DSP **-** 256KB of Internal Program Memory, 64KB of Internal Data Memory and 16MB of External SDRAM. * To get the basic knowledge of memory allocation of Sample Logging DSP. * Sample Logging DSP has a fixed memory allocated for storage of captured samples. It captures data into its internal memory and if the data exceeds then it captures data into its external memory. Memory Allocation depends on Sample Rate (Time Duration of Data) and Input Format (No of Bytes / Sample). * To get the basic knowledge of FIFO (First In First Out). * It is used for communication with other DSP processors. It transfers data to and from DSP. Each DSP has access to bidirectional FIFO. Data is in FIFO. * To get the basic knowledge of DMA (Direct Memory Access). * DMA1 **-** It copies samples from incoming FIFO to Internal Memory. * DMA3 **-** It copies samples from Internal Memory to External SDRAM. * To get the basic knowledge of Interrupts coming to DSP. * Host Interrupt**-**Writes new command, ReadDataFromFIFO**-**Process new command * ReadNewMail **-** Forms a mail message and send it to BBM. * DataReadComplete **-** Posted after DMA read operation from FIFO is completed. * To study the application programs. * To get the basic knowledge of Sample Logging Header Files**.** (.h Files) * DataBuffer, DspProcess, DspTime, DynamicMemorySize, Interrupts, SampleRates, SampleLogData and SampleloggingProcess. * To get the knowledge of GEL Files **-** Created for running Test application in DSP. * To get the basic knowledge of **CCS (Code Composer Studio)** and **MATLAB**. * To get the basic knowledge of Unit Testing. * Getting knowledge of Input (.Inp), Expected Output (.Exo) and Actual Output (.Aco)) Files. Running of Unit Test Function. * Comparing of Expected Output (.Exo) and Actual Output (.Aco) Files. * On Code Composer Studio, a small window below shows test passes or fails. * To enhance the code as per requirement. * To get the knowledge of hardware used **-** Baseband module (BBM) Board (One DSP), Evaluation module (EVM) Board (Eight DSP (Four on Tx side and Four on Rx side)). |
|  |  |
| **Project** | The project involved DSP Development and Test Strategy (source code, test harnesses, test vectors, models used to generate test vectors). |
|  |  |
| **Work Location** | CoE Embedded Systems, **Tata Consultancy Services (TCS)**, Kolkata, India |

|  |  |
| --- | --- |
| **Title** | DSP Inverter (Full Bridge, Sine Wave) |
|  |  |
| **Domain** | **Telecom (Embedded Systems)** |
|  |  |
| **Responsibilities** | * To get the basic knowledge of sine wave DSP inverter. * Got knowledge of ADMC326**-**YR**-**SKD**-**1F **(Universal Embedded Controller)**. ADMC326**-**YR**-**SKD**-**1F is a very powerful DSP processor. (DSP platform coding was done in Assembly Language and C Language) * To Understand the **Datasheet (Technical Data)** of **ADMC326-YR-SKD-1F**. * To get the basic knowledge of features of ADMC326**-**YR**-**SKD**-**1F. * 20 MIPS Fixed**-**Point DSP Core, ADSP**-**2171 Code Compatible * Independent Computational Units * ALU, Multiplier/Accumulator Unit, Barrel Shifter * Single Cycle Instruction Execution (50ns), Multifunction Instruction * Two Independent Data Address Generators * Memory Configuration **-** 512 x 24 bit PM RAM, 12 x 16 bit DM RAM, 4096 x 24 bit PM masked ROM * To get the knowledge of Instruction Set of ADMC326**-**YR**-**SKD**-**1F. * Multiplier and Accumulator Functions (Performs single**-**cycle multiply, multiply/add, and multiply/subtract operations) * Standard ALU Functions (Performs standard arithmetic & logic operations), Data Move (For data moving), Barrel shifter (Shifts bits) * To get the knowledge of Pin Configuration of ADMC326**-**YR**-**SKD**-**1F. * Got knowledge of Functional Working of a sine wave DSP inverter. * Inverter is a device that converts DC supply into AC supply. Battery (24V) is used for DC supply. If AC is present, Inverter gives output according to AC input, whenever AC supply is not present inverter starts functioning. * ADMC326**-**YR**-**SKD**-**1F has built**-**in ADCs (Analog to Digital Converter) and PWM (Pulse Width Modulation) block for the generation of PWM signals required for extremely fast switching of power devices (MOSFETs). * Used .bas File (Basic Source Code File) for communication. * Communication port is a serial port communication used to connect devices. Communication ports are designated as COM1, COM2, COM3, and COM4 on computer running Microsoft windows. * Performed **ROM Programming. (Serial ROM I2C-24C16)** The information is embedded in ROM in the form of bits by a process known as programming the ROM. * .Bin File is programmed. The BIN file is a binary file that contains binary data that is used by different applications. * Performed Testing of sine wave DSP inverter. * To get the knowledge of hardware used **-** ADMC326**-**YR**-**SKD**-**1F, Mosfet, Diode, Resistor, Capacitor, Inductor, Rectifier, Condenser, Zenor Diode, Regulator, Transistor, Gate Driver, Opto Coupler and Op**-**Amp. * To see components whether they are correctly installed or not. (Visual Inspection) * To perform the Low Voltage Testing of a sine wave DSP inverter. (5V, 12V) |
|  |  |
| **Project** | It is a 24V, 1KW, 1250VA low distortion inverter based on DSP Technology. This inverter gives a sine signal. The inverter was developed by using ADMC326**-**YR**-**SKD**-**1F.  ADMC326**-**YR**-**SKD**-**1F is a masked ROM component of ADMC326. ADMC326 is a 28**-**Lead ROM**-**Based DSP Motor Controller. |
|  |  |
| **Work Location** | Roorkee, India |

|  |  |
| --- | --- |
| **Title** | Network Camera |
|  |  |
| **Domain** | **Telecom (Embedded Systems)** |
|  |  |
| **Responsibilities** | * To get the basic knowledge of **Image Processing Development kit**. * Got knowledge of ADSP**-**21535. ADSP**-**21535 is one of the Processor of Blackfin Family. (DSP platform coding was done in Assembly Language and C Language) * To Understand the **Datasheet (Technical Data)** of **ADSP-21535**. * To get the knowledge of **VisualDSP++ Development Environment**. * The VisualDSP++ development environment consists of a full range of tools such as C/C++ compiler, linker, debugger and simulator. (Compiler converts the C/C++ code to DSP Assembly) * The VisualDSP++ development environment gives the ability to perform advanced application code development and debug such as * Create, compile, assemble and link application programs written in C and ADSP**-**21535 assembly. * Load, run, step, halt and set breakpoints in application programs. * Read and write data and program memory. * Read and write core and peripheral registers. * To get the basic knowledge of DSP core of ADSP**-**21535. (RISC like Register and Instruction model for ease of programming) **-** Two 16**-**bit MAC, Two 40**-**bit ALU’s, Four 8**-**bitVideo ALU’s and 40**-**bit Shifter. * To get the basic knowledge of Pin Configuration of ADSP**-**21535. * Got knowledge of **Microsoft Visual Basic**. (Coding is done in Microsoft Visual Basic for the front end of Image Processing Development kit) * Got knowledge of chips through Datasheet or Testing **-** The main chips used were K6T4008C18 (Static RAM), 74HC393 (Counter), ADV7183 (Analog Video Decoder), RTL8139 (LAN Networking) and LM9627 (Camera). * Performed **ROM (Read Only Memory) Programming.** The information is embedded in the ROM in the form of bits by a process known as programming the ROM. Programming is used to refer to the hardware procedure which specifies the bits that are going to be inserted in the hardware configuration of the device. This is what makes a ROM a Programmable Logic Device. * .Bin File is programmed. The BIN file is a binary file that contains binary data that is used by different applications. * Performed Testing of Image Processing Development kit. (Hardware) * The main testing is performed on PCB **-** 1. (Consists of Four PCB’s) * It is a four layer board. It consist of ADSP-21535 (Processor), WED416S8030 (SDRAM), SST39VF010 (Flash), AN5015 (USB), MAX232 (Serial Comm.), JTAG Connector, 64C128 (SPI BOOT ROM), AD1885 (Codec), regulators (L4960, LM317), 3 crystals for providing clock (8MHZ, 48MHZ, 34.17MHZ) * Tested the Hardware at testing points (Where testing is done). * Three regulators (One L4960 and Two LM317) at PCB **-** 1. * Three crystals (8MHZ, 48MHZ and 34.17MHZ) at PCB **-** 1. * Serial Communication Port, USB Port, SDRAM, FLASH & JTAG Connector. |
|  |  |
| **Project** | The project was used to make the Network Camera. Coding was done in Visual Basic Language for front**-**end and DSP platform coding was done in Assembly Language and C language. Blackfin processor ADSP**-**21535 was used. |
|  |  |
| **Work Location** | Roorkee, India |

|  |  |
| --- | --- |
| **Title** | Web Page Development |
|  |  |
| **Domain** | **Web** |
|  |  |
| **Responsibilities** | * Development, Implementation and Support of websites. (Web Page make information available on the internet. HTML is the standard markup language used to creating Web Pages that are displayed on the internet. A Web Document consist of three main sections Head, Title and Body.) * Developed Web Pages using Hypertext Markup Language (HTML) code. * Used HTML Editor for creating HTML documents. (HTML Document consist of HTML elements. HTML elements (Represented by Tags) tell the Web Browser how to display the content. Web Browser reads the HTML documents and display them on the internet.) * Creation of section like Headings, Paragraphs, Links, Images and Lists. * Used six levels of headings, numbered 1 through 6, with 1 being the largest. (<H1>, <H2>, <H3>, <H4>, <H5>, <H6>) * Created Paragraphs. Used Paragraph Alignment. (ALIGN=RIGHT, ALIGN=LEFT, ALIGN=CENTER) * Creation of Hyperlink (Internet Navigation) for navigating within the Web Pages, between different Web Pages and between different websites. * Placing of Image on a Web Page in a proper format. * Displayed Image without any associated text. Adjustment of Image to Left, Right and Center. * Displayed Image aligned with text. * Used unnumbered, numbered and definition Lists. (<UL>, <OL>, <DL>). Used nested Lists. * Creation of division that contain several Paragraphs, Links, Images and Lists. * Formatting of Web Pages for display them as they are intended to look. (Formatting includes Paragraph breaks, Bold, Italics, Font faces and Sizing (Text Fonts, Text Size), Colors (Background color, Text color, Border color) and Text Alignment) * Creation of Web Forms for users to input information like text boxes for typing data and drop down menus for multiple like choice input. * Creation of Tables for arranging data in rows and columns. * Used Table attributes. (Horizontal alignment of a cell (ALIGN (LEFT, CENTER, RIGHT)), Vertical alignment of a cell (VALIGN (TOP, MIDDLE, BOTTOM)), The number (n) of columns a cell spans (COLSPAN=n), The number (n) of rows a cell spans (ROWSPAN=n), Turn off word wrapping within a cell (NOWRAP)) * Used Forced Line Breaks for line break with no extra (white) space between lines. (<BR>) * Used Horizontal Rules for producing a horizontal line. (<HR>) * Used Extended Quotations to include lengthy quotations (<BLOCKQUOTE>) * Created Background Graphics. A background can be an image of an object. (A Logo possibly) * Used Attributes in HTML source code for providing additional information about the HTML element. (Like <element attribute="value">) * Added comments in HTML source code for debugging, notifications and reminders. |
|  |  |
| **Work Location** | Delhi, India |