



COLLEGE OF ENGINEERING DESIGN ART AND TECHNOLOGY

DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING

**DEVELOPMENT OF A LOW-COST LINUX-BASED
MICROCOMPUTER EVALUATION KIT USING ELECTRONIC
WASTE**

OOLA EDWARD: 18/U/26939/PSA

**Main Supervisor: Dr Bakkabulindi Geofrey
Co Supervisor: Dr Edwin Mugume**

**A final year full project report submitted in a partial fulfillment of the
requirements for the award of the degree in Bachelor of Science in Electrical
Engineering.**

Approval

This report has been submitted for examination with approval of the project supervisors.

Main supervisor

Dr. Bakkabulindi Geofrey

Lecturer

Department of Electrical and Computer Engineering Makerere University

Signature..... Date.....

Co supervisor

Dr. Edwin Mugume

Lecturer

Department of Electrical and Computer Engineering Makerere University

Signature..... Date.....

Declaration

I OOLA EDWARD, sincerely declare that to the best of my knowledge, the information presented in this project report is an original work resulting from my sole effort and intellect. Except for extracts whose references are stated here in, it has never been published or submitted to this organization or any other institution of training for any academic reward.

Signature..... Date.....

Dedication

I dedicate this report to my family who morally, financially and spiritually supported me throughout the execution of the project. I dedicate this report to the Electrical Engineering class of 2018-2022 for all the good time we shared during the course.

Acknowledgements

I thank the Almighty God for giving me the courage, strength, wisdom, understanding, guidance and the ability to successfully complete my final year project and report writing. I extend my sincere gratitude to my project supervisors Dr. Bakkabulindi Geofrey and Dr. Edwin Mugume for the tireless effort towards guiding the direction of research and implementation of this project. Lastly, I thank the department of Electrical and Computer Engineering staff members for the guidance and knowledge they have equipped me with as it came to the implementation of the project.

Abstract

Linux-based Evaluation kit involves the use of ARM microprocessor chips with a memory management unit (MMU) to run an embedded microcomputer application such as Robotics, home security system, medical and automotive systems among others. In this project, a low cost Linux-based Evaluation kit was developed for embedded system engineers to implement and confirm the functionality of various Linux operating systems such as Debian, android and Ubuntu which could lead to a comfortable deployment.

ARM9 microprocessor was used as the main processor in the development of the kit. The schematic for the evaluation kit was developed using Ki-CAD software. The schematic was tested for errors through simulation after which a two layers PCB was designed with Ki-CAD tool in order to lower the cost of its fabrication. Length tuning, power, signal width, via hole diameter, voltage zoning were all done in order to meet the PCB fabrication requirements and to reduce noise interference on the board.

The designed PCB was tested, modelled and simulated in Ki-CAD for any design constraint violation errors before proceeding to the fabrication stage.

Gerber files were then generated and sent to the PCB manufacturer (PCBWAY China) for the final PCB fabrication.

Key words used: ARM-Microprocessor, Linux-OS, Length-tuning, PCB, fabrication, embedded-systems

List of Abbreviations

ARM - Advanced Risc Machines
SMD-Surface Mount Device
IC – Integrated Circuit
RAM-Random access memory
DDR2 – Double Data Rate 2
LQFP – Low-profile quad flat package
LBGA – Low-profile Ball grid array
OS – Operating system
PCB- Printed circuit board
GSM- Global system for mobile
MODEM-Modulation and demodulation
MMU-Memory management unit
IoT-Internet of things
SOC-System on chip

Contents

1 Chapter 1: Introduction	1
1.1 Background	1
1.2 Problem statement	2
1.3 Justification	2
1.4 Project objectives	2
1.4.1 Main objective	2
1.4.2 Specific Objectives	3
2 Chapter 2: Literature review	4
2.1 Linux-based microcomputer evaluation kit.	4
2.2 Beagle bone black open source project	4
2.2.1 Schematic Design	5
2.2.2 BOOTING SEQUENCE CONFIGURATION	8
2.2.3 Multi-layer PCB settings	8
2.2.4 Results	9
2.2.5 Drawbacks of the Beagle bone black board	10
2.3 Design and development of AT91SAM912N evaluation kit.	10
2.3.1 Schematic Design	11
2.3.2 PCB design	12
2.3.3 Results	13
2.3.4 Limitations of AT91SAM912N Evaluation kit project.	14
2.4 Olimex A13 evaluation board.	14
2.4.1 Schematic design	16
2.4.2 PCB Design	18
2.4.3 Drawbacks of the project	19
3 Chapter 3	20
3.1 Methodology	20
3.2 Schematic design	20
3.3 PCB design	23

3.4	PCB fabrication and assembly	29
3.5	PCB ASSEMBLY	31
3.6	LIST OF COMPONENTS USED FOR THE ASSEMBLY	33
3.7	CREATION OF A BOOTABLE LINUX SDCARD	34
4	Chapter 4	37
4.1	Results and discussion	37
4.2	Full systems simulation test of the schematic design	37
4.3	Testing the assembled evaluation board	38
4.4	DISCUSSION OF RESULTS	39
4.5	Deployment of the prototype	40
4.5.1	Wireless Router	40
4.5.2	Point of sale Machine	41
4.5.3	Terrestrial set top box	41
5	Chapter 5	42
5.1	Conclusion, Recommendations and Challenges	42
5.2	Conclusion	42
5.3	Challenges	42
5.4	Recommendations	43

List of Figures

1	ARM CORTEX A-8 Linux Board Block Diagram [9]	5
2	Booting sequence schematic [16]	8
3	Net class DRC Settings	9
4	Beagle bone black PCB Schematic	9
5	Schematic Design	12
6	4-Layers PCB Design	13
7	A13 Block diagram [12]	14
8	A13 Schematic [12]	17
9	A13 4-layers PCB	18
10	Methology Flowchart	20
11	ARM9 Block diagram [18]	21
12	ARM9 Schematic	22
13	ARM9 Ki-CAD PCB design	25
14	Copper Pours	26
15	3-D View of the schematic	27
16	Proteus schematic	28
17	Gerber files views	29
18	PCB Top view	30
19	PCB bottom view	31
20	Assembled PCB	32
21	Table of components	34
22	Putty serial interface	36
23	Full system simulation with Proteus	37
24	ARM9 booting ARCH LINUX	38
25	Arch Linux booting from the board	39
26	Application of the evaluation board	41

1 Chapter 1: Introduction

1.1 Background

Embedded systems used in the fields of Robotics, IoT, embedded systems, telecom and medical equipment are operating-systems-based applications hosted on microprocessor based systems [1]. Such systems are associated with high processor speed, high memory capacity and high storage capacity which are not available in a typical microcontroller.

The hardware and software prototype design and development of such system are done using an operating system based microcomputer evaluation kit before going into their actual implementation.

Linux based microcomputer evaluation kit is a complete system-on-chip built around the ARM microprocessor[1] [2] [3]. ARM (Advanced RISC Machines) is a family of CPU based on RISC (Reduced Instruction Set Computer) architectures. They exist as a 32 bit or 64 bit CPUs.

ARM microprocessors from ARM9 to the current ARM Cortex A-series are built with memory management unit (MMU) which enables them to have operating systems hosting capabilities. Various embedded operating systems such as Linux, Symbian and windows are hosted on such processors. MMU is used in linking the virtual memory address from the operating system software to the physical addresses of the hardware. This enables them to perform automatic memory map creation required for CPU address decoding [2][3][4].

The common microcomputer evaluation kit available on the market includes Raspberry Pi, Beagle bone black, Olimex kits among others. These evaluation kits are mainly ARM Cortex A series, 32 bits or 64 bits microprocessors systems with high clock speed of 1 GHz and high memory capacities[5][2][6].

They run various versions of Linux such as Debian Linux, Ubuntu and Android. Linux OS contains most of the firmware and device drivers for the System on Chip (SOC) hardware interfaces [4] [3].

However, these kits are not readily available on the local market and must be imported expensively from online platforms such as Amazon and Alibaba.

Through careful design of the hardware kit and PCB fabrication of the schematic, a lower cost evaluation kits can be developed in the country using components from old smart phones.

1.2 Problem statement

The high cost of the current microprocessor based evaluation kits has slowed down hardware prototyping of various embedded systems innovation projects. Hence, there is a need to provide an alternative to these expensive evaluation test kits.

1.3 Justification

The operating system firmware and device drivers for embedded electronic hardware cannot be tested and evaluated on a microcontroller based kits such as Arduino board since they lack memory management unit (MMU) needed to run an operating system [4].

Such firmware can only be evaluated on a microprocessor-based evaluation kits such as Raspberry pi, Beagle bone, Olimex kits among others.

The average cost of a microprocessor-based evaluation kit is 200 US dollars from the online stores such as Alibaba and Amazon [6]. This makes it unaffordable to most student developers in the country coupled with their unavailability on the local market.

By developing a customized evaluation kit with components extracted mainly from old phones, a cheaper version of the hardware test kit can be made to address the above problem.

This will also help in the recycling of the electronic waste which would have otherwise been disposed into the environment causing environmental degradation.

1.4 Project objectives

1.4.1 Main objective

To develop a microprocessor based evaluation kit cheaply for embedded application testing.

1.4.2 Specific Objectives

- To develop hardware evaluation kit schematic targeting primarily Linux operating system firmware and device drivers testing.
- To model and simulate a 4-layered PCB of the circuit for hardware PCB fabrication.
- To fabricate and assemble a Printed Circuit Board with components mainly from old smart phone and laptop motherboards.
- To evaluate the performance of the designed kit.

2 Chapter 2: Literature review

2.1 Linux-based microcomputer evaluation kit.

These types of board comprise of ARM based microprocessor with memory management unit (MMU).

ARM (Advanced RISC Machines) is a family of CPU based on RISC (Reduced Instruction Set Computer) architectures. They exist as a 32 bit or 64 bit CPUs.

To be able to run a Linux operating system, the ARM microprocessor must have MMU (Memory Management Unit) in order to link the virtual memory address from the software to the physical addresses of the hardware [7] [8].

They include ARM9 and ARM Cortex-A series microprocessor with speed ranging from 454 MHZ up to 1 GHZ.

The design and development of a microcomputer evaluation kit capable of running Linux operating systems had been carried out by various open source embedded hardware developers such as Texas instruments with Beagle bone black, Olimex with All-winner A13 board, Broadcom with Raspberry pi series among others [6] [5] [9]. Some of the related project designs are discussed below.

2.2 Beagle bone black open source project

This project is one of the Texas Instruments open source project which has been adopted by the Beagle bone community [9] [3] [10]. The design is based on ARM-Cortex A8 microprocessor interfacing peripheral interfaces such as:-

- 512MB DDR3 memory, 4GB NAND flash storage, Micro SD storage
- USB Hosts and USB clients
- Power IC, 24MHz Clock
- 10/100 Ethernet LAN, HDMI interface, LCD interface,
- UART and SPI interface as illustrated in the block diagram below.

Beagle Bone Black Hardware Architecture

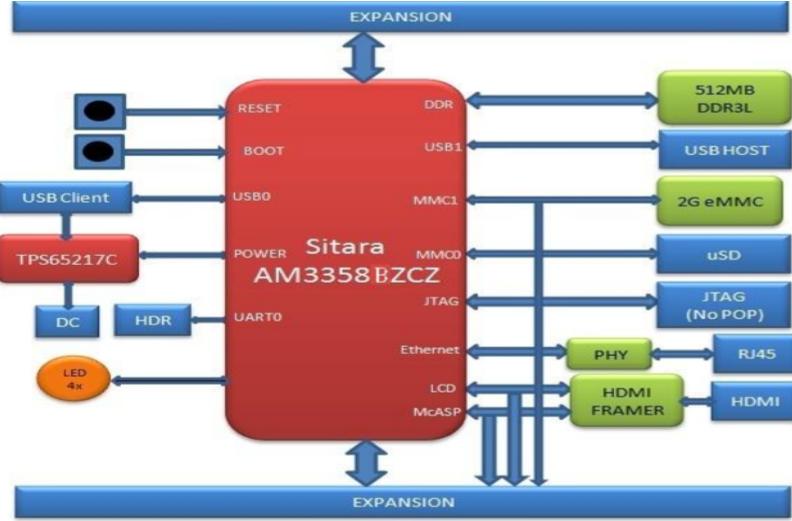


Figure 1: ARM CORTEX A-8 Linux Board Block Diagram [9]

2.2.1 Schematic Design

The schematic design of the above architecture is comprised of a 512MB of DDR3 memory, 4GB NAND flash storage, USB hosts, LAN, HDMI, 24Hz oscillator clock, power IC, LCD, GPIO, SPI, UART, microSD connector [9] [6] [11].

The TPS65217C power IC from Texas Instruments provides the +3.3V, +1.8V, 1.5V, 5V, 1.35V voltage outputs needed to power the CPU, RAM, clock, NAND flash, microSD and other peripheral devices [2].

The schematic design addressed the key areas such as the power bus, address bus, data bus and control bus interface between the ARM processor and the external peripheral interfaces. The schematic of some of the key areas in the design are illustrated below.

RAM Memory Device

ARM CORTEX A-8 processor supports the standard DDR3 and DDR3L x16 devices. The DDR3 devices work at 1.5V and the DDR3L devices can work down to 1.35V to achieve lower power. The DDR3L comes in a 96-BALL FBGA package with 0.8 mil pitch. DDR3L is the lower power device working at 1.5V or 1.35V. The standard operating frequency of the DDR3L is 400MHZ. The DDR3L chip has 16 data bus lines and 16 address bus lines used for memory addressing to the MPU buses as illustrated below [9] [6] [12] [2].

DDR3L Memory Addressing

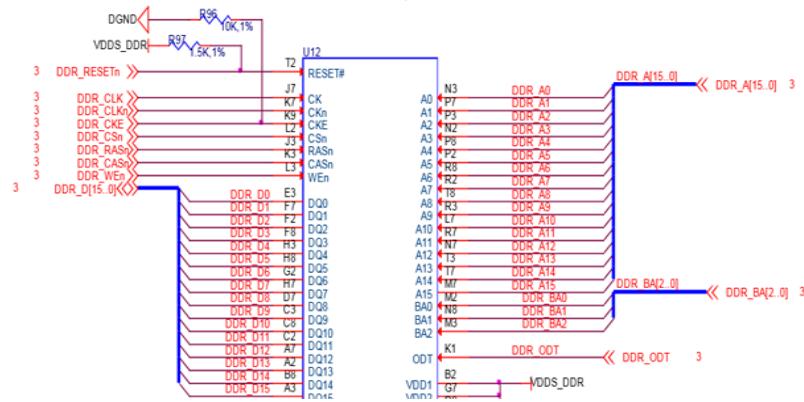


Fig 4, DDR3L Memory Interfacing Circuit [9] [6] [7] [13]

eMMC Memory Interface

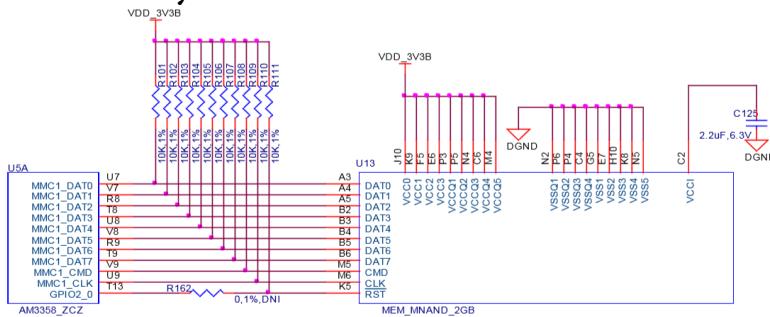
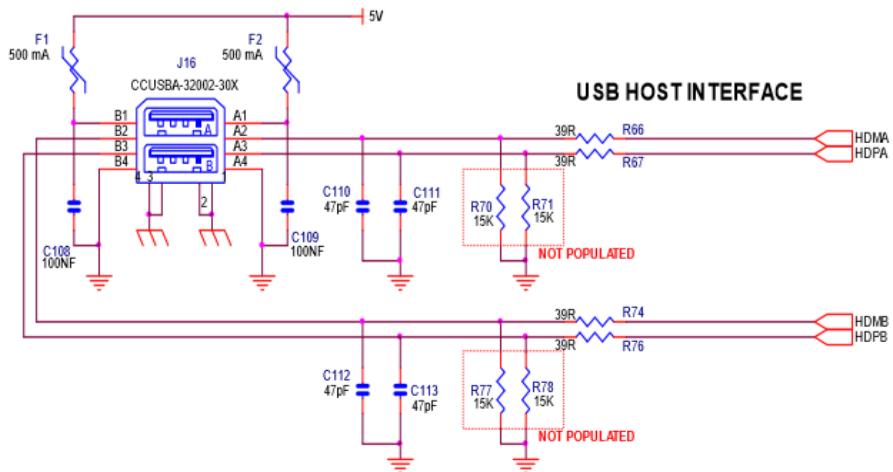


Fig 5, NAND Flash Memory Circuit schematic [9] [6]

USB Host Interface



Micro SD storage

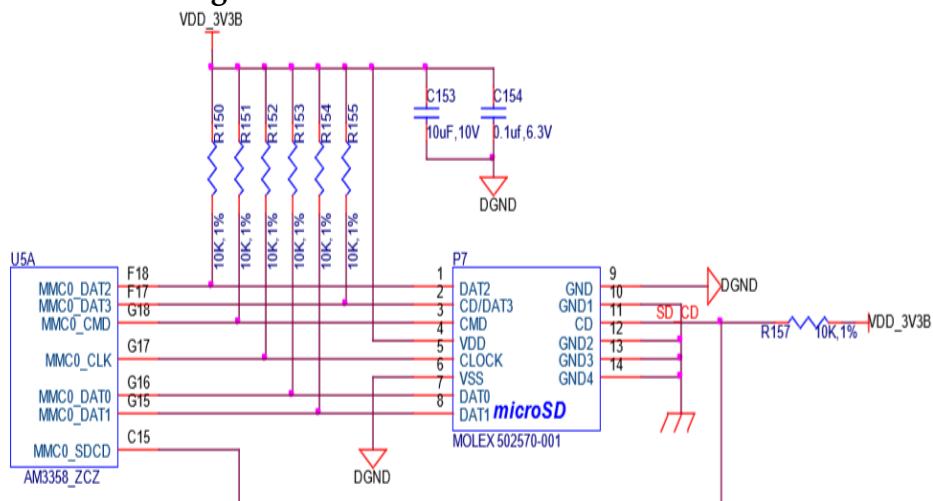


Fig 6, USB HOST, microSD schematic [6] [2] [5]

2.2.2 BOOTING SEQUENCE CONFIGURATION

The booting sequence of the board is set using SYSBOOT [4..0] address line from the MPU_LCD data bus address [14] [15]. For example SYSBOOT [0..4] address 10111b gives the booting sequence of MMC, SPI, UART, and USB. The above digital bits are implemented using pull-up and pull-down resistors to provide a volt of 3.3V or 0V for each respective LCD data address lines indicated above. The implementation of a 16 bit LCD data address 0100000010111b for the booting sequence order of MMC, SPI, UART and USB is illustrated with a schematics below where DNI indicates not connected terminal. SYSBOOT [15..14] is used to select the clock frequency of the oscillator. The left most 01b bits indicates oscillator clock frequency of 24MHz [14] [15]. Various Boot sequence can be implemented by selecting the appropriate SYSBOOT address.

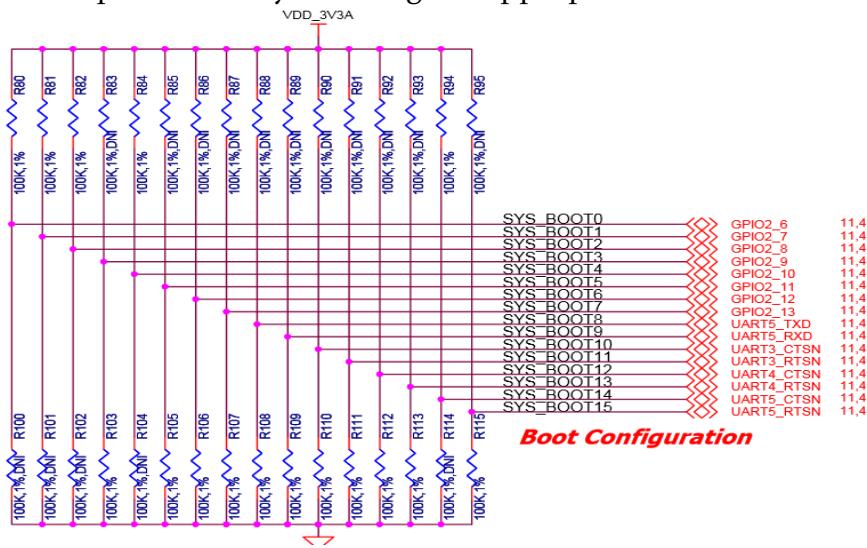


Figure 2: Booting sequence schematic [16]

2.2.3 Multi-layer PCB settings

A six layered PCB of the Beagle bone evaluation board designed by Beagle bone community applied the following PCB design parameters as illustrated in the table below.

NET CLASS DRC SETTINGS

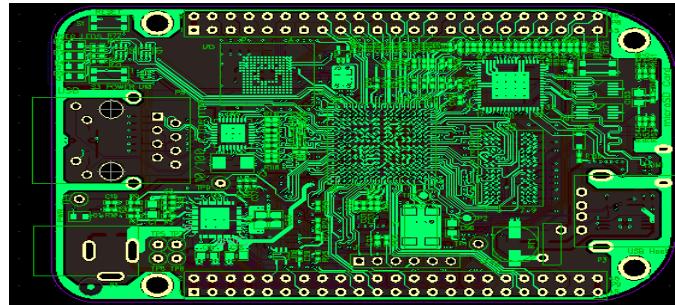
Name	Clear-ance	Track width	Via Size	Via Drill	uVia size	uVia Drill	dPair Widrh	dPair Gap
Default	4.76 mils	4.76 mils	23.62 mils	11.81 mils	11.81 mils	3.937 mils	7.87 mils	9.84 mils
POWER	4.76 mils	25 mils	39.37 mils	23.62 mils	11.81 mils	3.937 mils	7.87 mils	9.84 mils

Figure 3: Net class DRC Settings

A 6 layered PCB with Diptrace PCB design software has the top layer set for signal, layer2 for ground power, layer3 for signal, layer4 signal, layer5 power and layer 6 for signal. Power plane zones for 1.8V, 3.3V, 1.35V, 5V are in layer 5 of the power plane [17].

2.2.4 Results

The result of the routed footprints gave an error free 6-layered PCB of the Beagle bone black using Diptrace PCB software as shown below [17].



Source: Beagle Bone Black PCB sample design from Dip-trace 6.1 PCB Suite [17]

Figure 4: Beagle bone black PCB Schematic

2.2.5 Drawbacks of the Beagle bone black board

- Six layers PCB is very expensive and difficult to design.
- Trace width and clearance too small to be produced by most PCB fabricators.
- Six layers PCB takes a long time to fabricate compared to a double layer PCB.
- Since the processor and the RAM are LFBGA types, they are very difficult to solder.

Related Work

2.3 Design and development of AT91SAM912N evaluation kit.

In this paper, a project for the Evaluation kit was designed for embedded system engineer to implement and confirm the functionality of their operating systems which could lead to a comfortable deployment. The independent modules for the interfaces of the **AT91SAM912N** board have been designed and the schematics have been developed using Ki-Cad.

AT91SAM12CN IC from Atmel Corporation supports deterministic, real-time operation, offers supervisory functions, and has third-party real time OS (RTOS) support comparable to those of 8-bit controllers [1] [4] [10] [11].

Developed for highly-connected image-processing applications such as point-of-sale terminals, Ethernet-based IP cameras, and bar code readers, AT91SAM12CN integrates a 200 MIPS ARM926EJ-S core with: a camera interface; seven USARTs; 10/100 Ethernet MAC; 12 Mbps USB device and host controller with on-chip transceivers; external bus interface supporting SDRAM, Flash, NAND Flash with built-in ECC, SD, SDIO, and Multimedia Card interface (MMC); three synchronous serial controllers (SSC); two master/slave Serial Peripheral Interfaces (SPI); a three-channel 16-bit timer/counter; two-wire interface (TWI); and IEEE 1149.1 JTAG Boundary Scan on all digital pins among others [4] [1] [6] [13].

2.3.1 Schematic Design

The board schematic was designed to provide the following interfaces

- 64 MB SDRAM, 256 MB of NAND Flash Memory
- 1 USB Host and 1 USB Device Port Interface, 1 RS232 Serial communication Port
- 1 PHY Ethernet 100 base TX with three status LEDs
- 1 Data-Flash, SD/MMC Card Slot 1 External Bus Interface (EBI) signals routed through 96 Pin Euro Connector, 1 Lithium Coin Cell Battery Retainer for 12mm Cell
- JTAG /ICE Debug Interface

The DC voltage input of the board included +5V, +3.3V, +1.8V, 1.1V required to power the ARM9, DDR2 memory, NAND flash, microSD card and other peripherals [11].

Schematic design

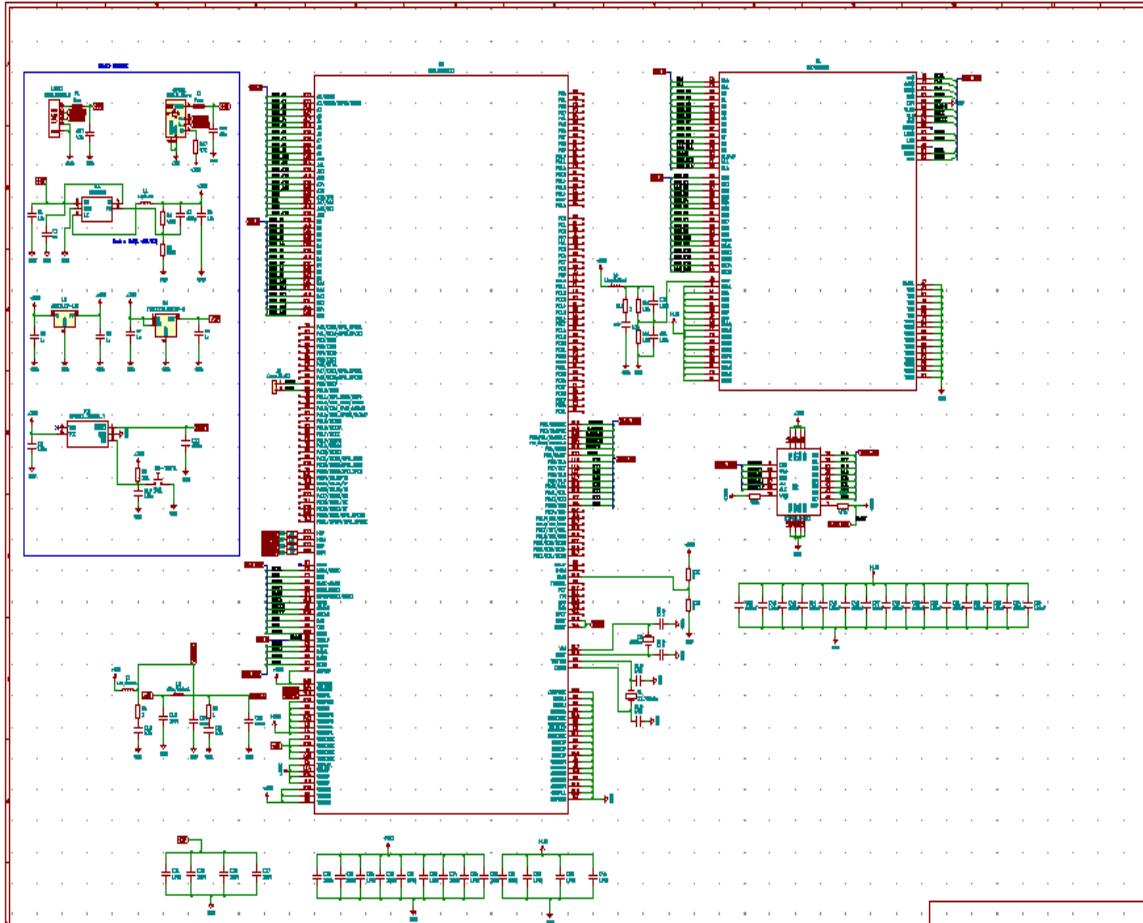


Figure 5: Schematic Design

2.3.2 PCB design

From the above project, the implemented four layered-PCB was designed with the top layer for signal, inner layer2 for signal, inner layer3 for signal and bottom layer for power plane as indicated in the Ki-Cad template below.

The PCB was design to give 50 ohm impedance on the copper layer with a board thickness Of 1.6mm to meet the fabrication requirements of most manufacturer.

Power plane zones were created on the top copper layer to solve the problem of multiple voltage levels of +5V, +3.3V, +1.8V and 1.1V [4] [1].

The trace width of 5mils, trace clearance of 5mils and VIA size settings were made in the Ki-Cad design rule as shown in the template below. The power lines +3.3V, +5V, +1.8V, +1.1V were given a thicker trace width of 11.81102 mils to

facilitate faster conduction of large amount of current through the copper layer to the ground to avoid too much heating [4].

Net Classes				
Name	Clearance	Track Width	Via Size	Via Drill
Default	5 mils	5 mils	18 mils	10 mils
50ohm_in_microstrip	5 mils	11.81102 mils	18 mils	10 mils

2.3.3 Results

After routing and simulating, an error free 4 layered PCB was obtained with Ki-cad PCB software as shown below [4] [1].

4-layers PCB design

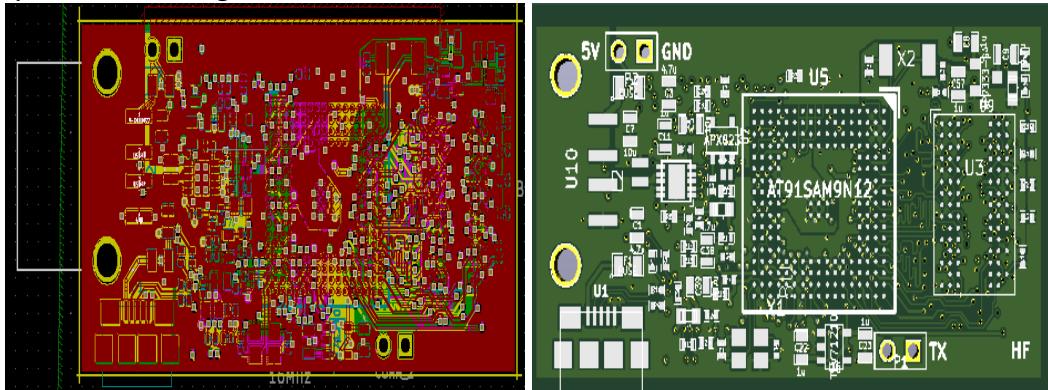


Figure 6: 4-Layers PCB Design

2.3.4 Limitations of AT91SAM912N Evaluation kit project.

- Low processor speed compared to beagle bone black
- Being a 4-layer PCB, it is expensive to fabricate compared to a double layer PCB
- Longer fabrication time required compared to a two layered board.
- Unable to run android and Arch Linux operating systems.
- Since both RAM IC and processor are of LQBGA type, they are very difficult to solder.

Related Work

2.4 Olimex A13 evaluation board.

This is an open source hardware project built around ARM CORTEX-A8 by Olimex. A13 is an ARM CORTEX-A8 with LQFP pin configuration making it easier to solder on the PCB compared to LBGA ball type [12].

The A13 board was built around the hardware architecture shown in the figure below.

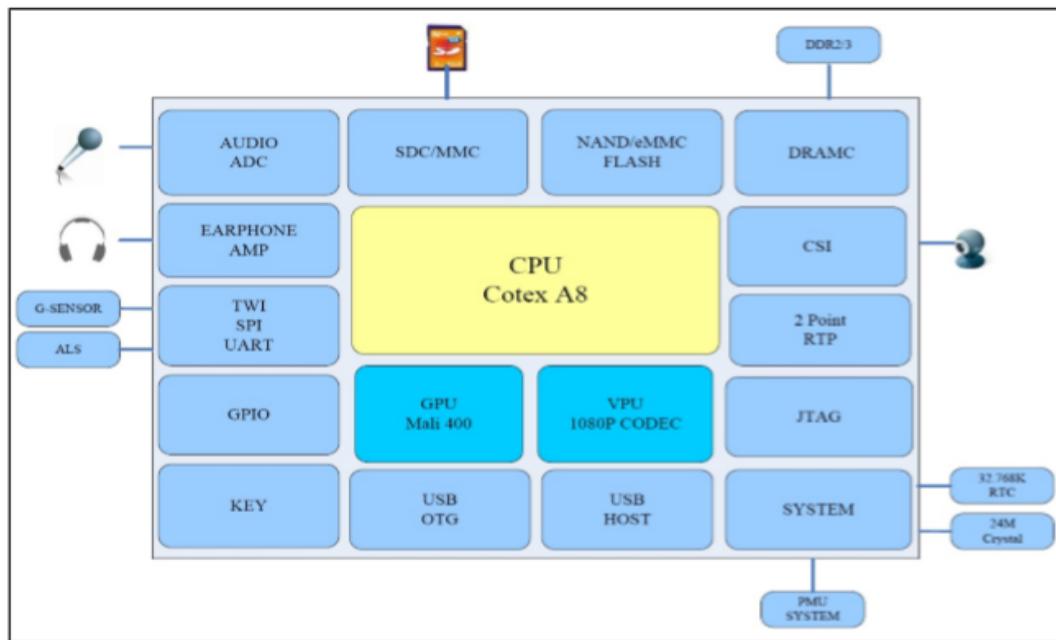


Figure 7: A13 Block diagram [12]

The board has the following set of features:

- A13 Cortex A8 processor at 1GHz, 3D Mali400 GPU
- 512 MB DDR3 RAM
- 6-16VDC input power supply, noise immune design
- 3 + 1 USB Host, 3 available for users 1 for (only in A13-OLinuXino-WIFI)
WIFI RTL8188CU 802.11n 150Mbit module on board
- 1 USB OTG which can power the board
- SD-card connector for booting the Linux image
- (only in A13-OLinuXino-WIFI) 8GB NAND flash (older revisions had 4GB
NAND flash)
- VGA video output – 800×600 resolution
- LCD signals available on connector so you still can use LCD if you disable
VGA/HDMI
- Audio Output
- Microphone input
- RTC PCF8536 on board for real time clock and alarms
- 5 Keys on board for android navigation
- UEXT connector for connecting additional UEXT modules like Zigbee, Blue-
tooth, Relays, etc.
- GPIO connector with 68/74 pins and these signals: 17 for adding NAND
flash; 22 for connecting LCDs; 20+4 including 8 GPIOs which can be in-
put, output, interrupt sources; 3x I2C; 2x UARTs; SDIO2 for connection SD
cards and modules; 5 system pins: +5V, +3.3V, GND, RESET, NMI

2.4.1 Schematic design

The designed schematic consists of the following key components among which include:-

The clocks

24 MHz quartz crystal Q1 is connected to pins 91 and 92 of the A13 processor.

12 MHz quartz crystal Q2 is connected to pins 6 and 7 of the GL850G (the USB controller).

32 768 kHz (RTC clock) quartz crystal Q3 is connected to pins 1 and 2 of the RTC_MODULE (PCF8563T)

The power supply circuit

The power supply is handled mainly by AXP209 power management system, an All-winner voltage regulator chip that goes together with the A13 processor. The power supply circuit of A13-OLinuXino allows flexible input supply from 6V to 16V. The minimum amperage suggested is 1A but this threshold can rise if all the three USB-HOSTs, GPIOs and LCD are on.

The board can also be powered by 3.7V Li-Po battery or from a USB power supply.

DDR3 SD RAM

2×2GB (512M x 8 bit) DDR3 SDRAM: SAMSUNG K4B2G0846Q SD RAM was used in the schematic design as the main working memory. The complete Ki-Cad schematic is shown in the figure below.

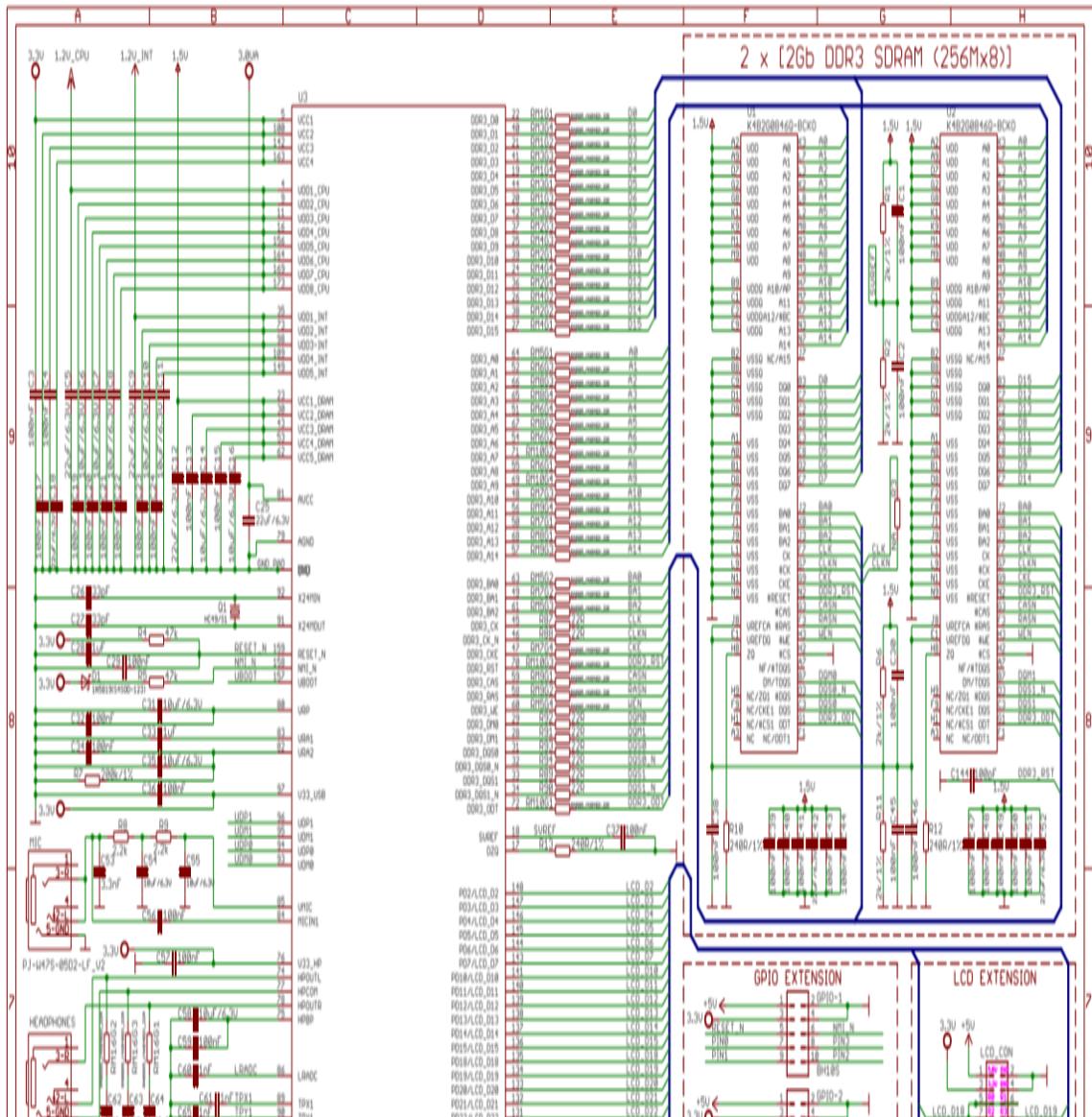


Figure 8: A13 Schematic [12]

2.4.2 PCB Design

The PCB designed in K-Cad is a 4 layered PCB with the net class settings as shown in the screen shot below.

Net Classes					
Name	Clearance	Track Width	Via Size	Via Drill	μ Via Size
Default	5.200787 mils	9.84252 mils	31.49606 mils	15.74803 mils	11.81102 mils

The minimum track clearance and width has been set to 5.2mils, VIA size and drill set to 31.49, 15.74mils respectively.

The four layers power zones were set in the order, Ground, Ground, VCC and Ground.

The power of 3.3V, 1.5V, 5V, 1.03V were partitioned on the different planes to reduce on the radio interference on the board. The created PCB is shown in the figure below.

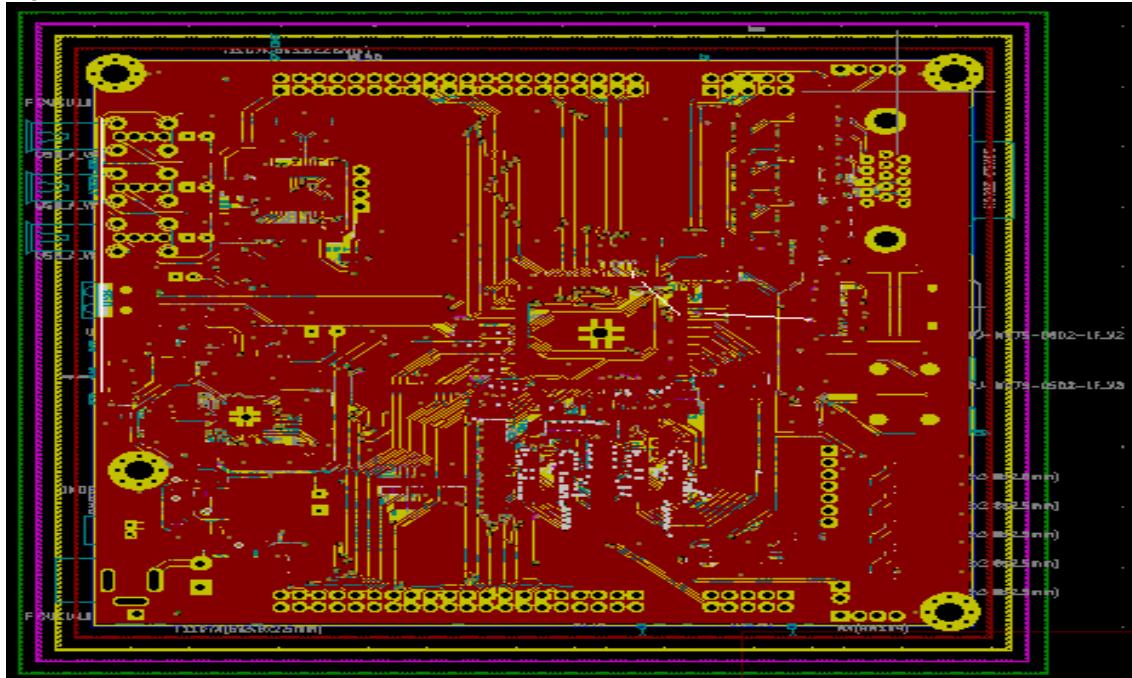


Figure 9: A13 4-layers PCB

2.4.3 Drawbacks of the project

- The 4 –layers board is difficult to fabricate and takes longer fabrication time
- The PCB is more expensive to produce compared to a two layer PCB
- The above processor is only limited to Debian and arch Linux operating system.
- It does not work properly with android OS.
- Since the DDR3 RAM used is a LBGA, the board is difficult to assemble

3 Chapter 3

3.1 Methodology

This chapter describes in detail the entire processing of developing a Linux-based microcomputer evaluation kit from schematic design, PCB design, systems modelling and simulation, PCB fabrication/assembly up to a complete hardware testing and evaluation. The process involves the use of different tools and software such as KI-Cad, Proteus and Diptrace among others. The methodology is described using the figure below.

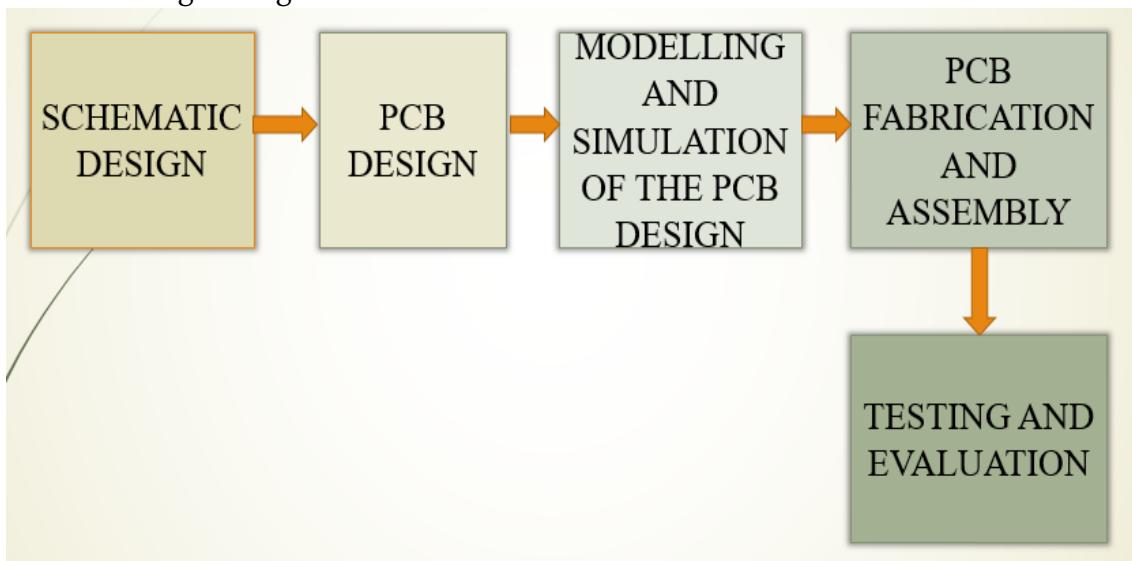


Figure 10: **Methodology Flowchart**

3.2 Schematic design

The schematic design was made in reference to the hardware architecture of NXP free scale IC MCIMX233CAG4C, which a 32bit ARM9 microprocessor with MMU having a speed of 454MHZ decent enough to run embedded Linux operating system such as Debian, arch Linux and android OS [5].

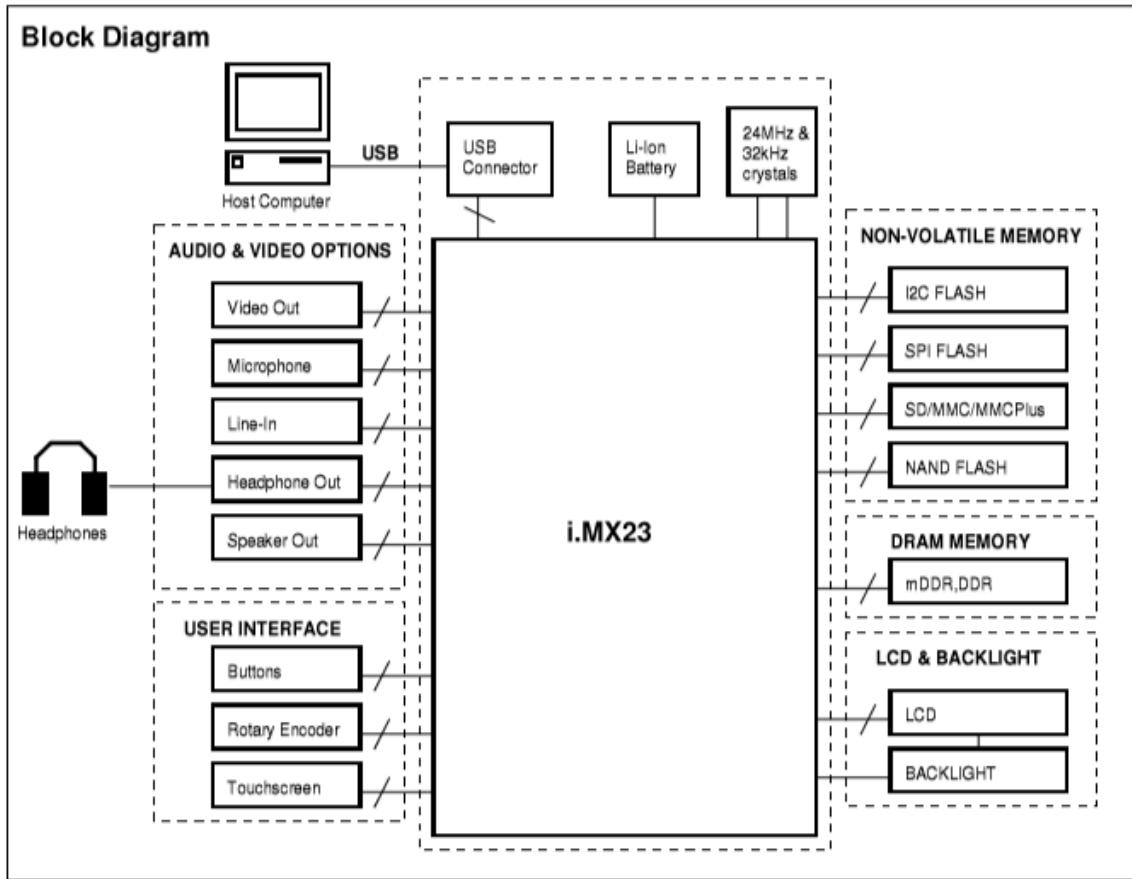


Figure 11: **ARM9 Block diagram [18]**

The processor is LQFP type IC with inbuilt voltage regulators having output of 2.5V, 3.3V, 1.02V, 1.5V to power the processor, main memory and other peripheral devices. The main memory is 64MB DDR2 memory of TSOP66 family with input voltage of 2.5V. Using Ki-Cad design software a schematic implementing the above computer architecture was made with reference to the datasheet of i.MX23 IC from NXP as indicated in the figure below [10] [5].

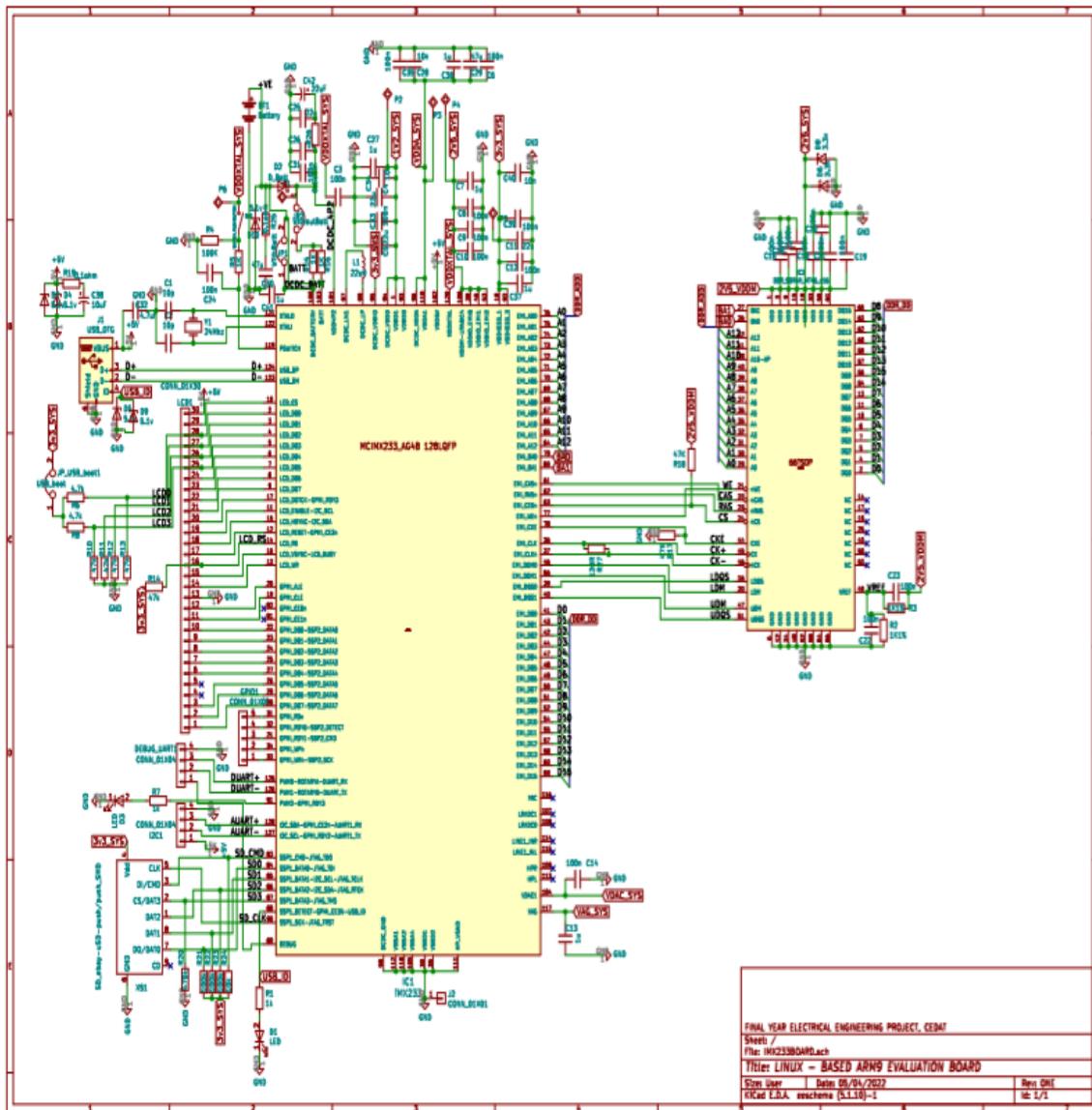


Figure 12: ARM9 Schematic

SD CARD adapter was used as an external storage to host the Linux operating system. A micro USB device port was included to provide the 5V D.C voltage required to operate the overall microcomputer system.

3.3 PCB design

Using Ki-Cad design software, the following design rule constraints were set up to guide in the design and fabrication of a two layer PCB for the above schematic based on the PCB manufacturer fabrication capability.

Name	Clearance	Track width	Via Size	Via Drill	uVia size	uVia Drill	dPair Width	dPair Gap
Default	6mils	6mils	23.62 mils	11.81 mils	11.81 mils	3.937 mils	7.87 mils	9.84 mils
POWER	6mils	25mils	39.37 mils	23.62 mils	11.81 mils	3.937 mils	7.87 mils	9.84 mils

The track width of the power lines were made much thicker compared to the signal tracks in order to conduct the high DC current quickly to the ground to avoid heat raising the temperature of the board to an abnormal level during its operation.

From Ki-Cad PCB suite, the board outline was drawn and footprints were placed inside the board outline in the best possible order for easy routing.

Both manual and automatic routing were carried out until all the footprints on the board were connected within the design rule constraints outlined above.

The designed PCB was then simulated to check for DRC and ERC errors and it returned zero errors in both cases.

Both the bottom layer and the top layer were created as ground planes. The board operating at a frequency of less than 1 GHZ implies it will not be significantly affected by interference due to the system voltages of 3.3V, 2.5V, 1.02V and 5V all existing on the same plane. Therefore the power planes for the above voltage levels were not created.

Length tuning of the data, address and clock lines between the processor and RAM were made to ensure that signals are in sync with the clock signal. This ensures that the boot loader and the operating systems do not crash due to loss of synchronism between signals and the clock.

The following length tuning data were used as shown below.

- D0…D7, LDQS, LDM – 40.2mm tuning
- CLK, CLKN -32.2mm tuning
- D8…D15, UDQS, UDM -25.5mm tuning
- Address: 25mm tuning
- CKE -20.2mm tuning
- SD: 24.8mm tuning

The designed PCB and the 3-D view of the board are shown in the figures below.

PCB showing the routed power and signal tracks

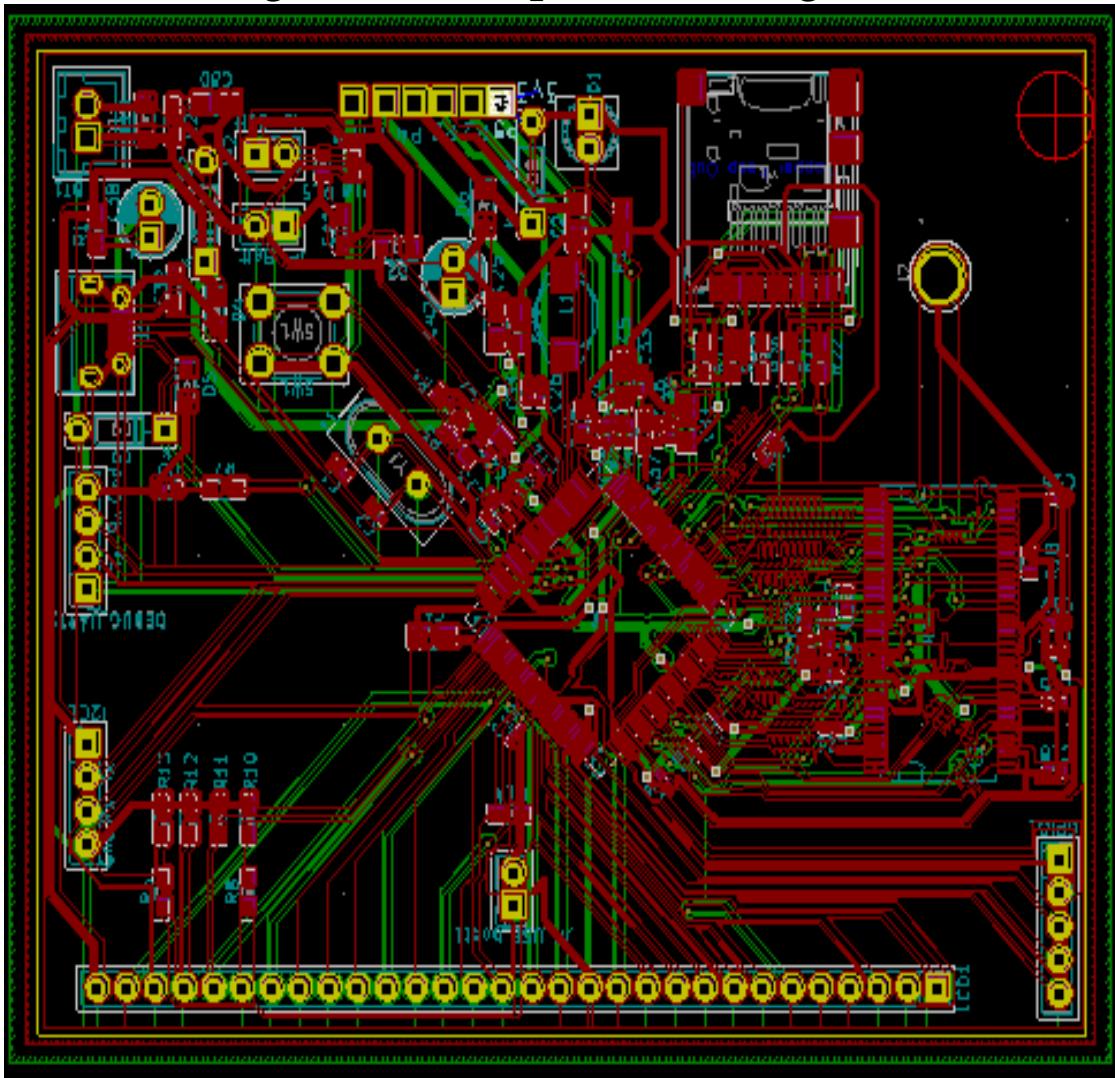


Figure 13: ARM9 Ki-CAD PCB design

Copper pours on the top plane and the Ground plane

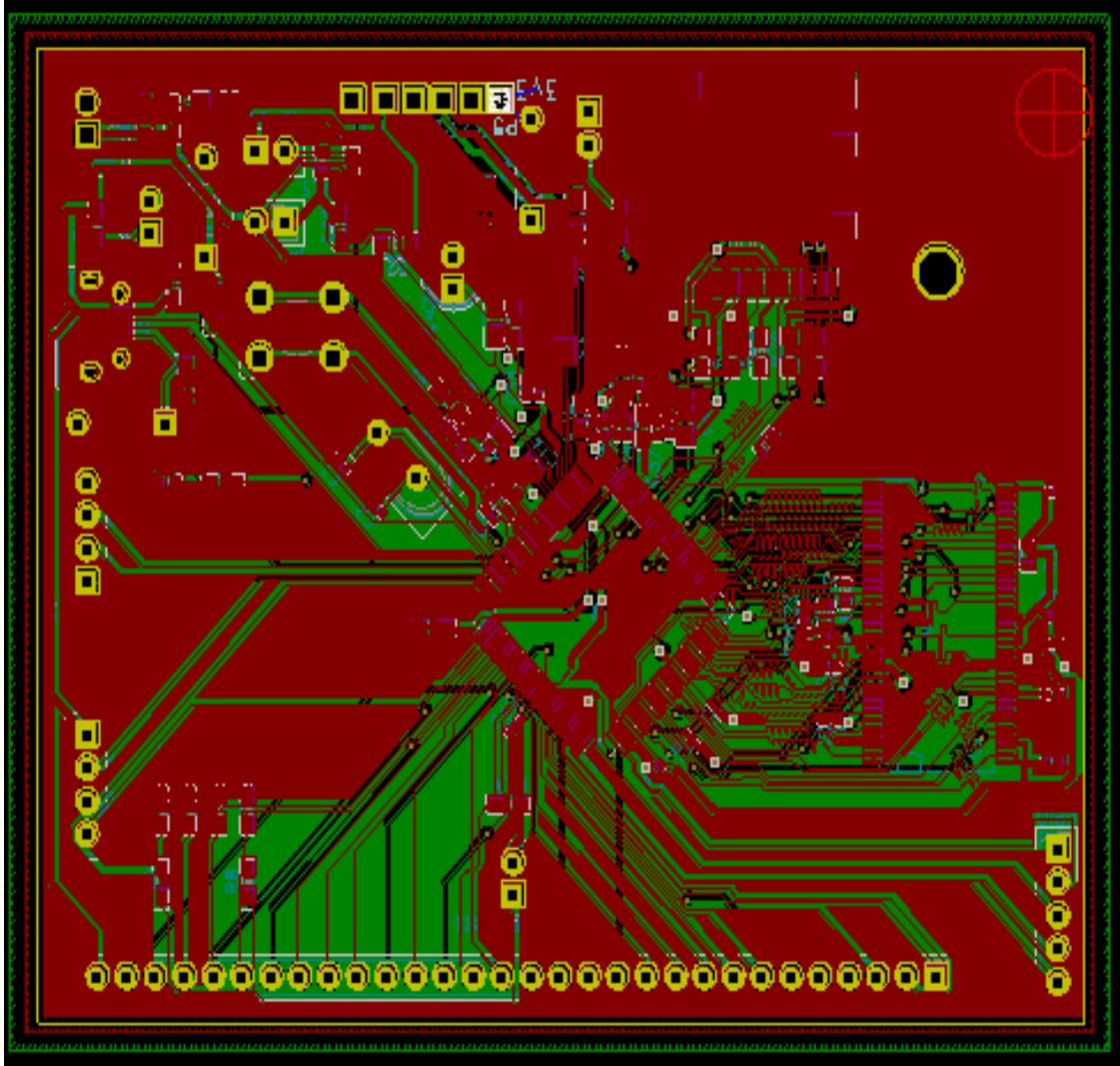


Figure 14: Copper Pours

The power zones has been designed in such a way that the top layer was made the ground plane, and the bottom layer also a ground plane. The VCC, 5V, 2.5V power planes were automatically created when the copper pours were made as shown in the above figure.

3-D SIMULATION MODEL OF THE PCB

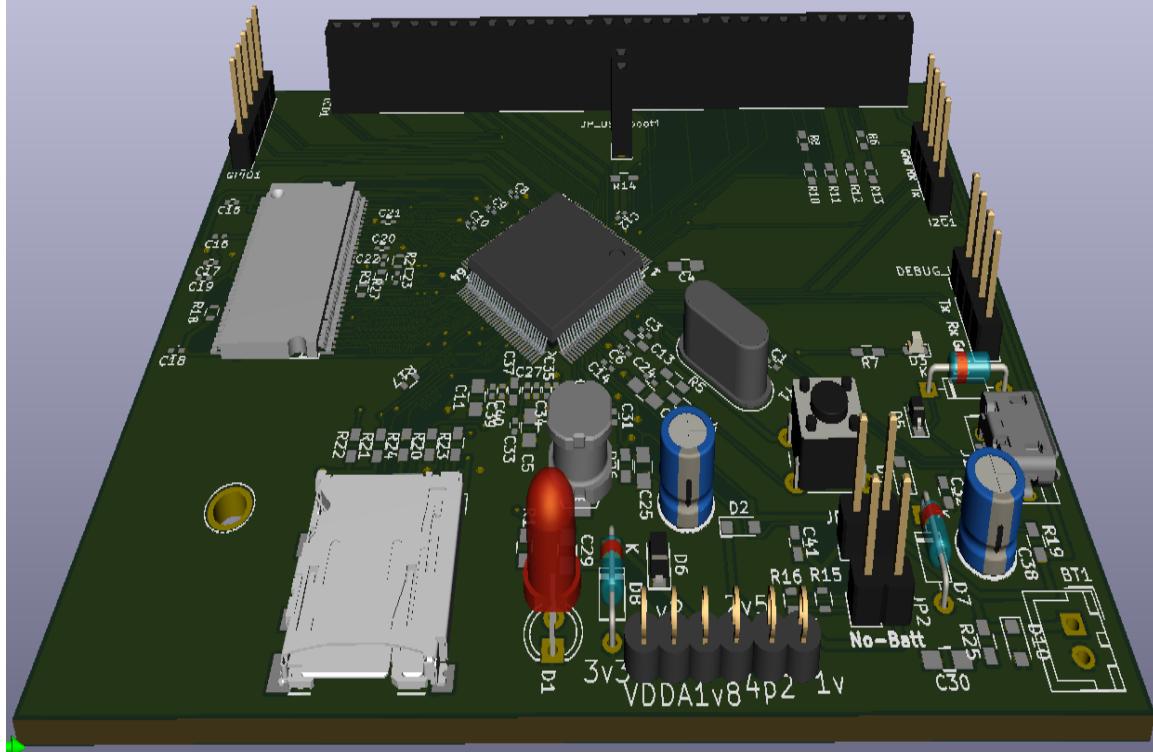


Figure 15: 3-D View of the schematic

The above CAD was derived from the Ki-Cad PCB, from the view menu. The 3-D components of the various footprints were imported from Proteus design suite to give a complete 3-D view shown above.

MODELLING AND SIMULATION OF THE SCHEMATIC

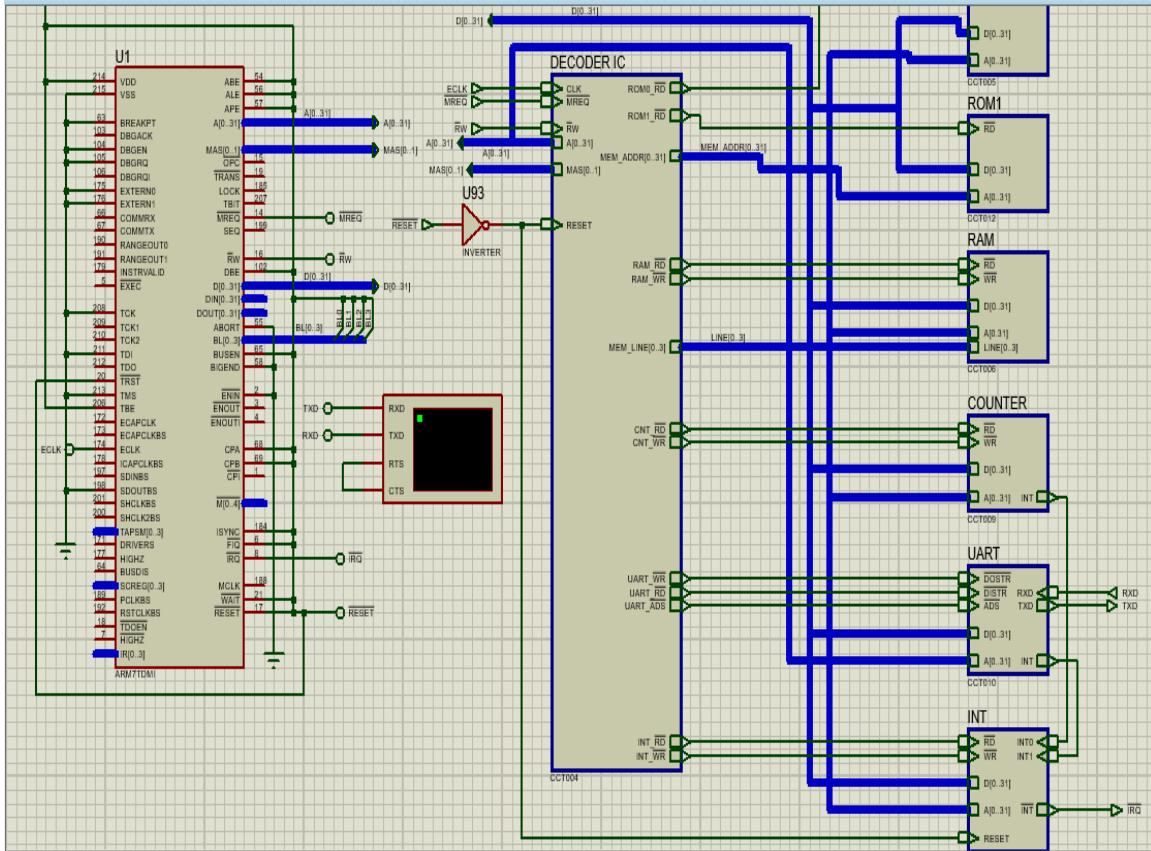


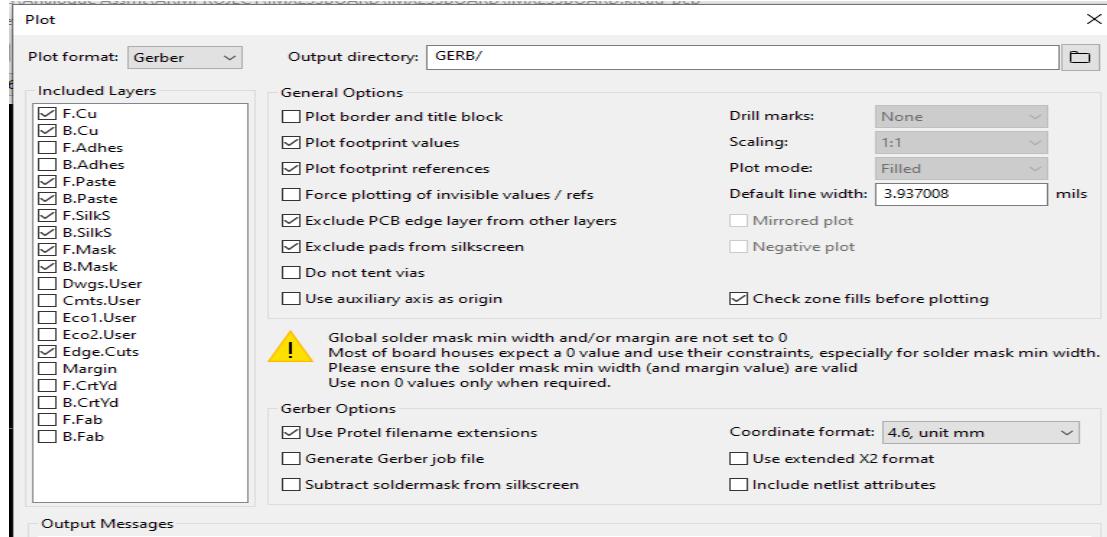
Figure 16: Proteus schematic

Using a 32 bit ARM7 TDMI microprocessor from Proteus library, a schematic of a microcomputer system comprising of the main memory (RAM), external storage memory (ROM), control signal generator, UART and INTERRUPT peripheral interfaces was designed as shown above. The decoder logic peripheral generated the memory map since ARM7 does not have the memory management unit (MMU) required for running an operating system. ROM0 was loaded with Amboot boot loader and uCLinux kernel, rootfs file system for Linux were loaded into ARM7 microprocessor at a clock speed of 8MHZ and system then simulated to boot the Linux kernel.

The results of the simulation is displayed in chapter 5.

3.4 PCB fabrication and assembly

From the Ki-Cad PCB, both the Gerber files and the drill files were generated from the plot menu by selecting top, bottom, F.Paste, B.Paste, B.silks, F.silks, B.mask, F.mask and Edge.cut as shown in the screen shot below.



The generated Gerber and drill files were viewed using Ki-Cad Gerber viewer before packaging them in zip format to send to the PCB fabricator as shown in the figure below.

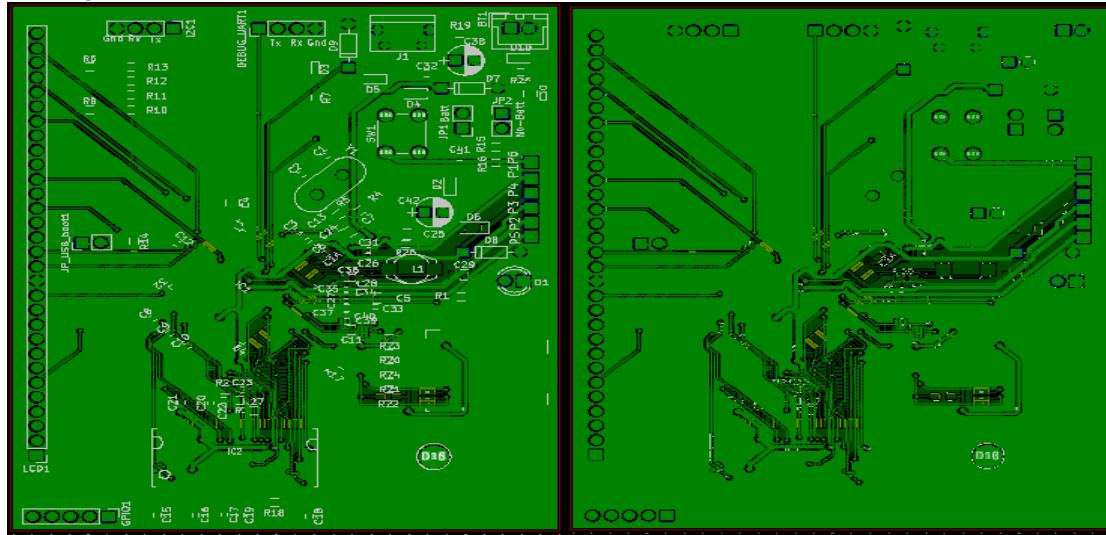


Figure 17: Gerber files views

The Gerber files were then sent to PCBWAY, a manufacturing company in China for PCB fabrication and the fabricated PCB is shown in the figure below.

Top view of the fabricated PCB

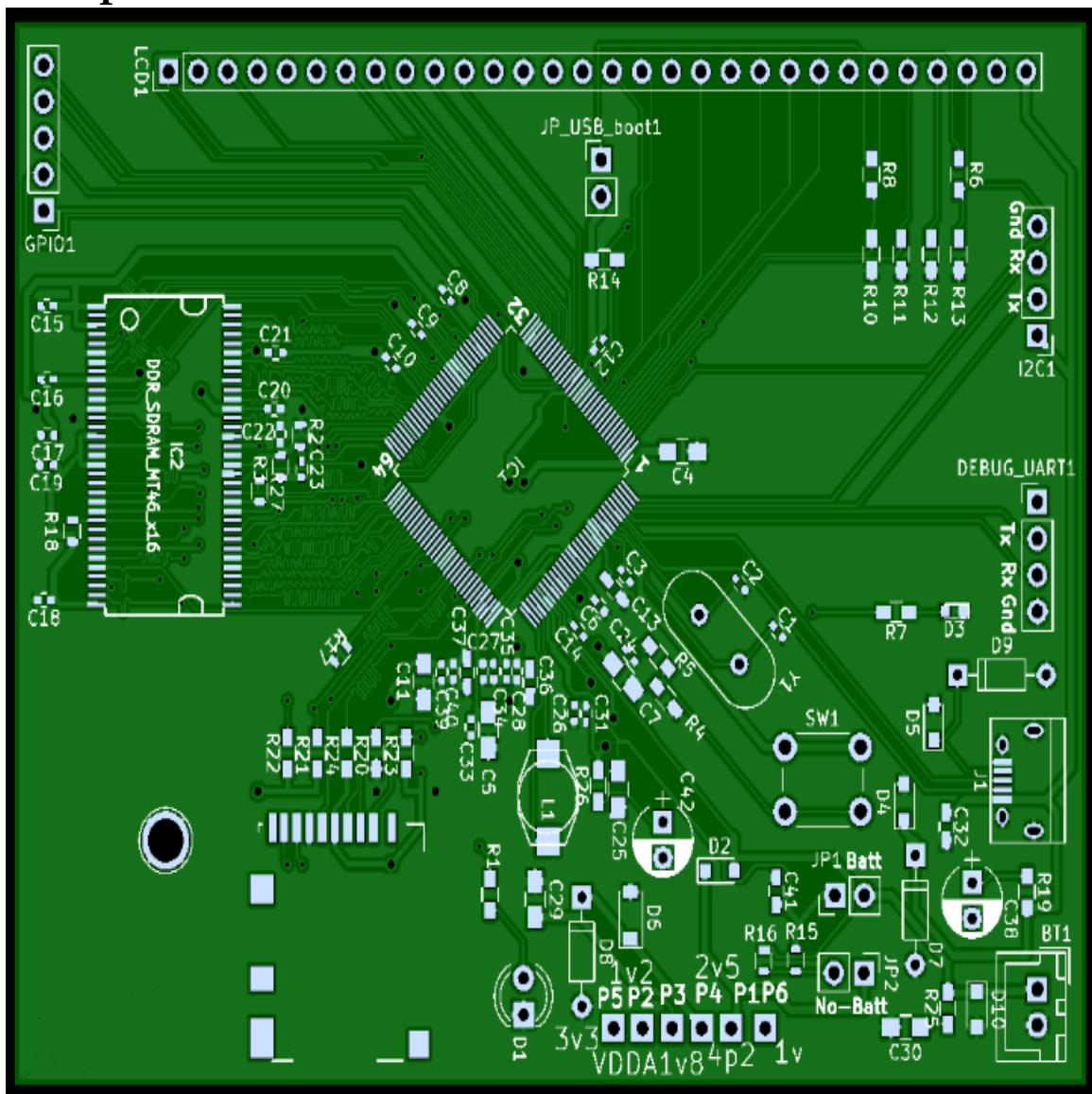


Figure 18: PCB Top view

Bottom view of the fabricated PCB

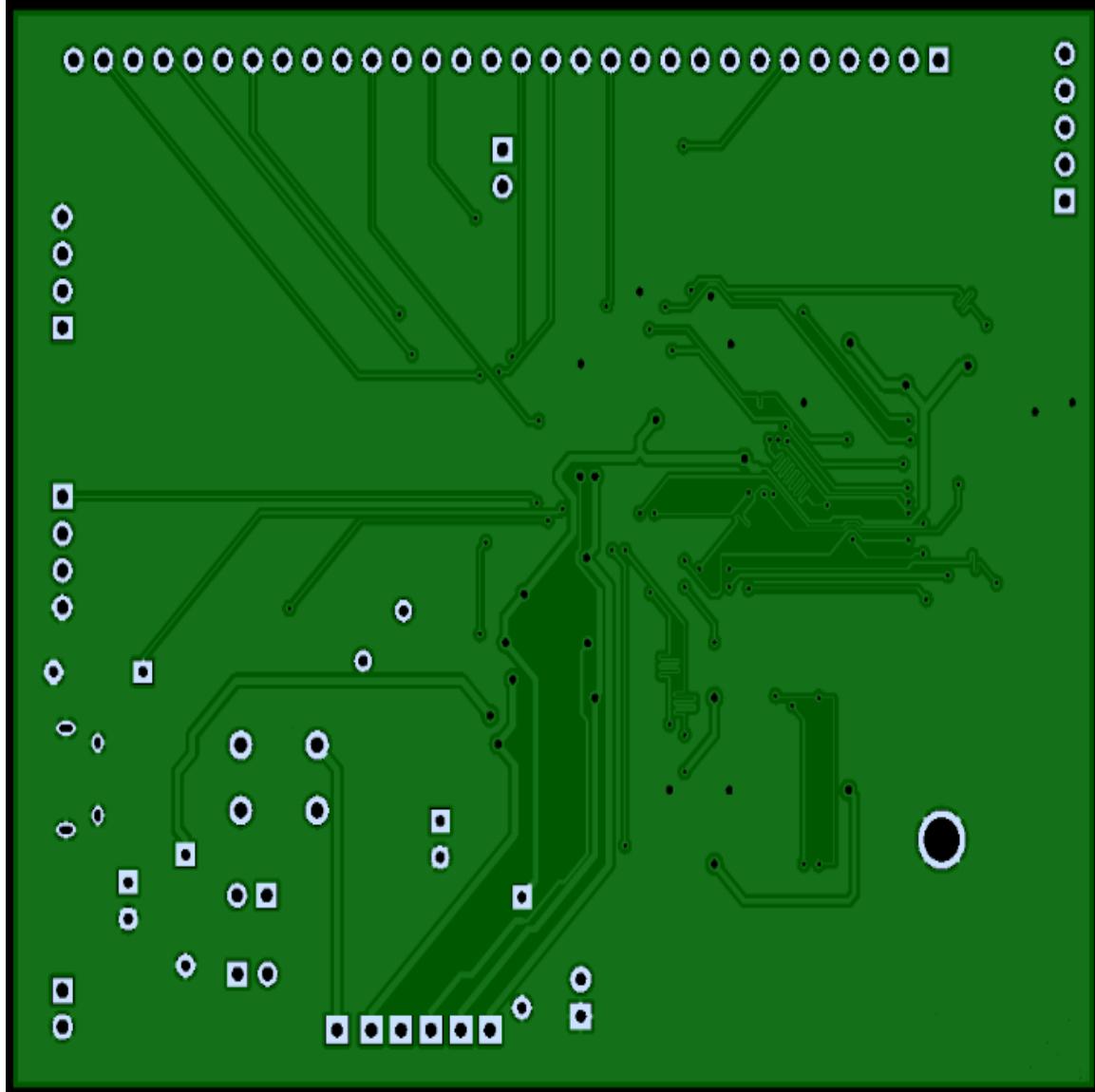


Figure 19: PCB bottom view

3.5 PCB ASSEMBLY

Components including capacitors, resistors, inductors, USB connectors, SD card connectors were procured from some of the laptop repair shops in town while few components including DDR2 RAM, ARM9 IC were shipped from Mouser electronics.

The components were assembled on the board using Hot air soldering workstation and the complete assembled board is shown in the figure below.

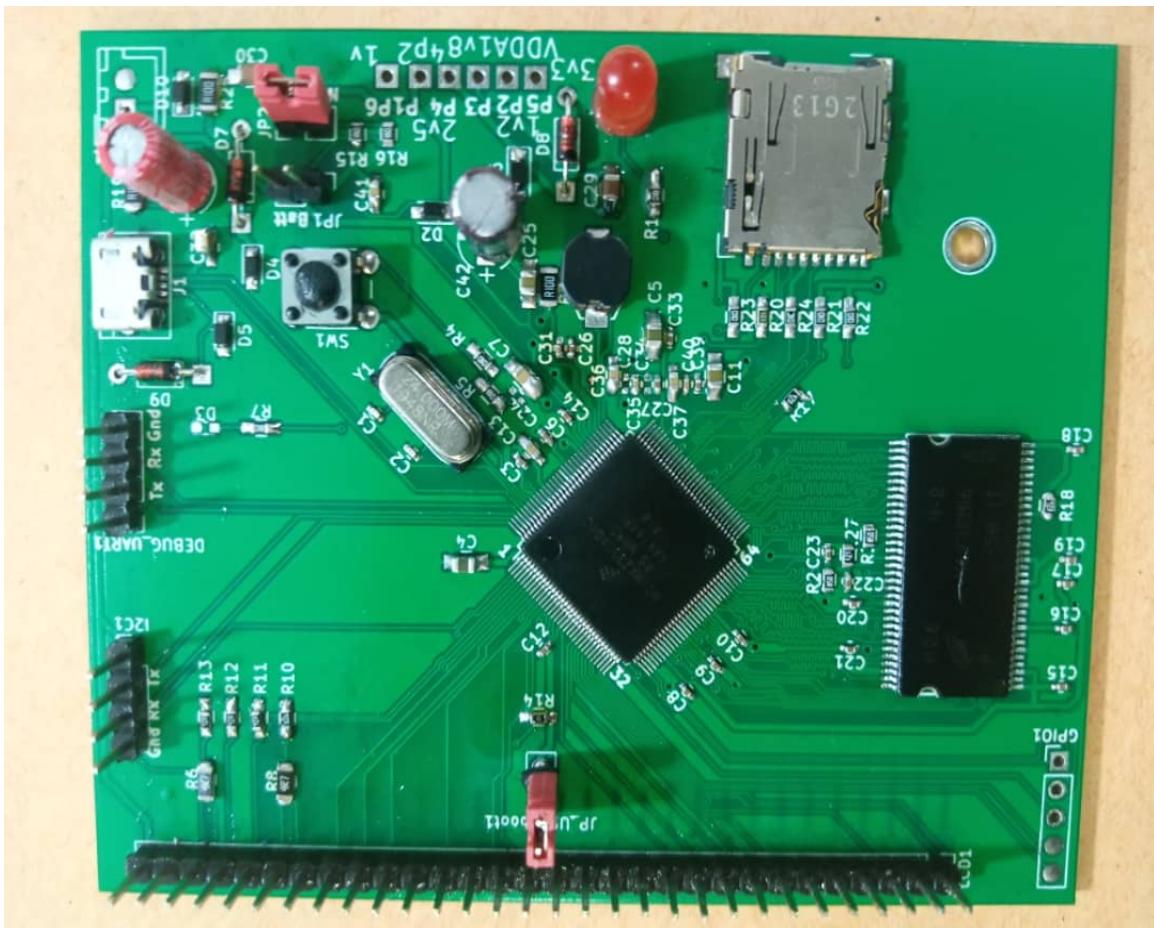


Figure 20: Assembled PCB

With the help of a multi-meter, the soldered SMD components were tested and verified for any short-circuits on the board before it was powered. By using a soldering lens, the legs of the RAM IC and the microprocessor IC were viewed to check for any short-circuit between the legs during the soldering process.

3.6 LIST OF COMPONENTS USED FOR THE ASSEMBLY

Part Number	Designator	Value
B2B-XH-A(LF)(SN)	BT1	Battery
C0402C100K5RAC	C1,C2	10p Capacitor
C0805C226M9PAC7800	C4,C5,C11,C25	22u Capacitor
CC0402KRX5R6BB105	C26,C27	1u Capacitor
CC0402KRX7R9BB103	C28,C34,C40	10n Capacitor
CC0805MKX5R5BB476	C29,C30	47u Capacitor
CC0603KRX5R5BB475	C32	4.7uf Capacitor
CC0402KRX7R9BB104	C18,C19,C20,C21,C22,C23,C24, C31,C33,C35, C39	100n Capacitor
CC0402KRX7R9BB104	C3,C6,C8,C9,C10,C12, C14,C15,C16,C17,	100n Capacitor
CC0603KRX5R8BB105	C13,C36,C37,C41	1u Capacitor
710-860020372001	C38	10uF Capacitor
UPW1H220MDD1TA	C42	22uF Capacitor
CC0805KKX7R8BB105	C7	1u Capacitor
ELM14MM5RD	D1	LED
D5VOL1B2WS-7	D4,D5,D10,D2	5.1v Diode
LTST-C194KRKT	D3	LED
SD03C-7	D6	3.3v Diode
1N4148 A0G	D7,D8,D9	5.1v Diode
61300511121	GPIO1	CONN_01X05
61300411121	DEBUG_UART1,I2C1	CONN_01X04
MCIMX233CAG4C	IC1	LQFP_128
AS4C32M16D1A-5TCN	IC2	DDR_SDRAM
10118192-0002LF	J1	USB_OTG
61300211121	JP1,JP2,JP_USB_BOOT	Pin Header 1 x 2
SRN6045-220M	L1	22uH inductor
RC0603FR-0747KL	R10,R11,R12,R13,R14,R17,R18	47k Resistor

RC0603FR-071KL	R15,R16,R5,R1,R7	1K Resistor
RC0603FR-07470KL	R20	470k Resistor
RC0603FR-07100KL	R21,R22,R23,R4	100k Resistor
RC0603FR-0710KL	R24	10k Resistor
RL0603FR-070R1L	R19,R25,R26	0.1ohm Resistor
RC0603FR-07120RL	R27	120R Resistor
RC0603FR-071K1L	R2,R3	1K1% Resistor
RC0603FR-074K7L	R6,R8	4.7k Resistor
COM-00097	SW1	PUSHBUTTON
DM3AT-SF-PEJM5	XS1	uSD- push/push_SMD
LFXTAL024417Bulk	Y1	24Mhz Crystal

Figure 21: **Table of components**

3.7 CREATION OF A BOOTABLE LINUX SDCARD

Arch Linux kernel image file was downloaded from Olimex hardware developer's website. By using Ubuntu desktop 14.10 installed on a laptop, the bootable SD-CARD was created following the procedure below:-

- The SD CARD was inserted into the Linux host.
- The card reader was un-mounted using the Linux command
- 'sudo umount /dev/sdb1'
- The Linux command 'sudo fdisk /dev/sdb' was made
- From the Linux console, the letter 'p' was entered to show the partitions on the card
- The partitions on the card were deleted by entering the letter 'd' in the console.

- A new partition was created by entering the letter 'n' on the Linux console
- A primary partition was selected by entering 'p' on the console
- The number '1' was entered to create partition 1 on the card followed by 'Enter' keyboard tab to start the partition from first block
- 16MB partitions was created on the first block by entering '+16MB' on the console
- The letter 't' was entered in the console to change the newly created partition type followed by '53'
- The letter 'n' was entered in the console to create a new partition, followed by 'p'(primary), then '2'(number) to create partition number 2.
- By entering 'w' in the Linux console, the two created partitions were written onto the SDCARD.
- The second partition on the SD card was formatted using the 'sudo mkfs.ext2 /dev/sdb2' command.
- The downloaded arch Linux kernel image file was copied to the first partition using the Linux command 'sudo dd if=/home/user/imx233-Olinuxino.img of=/dev/sdb1'

After that, the card was removed and was ready for use with the assembled board.

Putty software was also installed in windows 10 Laptop to provide a terminal for displaying the booting processes as well as programming the board from the laptop through its debug UART interface.

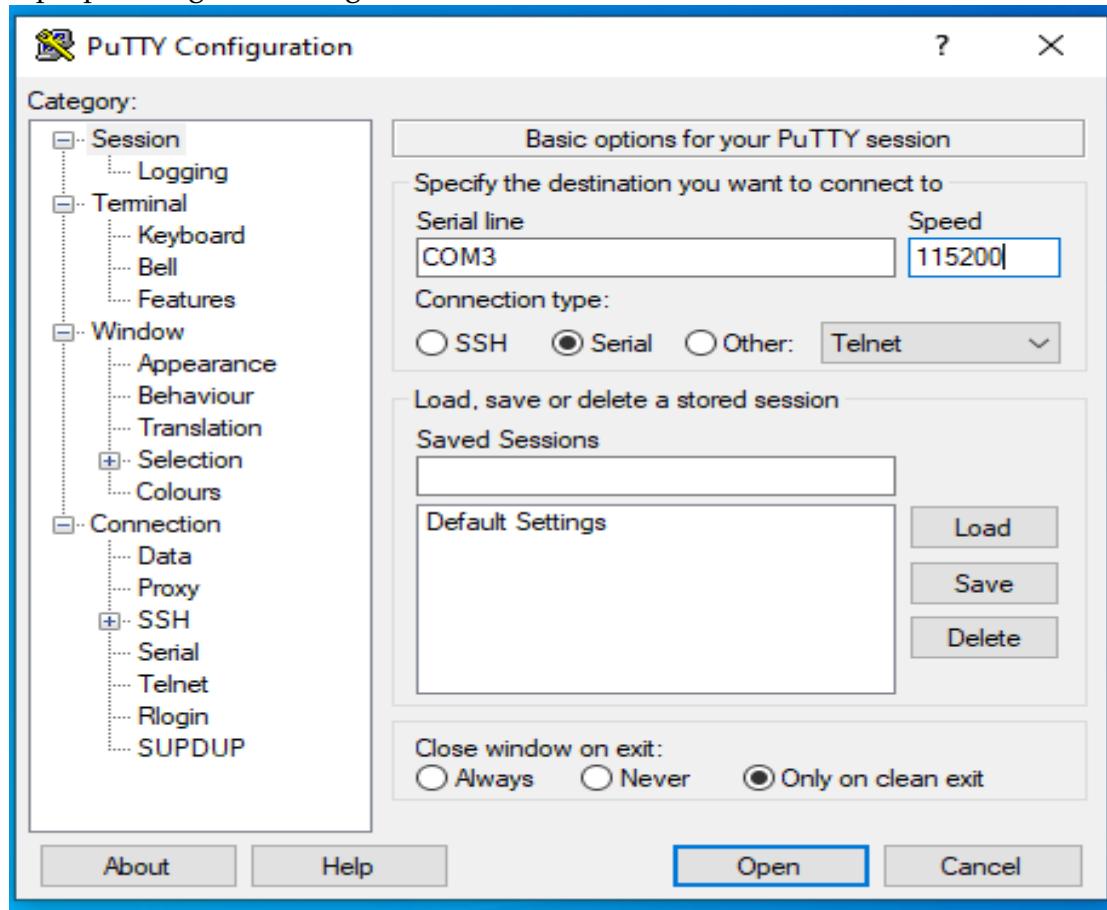


Figure 22: Putty serial interface

4 Chapter 4

4.1 Results and discussion

This chapter presents the test results of the design works that were carried out in this project. At the end of the design and construction, the system was tested to ensure that it meets the desired stated objectives and specifications that guided the entire project work. The results of the test carried out are shown below.

4.2 Full systems simulation test of the schematic design

Proteus software was used to model and simulate the above design using ARM7TDME which is a predecessor of ARM9; both being 32 bit microprocessor with a Linux running capabilities. Micro Linux kernel, Amboot boot loader and a Linux rootfs file system were loaded into the ROM and into the ARM microprocessor IC in order to test the booting capability of the schematic. Upon running the simulation, the following result was obtained as shown in the figure below.

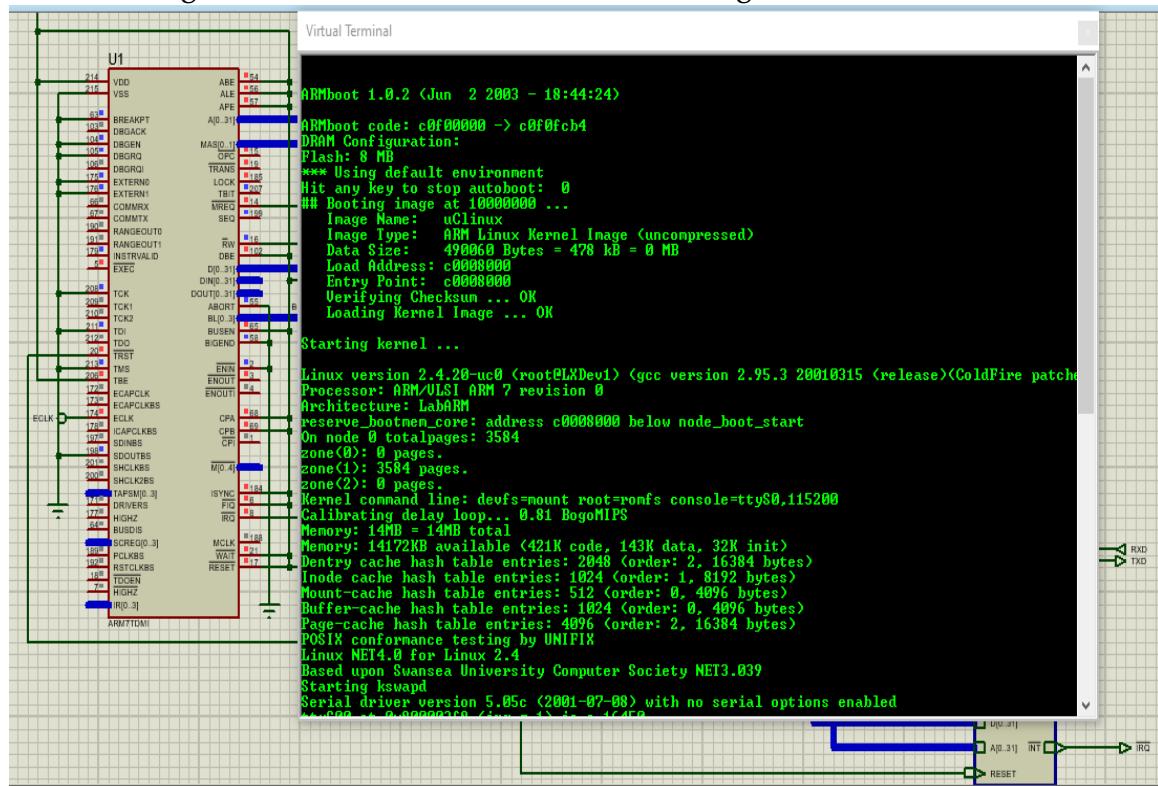


Figure 23: Full system simulation with Proteus

4.3 Testing the assembled evaluation board

The board was connected to the Laptop using a USB to UART serial cable to enable the prototyped board share the same screen with the laptop through a software called Putty. The Putty software was then configured to operate at a serial port COM3 at a baud rate of 115200bps as shown below.

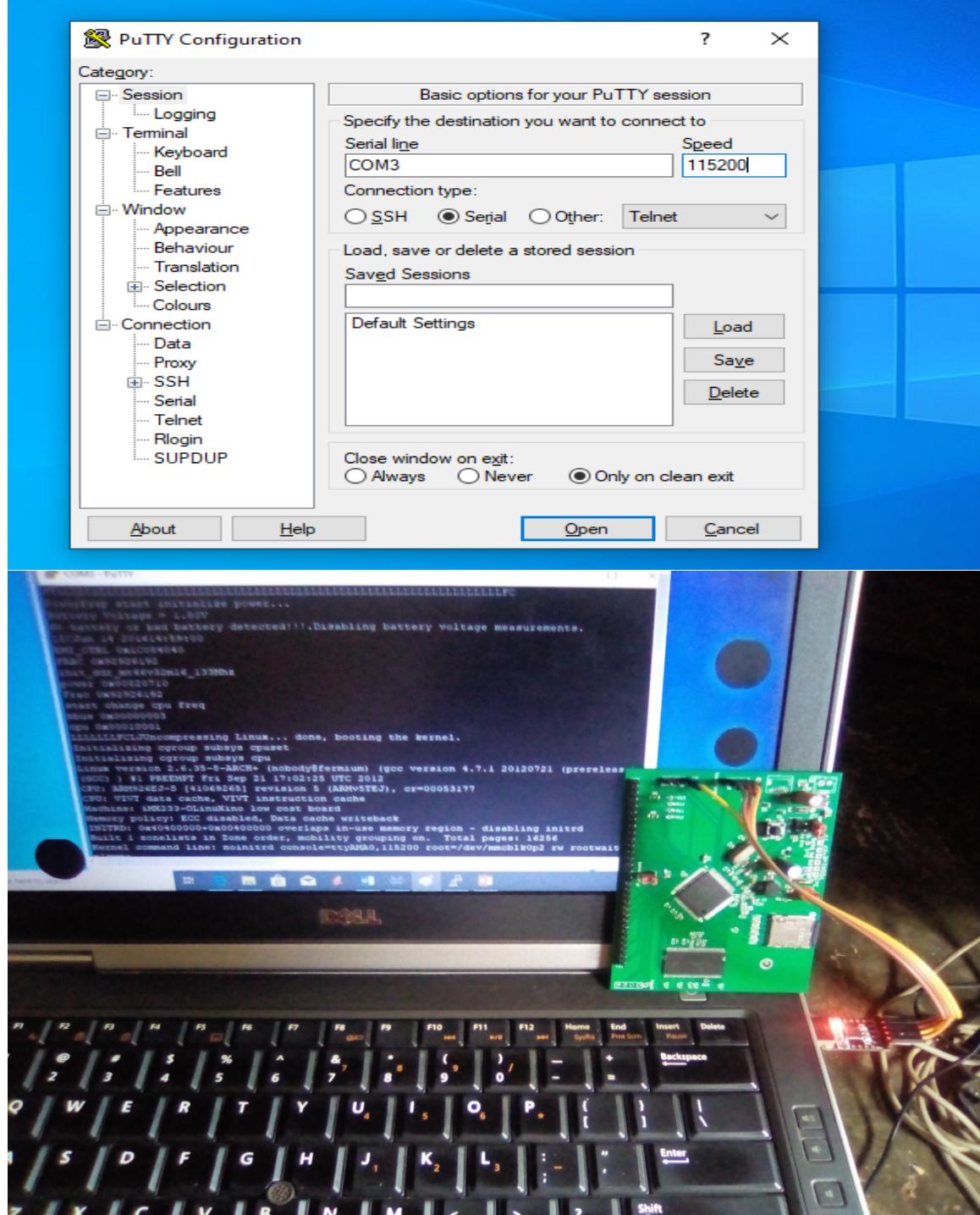
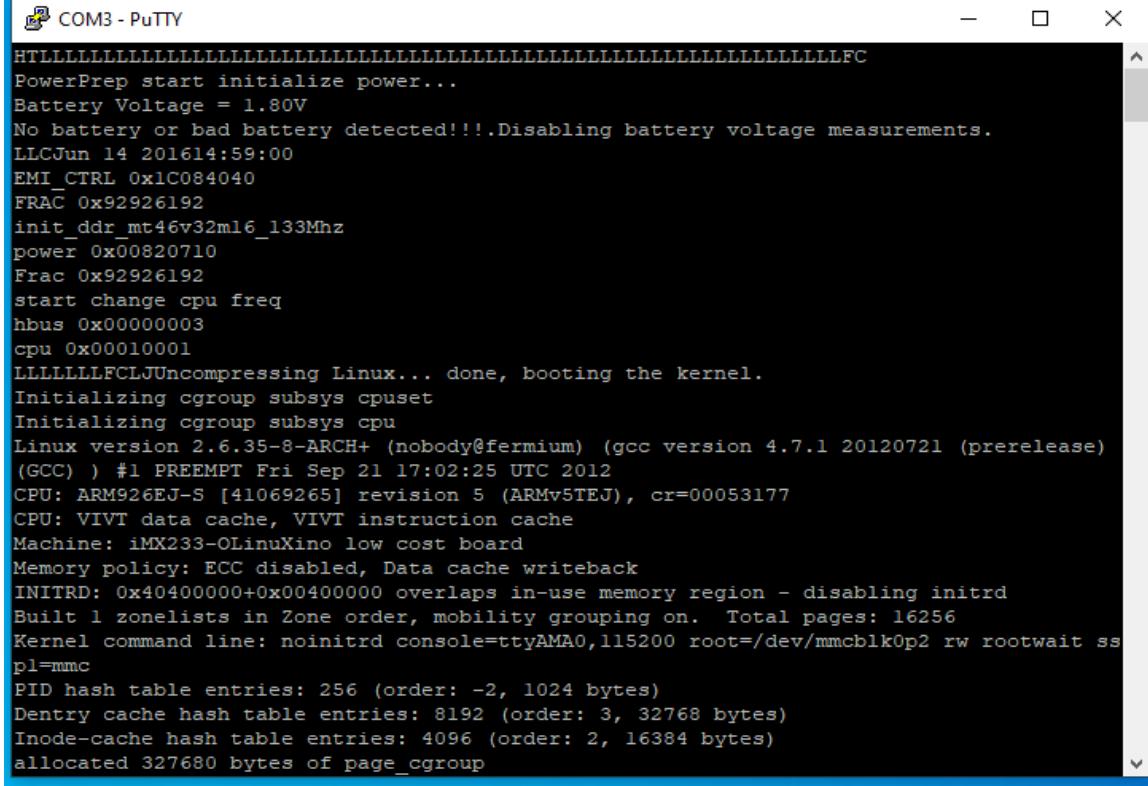


Figure 24: ARM9 booting ARCH LINUX

When board was powered by connecting the 5V DC voltage output of the USB-UART serial connector to the 5V header pin on the I2C programming pin, it was able to boot the arch Linux operating system on the SD Card as illustrated by the putty terminal screenshot shown below.



```

COM3 - PuTTY

PowerPrep start initialize power...
Battery Voltage = 1.80V
No battery or bad battery detected!!!.Disabling battery voltage measurements.
LLCJun 14 201614:59:00
EMI_CTRL 0x1C084040
FRAC 0x92926192
init_ddr_mt46v32m16_133Mhz
power 0x00820710
Frac 0x92926192
start change cpu freq
hbus 0x00000003
cpu 0x00010001
LLLLLFC LJUncompressing Linux... done, booting the kernel.
Initializing cgroup subsys cpuset
Initializing cgroup subsys cpu
Linux version 2.6.35-8-ARCH+ (nobody@fermium) (gcc version 4.7.1 20120721 (prerelease)
(GCC) ) #1 PREEMPT Fri Sep 21 17:02:25 UTC 2012
CPU: ARM926EJ-S [41069265] revision 5 (ARMv5TEJ), cr=00053177
CPU: VIVT data cache, VIVT instruction cache
Machine: iMX233-OLinuXino low cost board
Memory policy: ECC disabled, Data cache writeback
INITRD: 0x40400000+0x00400000 overlaps in-use memory region - disabling initrd
Built 1 zonelists in Zone order, mobility grouping on. Total pages: 16256
Kernel command line: noinitrd console=ttyAMA0,115200 root=/dev/mmcblk0p2 rw rootwait ss
pl=mmc
PID hash table entries: 256 (order: -2, 1024 bytes)
Dentry cache hash table entries: 8192 (order: 3, 32768 bytes)
Inode-cache hash table entries: 4096 (order: 2, 16384 bytes)
allocated 327680 bytes of page_cgroup

```

Figure 25: Arch Linux booting from the board

The booting speed was fairly fast and the ARM processor IC didn't heat so much for the time the systems was on power.

For fast booting, the Putty serial interface was set to operate at a baud rate of 115200bps using a COM3 port.

4.4 DISCUSSION OF RESULTS

Both full systems simulation of the board and the actual implementation of the board were able to boot the Linux operating system as shown above.

The ARM9 microprocessor with a processor speed of 454MHZ is only limited to operate with Debian Linux operating system and Arch Linux operating system due to its low processor speed.

Hence this board can only be deployed in applications where processor speed

is not very important. This is typical of most embedded applications which are developed to perform a single particular task such as wireless routers, terrestrial set top boxes, home security system, and point of sale machines used in supermarkets among others.

4.5 Deployment of the prototype

4.5.1 Wireless Router

This is done by adding a wireless access point peripheral having GSM modem with a SIM card to one of the telephone service company. The ARCH Linux wireless network interface is then activated and the following network interfaces configured.

Domain Name Server (DNS): From the Linux network interface, the DNS of the wireless network server is assigned if it is to work as a wireless access point. Domain names ending with .com, .go, .net, are used to assign the DNS.

Dynamic Host Configuration Protocol (DHCP): For a wireless access point, this is always configured to automatically allow the hosts devices to acquire the IP addresses automatically from the router. The IP address range to be used are given by the internet service provider (ISP).

Default Gateway: This is configured to the IP address of the internet service provider e.g. 192.168.1.1 is the most common default gateway used by ISP.

After all these changes have been made from the arch Linux terminal, the wireless network interface is saved and activated, ready to be deployed upon inserting the GSM SIM card.

Alternatively, an OPENWRT Linux firmware can be flashed into the SD Card and with the added GSM SIM card wireless access point, the board is automatically ready to be used as a MIFI router using Openwrt Linux firmware.

4.5.2 Point of sale Machine

These are portable computers which are usually installed at multiple payment points inside a shopping center. Modern point of sale computers have a touch screen where menus for selecting items bought together with their prices are displayed. By using the barcode machine linked to the point of sale machine, the items are scanned and their total costs displayed on the LCD screen. By adding a LCD screen onto this prototype, it can be deployed operate as a point of sale machines. The software interface for capturing every sales can be hosted on the arch Linux operating system and linked to the main database via router.

4.5.3 Terrestrial set top box

By installing the TV firmware into the Arch Linux OS and adding the MODEM circuit, the kit can be used as TV set top box decoder.

The deployment of the above kit is summarized in the figure below.

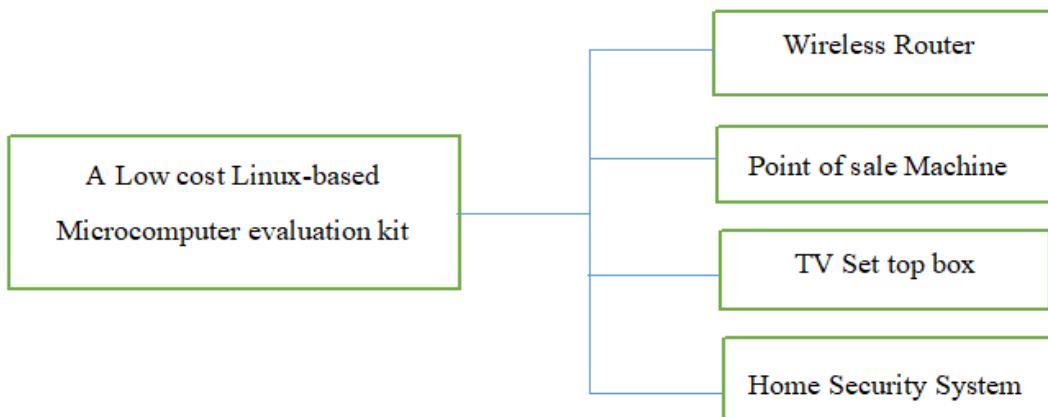


Figure 26: Application of the evaluation board

5 Chapter 5

5.1 Conclusion, Recommendations and Challenges

5.2 Conclusion

The above Linux based evaluation board was developed successfully right from the schematic design stage, up to the PCB design, PCB fabrication and assembly stage. Because of the LQFP type of the DDR2 RAM and processor used, the two layered board was fabricated cheaply and quickly by the PCB manufacturer; functioning with less signal interference as a result of its low processor speed of 454MHZ.

The arch Linux; one of the embedded Linux operating systems was able to perform well on the prototype board in terms of the booting speed and running quietly on the board without too much heat generated on both the ARM IC and the RAM.

5.3 Challenges

The following changes were encountered right from the design stage up to the assembly and testing stage among which included:-

- Scarcity of the ARM9 IC used in the design from both online suppliers such as Mouser, Digikey and from the chip manufacturer NXP semiconductor, USA.
- High shipping fees of both the components and the PCB from overseas suppliers
- Difficulties encountered in the assembly of components onto the PCB since most of the components are very tiny and required a powerful lens to view them on PCB while soldering.
- Poor quality of some of the components led to them cracking during the hot air BGA soldering work.

- Preparing a bootable Linux SD Card was a challenge since there are not very many open source procedure available online on making a bootable Linux kernel from the scratch.

5.4 Recommendations

- The above type of design and fabrication needs to done using recent high speed processors which are readily available in most online stores such as Mouser and Digikey.
- For the ease of PCB routing during the design stage, a minimum of 4-layers PCB must be used for both the signal and power line. This makes the work of length tuning to be less tedious compared to a two layer board.
- It is also necessary for the PCB fabricator to do the assembly service since they have the equipment and expertise to handle the assembly of very tiny SMD components.

References

- [1] S. D. S. S. P. Y. S. C. Nikhil Alex Thomas*, "Design and Development of ARM9 Evaluation Kit for Embedded Applications," International Journal of Reconfigurable and Embedded Systems (IJRES) , Vols. Vol.3, No2, no. ISSN: 2089-4864, pp. 62-75, 2014.
- [2] ARM-SITARA-MANUAL, "BeagleBone Black System Reference Manual," Texas Instruments, Texas, 2012-2019.
- [3] T.I, "Texas Instruments," T.I, 2016-2022. [Online]. Available: <https://www.ti.com/microcontrollers-mcus-processors/processors/arm-based-processors/>. [Accessed Monday May 2022].
- [4] H. FORSTEN, "Linux MicroComputer Board," 2012-2020. [Online]. Available: <https://hforsten.com/making-embedded-linux-computer.html>. [Accessed Saturday April 2022].
- [5] OLIMEX_BOARD_SCHEMATIC, "OLIMEX HARDWARE BOARD," OLIMEX, 2012-2020. [Online]. Available: <https://www.github.com/OLIMEX/OLINUXINO/tree/master/HARDWARE>. [Accessed Monday December 2021].
- [6] Raspberry_pi_4_schematic, "Raspberry PI," Tuesday June 2020. [Online]. Available: www.raspberrypi.org. [Accessed Sunday May 2022].
- [7] ARM-SITARA-MANUAL, "BeagleBone Black System Reference Manual".
- [8] T.I, "Texas Instruments," [Online]. Available: <https://www.ti.com/microcontrollers-mcus-processors/processors/arm-based-processors/>
- [9] BeagleBone_Black_System_Reference-Manual, Beagleboard-Community, December 2021. [Online]. Available: <https://github.com/beagleboard/beagleboard-black/wiki/System-Reference-Manual>. [Accessed Friday April 2022].
- [10] OLINUXINO, "GITHUB," OLIMEX, 2012-2019. [Online]. Available: <https://github.com/OLIMEX/OLINUXINO>. [Accessed Friday May 2021].
- [11] D. K. N. R. R. Lakshmi Narayana, "Design, Implementation of High Speed ARM Processor Based Data Acquisition and Control System Prototype," IJAREEIE, vol. Vol2, no. 10, p. ISSN (Online): 2278 – 8875 , 2013.

- [12] AllwinnerA13_Datasheet, SUNXI, April 2015. [Online]. Available: <https://linux-sunxi.org/A13>. [Accessed Monday July 2021]. W433W8640
- [13] N. B, "Design and Modeling of Arm Processor Microcontroller," International Journal of Advances in Engineering and Management (IJAEM) , vol. Vol2, no. 10, pp. 30-40, 2020.
- [14] ARMCORTEXA8_Datasheet, T.I, [Online]. Available: www.ti.com/lit/pdf/SPRUH73.
- [15] A. - S. MANUAL, T.I, 2011-2016. [Online]. Available: <http://www.ti.com/processors/sitara-arm/am335x-cortex-a8/overview.html>.
- [16] AM335X_SITARA_MANUAL, T.I, 2011-2016. [Online]. Available: <http://www.ti.com/processors/sitara-arm/am335x-cortex-a8/overview.html>. [Accessed Tuesday Feb 2022].
- [17] Diptrace, "BeagleBoneBlack_PCB_Example," Diptrace.com, Dnepropepetrovsk, Ukraine, 2021.
- [18] "nxp.com," nxp, [Online]. Available: <https://www.nxp.com/docs/en/factsheet/imx233>. [Accessed Thursday September 2022].
- [19] Alibaba, BROADCOM, May 2020. [Online]. Available: <https://www.alibaba.com/showroom/raspberry-pi-4.html>. [Accessed Monday Jan 2022].
- [20] ARMCORTEXA8_Datasheet, T.I, December 2020. [Online]. Available: www.ti.com/lit/pdf/SPRUH73Q. [Accessed Monday January 2022].