MIDDLE EAST

TECHNICAL UNIVERSITY

DEPARTMENT OF COMPUTER ENGINEERING

COSS

Computer Oriented System Solutions

SENIOR DESIGN PROJECT

INITIAL DESIGN REPORT

1. INTRODUCTION 4

1.1 PURPOSE OF THE DOCUMENT 4

1.2 PROJECT DEFINITION 4

1.3 PROJECT GOALS 5

1.4 PROJECT SCOPE 6

1.5 DESIGN OBJECTIVES 6

1.5.1 Usability 6

1.5.2 Functionality 6

1.5.3 Reliability 7

1.5.4 Efficiency 7

1.6 DESIGN CONSTRAINTS 7

1.6.1 Memory and Storage Constraints 7

1.6.2 User Interface Constraints 8

1.6.3 Response Speed 8

2. USER INTERFACE 10

3. ARCHITECTURAL AND COMPONENT LEVEL DESIGN 13

3.1 MODULES 13

3.1.1 Driver Module 13

3.1.2 AP-400 Module 16

3.2 DATA FLOW DIAGRAMS 17

3.2.1 Data Flow Diagram Level 0 17

3.2.2 Data Flow Diagram Level 1 18

3.2.3 Data Flow Diagrams Level 2 19

3.2.4 Data Flow Diagrams Level 3 21

3.3 DATA DICTIONARY 23

3.4 STATE TRANSITION DIAGRAM 28

4. TESTING 30

4.1 HARDWARE TESTING 30

4.1.1 Device Requirements Testing 30

4.1.2 Device Testing 31

4.2 SOFTWARE TESTING 32

4.2.1 Software Testing for the Main Processor 32

4.2.2 Software Testing for the Module Processor 33

5. CODING STANDARTS 34

5.1 FILE ORGANIZATION 34

5.2 VARIABLE NAMES 34

5.3 MODULE COMMENTS 34

6. GANTT CHART 35

7. CONCLUSION 40

8. REFERENCES 41

1. INTRODUCTION

1.1 PURPOSE OF THE DOCUMENT

This report mainly covers the design of the project. Project is not fully designed at the moment; therefore this document will give a simple introduction about the design. In this introduction part, you can find what the project is and background information about the project. This part will also include the design requirements for an embedded software and hardware development which are design constraints.

In second part you can find how a user interface can be designed. Since the project is also about electronic with embedded programming, the user interfaces can be different than usual. It will be a graphical liquid crystal display (commonly abbreviated LCD).

Next part contains the architectural design of the system and design of the components. Components are explained as modules. The two modules, namely driver module and AP-400 module, are explained.

The forth part is about the testing of the system. When a prototype is produced, the team has to control all functionalities of this prototype whether working perfectly or not. Therefore project team prepared a testing progress for prototypes.

The fifth part is about syntax and file naming.

At the sixth part, you can find the Gantt chart of project.

The last part is the conclusion part which contains the important points and the milestones of the project. Team decided on these subjects and will obey the rules to cover these points.

1.2 PROJECT DEFINITION

The name of our senior project is defined to be AP-490. AP-490 is an upper version of AP-400 produced by AirTies that is one of leader companies in the network sector in Turkey. We decided that name to be AP-490 since the last product will be presented to AirTies, who will manufacture the product, if the quality is enough for production.

The main difference between AP-490 and AP-400 is the graphical LCD. This module will show the general data in a simple way that user can understand the information easily. User can see wireless networks that AP-490 found, connected PCs and their detailed

information like upload/download speed and signal strengths. The most important key is that users do not need to establish a connection between AP-490 and PC. Users can learn the answers to questions like “Which PCs are connected to my network?” by just pressing some buttons.

1.3PROJECT GOALS

At the beginning, our aim is to add LCD on AP-400 and produce a new product type, AP-490. However we can include some other extensions to project for further improvements. New device will have nicer view and more functionality without any complexity. So the end-user will want to buy this product more. By this project, we are also aiming to be first with this new unit. Result of researches and literature surveys show that there is not any device in that class (802.11g) with an LCD in Turkey’s Market.

In technical point of view, we have entered into a field that needs more and more employment than other sectors of computer science. While electronic devices become smaller and portable, the development of them becomes more important. To be in such a sector, we expect to find more opportunities as we improve.

Main Goals from Co-company’s Point of View:

- There will be an addition into their product catalog
- More potential customer
- The advantage of being first in sector
- New opinions can be found while project is developing

Main Goals from End-User’s Point of View:

- Easy to see information about networks
- Nice representation of information
- Some options that can be changed like LCD contrast
- Not need to establish PC connection

1.4PROJECT SCOPE

Primarily, we will have an AP-400 with LCD functionality. The basic functions shown on LCD will be:

- How many connections there are
- MAC addresses' of connections
- Current connection speed

Secondarily objectives are:

- Using high resolution LCD to have better user interface.
- Ability to warn by sound effects
- Displaying the amount of data flow for specific periods of time.

1.5DESIGN OBJECTIVES

1.5.1 Usability

Since the product addresses to everybody; the control of the buttons, which control what to display on LCD, must be easy. Our menus will be easy to understand and will have tolerable menu depths in order for the users to be able to control the LCD easily and not to get lost within the menus.

“The simpler the better” is our motto.

1.5.2 Functionality

In order to specify what to show on screen, we have made surveys, asked end-users about what would be best to view. LCD will view only these mostly wanted data on the screen. We will not confuse the user with unnecessary data or even with the options of viewing these data. So the device will meet end-user expectations but will not confuse them at the same time.

1.5.3 Reliability

The device will always show real time data to the user. We do not have any fault toleration about this subject. We do not have the luxury to view wrong data on the screen.

There will not be an upper time limit for the device to function. LCD will function as the device functions; it will stop functioning as the device itself stops functioning.

Unexpected inputs (button configurations) will not result in device halt. At the worst case, we will show an error message on LCD.

1.5.4 Efficiency

The device will always be efficient to use.

Number of connections to the device will not affect LCD. One connection or n connections will make as much difference as it does with the main AP-400 device.

The type of the information shown on the screen will not result in any performance change. Getting connected IP addresses or the connection speeds and showing them on the device will cost the same for the module circuit.

1.6DESIGN CONSTRAINTS

1.6.1 Memory and Storage Constraints

Since the memory and storage resources of AP-400 processor and module processor are limited, we consider these as constraints.

While programming processors, we will pay attention to storage and memory of the devices. Memory allocation and deallocation will be handled with the utmost priority to prevent a crash in the system. We are limited while programming the devices; even our codes must not be too long.

With the addition of the new processor we guess memory will not be that much of a problem for us, but still, we are well aware of that constraint.

1.6.2 User Interface Constraints

Project has user interface constraints that a commercial product project should have.

One of the user interface constraints we must pay attention to is the simplicity of the UI. As a commercial product, the LCD functionality should be easy to control. User should be able to handle the LCD without so much effort.

For that purpose, we plan to make the control panel similar to other devices used in everyday life. There will be up-down-enter-back buttons as there are within every computer. By such a control mechanism that almost everybody are familiar with, we aim for the users not to feel any unfamiliarity while using the device.

Another important user interface constraint is the nice view of the UI.

Product is for customers and as a customer people want a nice view of the device they buy. We cannot just show them the data. LCD must have a nice view and people should enjoy using it, not hate it at least. That is why we consider UI as a constraint.

For nice view, we plan to produce simple graphics or even some animations. LCD will have a charm itself.

1.6.3 Response Speed

The project has real-time performance constraints that every embedded design project has.

For reasons such as usability and functionality, the system might result in decreased performance for the system. That would be intolerable for such a commercial product. It must respond to the user requests immediately. Otherwise, the product would fail although it would meet the project objectives.

One of the reasons why we add a new processor to the existing system is for keeping the performance of the already existing system as it is. Main processor and its circuit are not designed for the addition of a new LCD functionality module and thus, without a new circuit, addition of an LCD and its functionalities by reprogramming AP-400 processor will most probably result in a performance decrease in the system. By adding a new processor and its circuit as a new module to the system, we plan to reduce the cost of addition of a new functionality to minimum.

We do not expect the new PIC processor, itself, to slow down the system since it will only get some information from processor of AP-400, process it and send the information, requested from the user as inputs from buttons, to LCD.

The performance is one of the most important aspects of an embedded project and thus, we will not let any significant performance decrease because of adding a new functionality to an existing system.

2. USER INTERFACE

User interface is the main object of this project; since the end-user will prefer this product due to user interface property. The purpose of the user interface is to display variety of information without a PC connection. Simply, user can check many data by pressing a few buttons.

The graphical user interface is not a usual computer screen with many buttons and not controlled by a mouse. Interface is composed of only a simple graphical LCD (Liquid Crystal Display) which makes the project an embedded development project. For designing user interface, we searched various GUI drawing programs but we cannot find any user friendly program so we prefer to design in paint program.

In the start page, our product name and our company name will be displayed like the figure below (Figure 2.1).

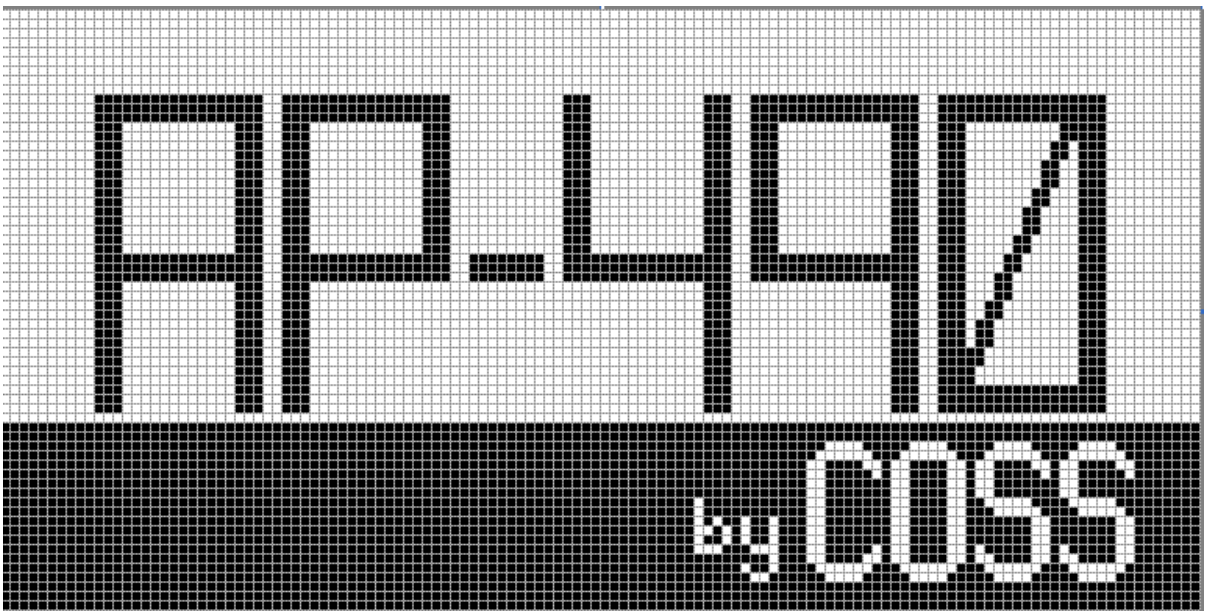


Figure 2.1

In the Main menu (Figure 2.2), networks' names, connected PCs, statistics and options submenus will exist. In addition, for the purpose of showing activities of AP-400, there will be a wave icon to determine the wireless activities and a computer icon to determine the wired activities on the top of the screen. In addition to these, we planned to show the time on each menu.

Figure 2.2

In the submenu Connected PCs; MAC addresses, connection status, IP addresses, computer names, upload download speeds, last day statistics and rates will be displayed. As an example for these MAC addresses is shown in the figure 2.3.

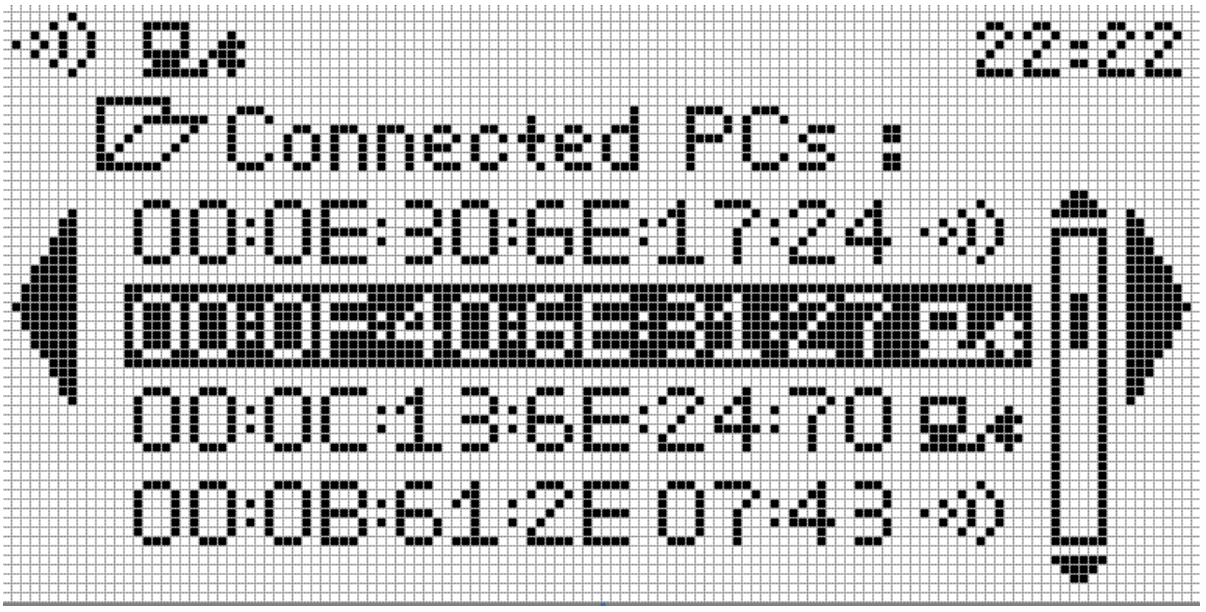


Figure 2.3

In the submenu Options, there will be contrast settings, language selection, and sound on/off selection as you can see in the figure 2.4.

Figure 2.4

In the Networks submenu; wireless networks, which are detected by AP-400, are viewed like Figure 2.5.

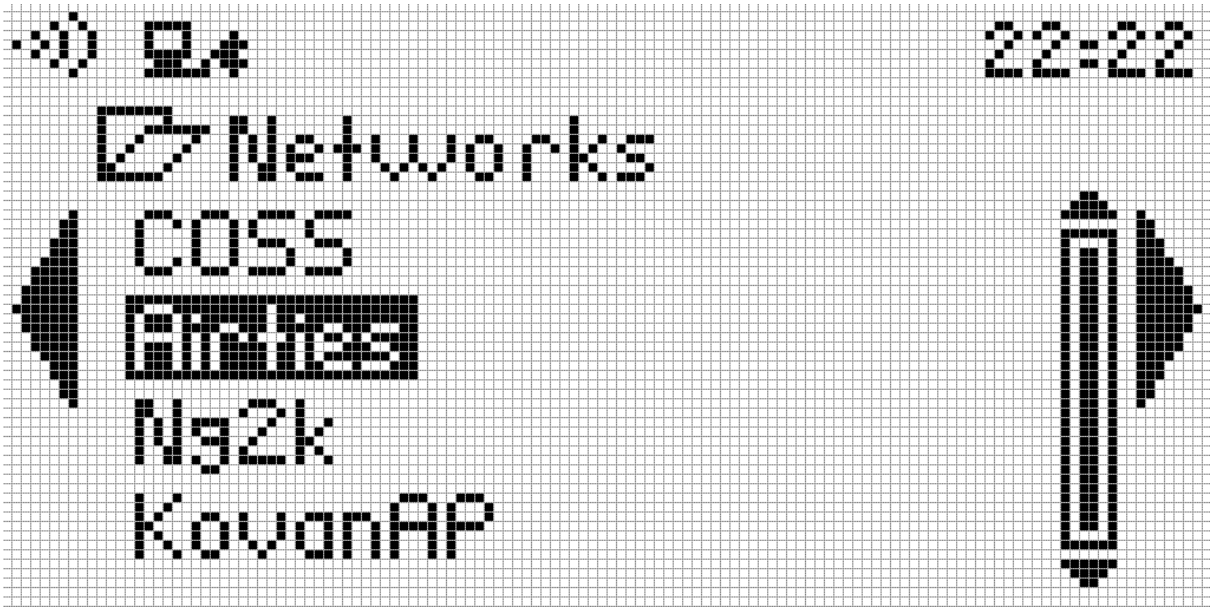


Figure 2.5

3. ARCHITECTURAL AND COMPONENT LEVEL DESIGN

3.1MODULES

There are two part of project mainly, the additional module and AP-490 module. In driver module, we will design a driver board which helps AP-490 to display on LCD. The other module is the main module which contains developments on processor of AP-400 and embedded system.

3.1.1 Driver Module

This module has been developed completely by the team. This module is separated to four sub modules according to their hardware:

- Main Processor
- Liquid Crystal Display (LCD)
- Buttons
- Serial Port Converter

This four part will be on an electronic board, which will designed by the team. Main processor is the processor that will do all actions. The other three sub modules are like slave devices that obey processor commands. The system is similar to a PC. Processor is like processor of a PC, and LCD is like a monitor and buttons are keyboard.

The design of driver board is shown on below figure.

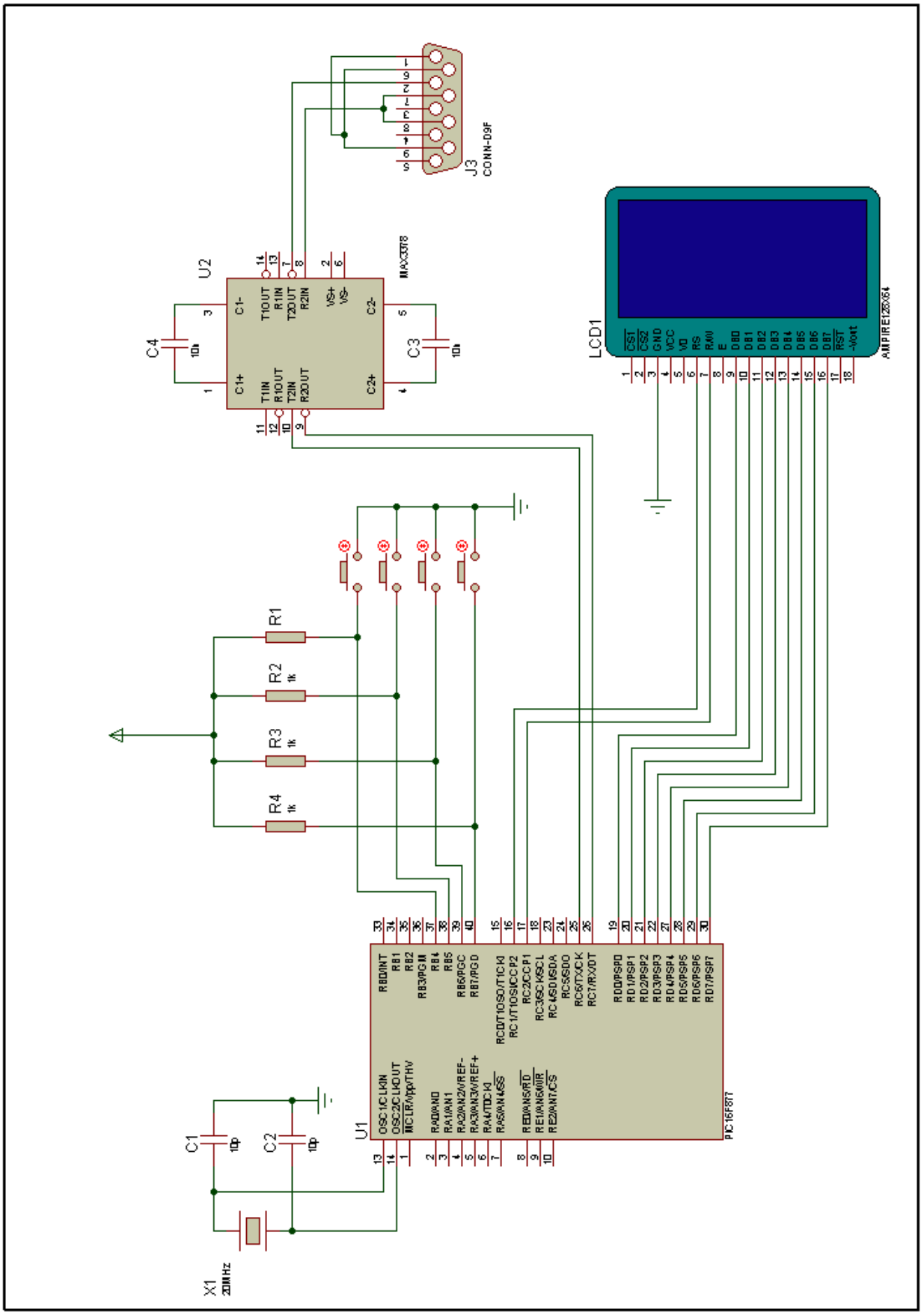


Figure 3.1

3.1.1.1 Main Processor

Main processor is the brain of this module and project. Every task will work on this part. It should have the capabilities that we decided before. Therefore we decided to use PIC16F877. Team members are familiar with this chip from CEng 336 course of Computer Engineering. Chip has mainly below features:

- There are 35 single word instructions
- Speed: DC - 20 MHz clock input
- Interrupt capability (There are 14 different sources)
- Hardware stack has eight level
- Serial Communication
- 10 Channels A/D converter

AP-400 processor runs with 180MHz clock, hence we will run this processor with 20MHz clock in spite of 4MHz. Serial communication feature will be used to establish a communication between this processor and AP-400's processor. The requests will send to AP-400 and all data will be received from AP-400 by using this serial port. The signal transfers will be checked by interrupts, so the communication will can be asynchronous.

3.1.1.2 Liquid Crystal Display (LCD)

LCD is the screen of AP-490. All visual interaction between user and AP-490 will be done via this hardware. The model selected to be used in this project is ABG128064 - A15. It has 20 pins connection and we will use eight pins for data transfers.

Since it will be a graphical LCD, we have to create our fonts and images for screens. The created images and fonts will be designed to reusable. So that the writing progress on LCD will be easier. To display on LCD, we decided on a hidden cursor and move it on the screen. After the correct location is reached, the data will be sent to display.

3.1.1.3 Buttons

Buttons provides user to make requests on AP-490. They will be simple electrical circuits which will be controlled and checked by processor. The buttons will create an interrupt when pressed. That makes processor goes into interrupt routine. In interrupt routine,

processor will check which button is pressed and will prepare a view by considering current state of LCD.

Since buttons are only simple electrical circuits, only one action of press will give many signals. Therefore, we have to control additionally this situation.

3.1.1.4 Serial Port Converter

Serial port converter is one of the essential hardware of this project. The previous model of AP-490 works with 3.3 inner circuit voltage. On the other hand, the driver board works with 5.0 inner circuit voltages and this difference will make a corruption in the system.

In this manner, we have to convert one voltage to other. This part will do this task. The converter circuit is a standard circuit. We will change only the units of it.

3.1.2 AP-400 Module

AP-400 module has mainly two different aspects. Firstly, there will be a sort of data retrieving from current state of AP-400, like data retrieving from database. And secondly, the collected data will be sent from AP-400 to driver board processor.

The data retrieving consists of researching in the system of embedded Linux in AP-400. Team will develop codes to retrieve the needed data. The programming language is surely C. However we will use bash scripts also if needed, in order to take information.

The information sending procedure works with serial communication hardware of AP-400 processor in order to send data and receive commands. It will work asynchronously which means when a command comes to AP-400; it will do the necessary processes and will send a respond.

3.2DATA FLOW DIAGRAMS

3.2.1 Data Flow Diagram Level 0

Data flow diagram level 0 is the top most model of our project. In this level, only general modules and relations between these modules are drawn. End-User is the user who wants to gather some information from AP-490 by using the buttons. Driver board is the board that will interact with user, LCD and AP-400. Data will be gathered from AP-400 and displayed on the LCD.

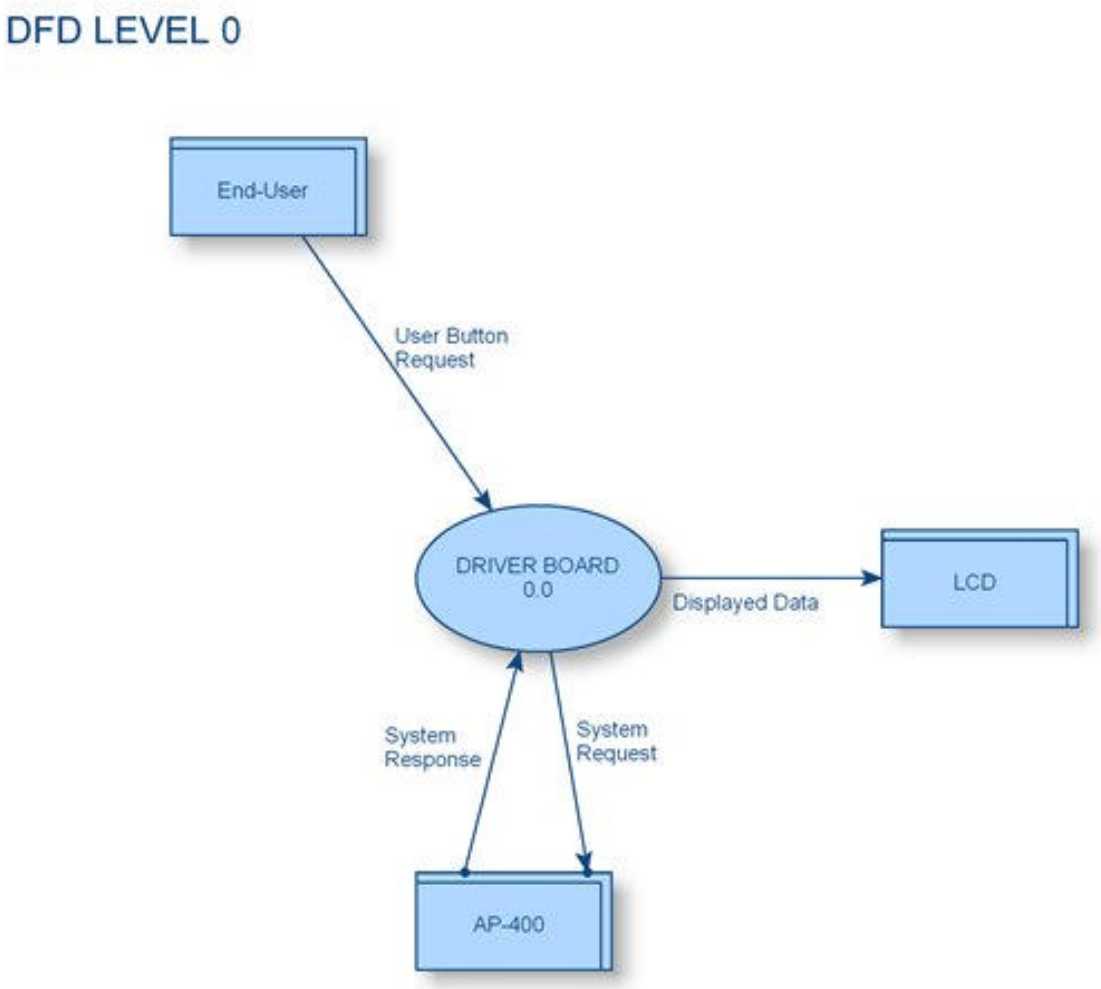


Figure 3.2

3.2.2 Data Flow Diagram Level 1

Data flow diagram level 1 is a lower model of our project. In this level, inside of driver board is drawn with relations. Get info from AP-400 module is active when back or enter button is pressed by the user, because only at these steps new data must be gathered from AP-400. Change menu module is active every time; because no matter which button is pressed, the menu displayed on LCD will be change.

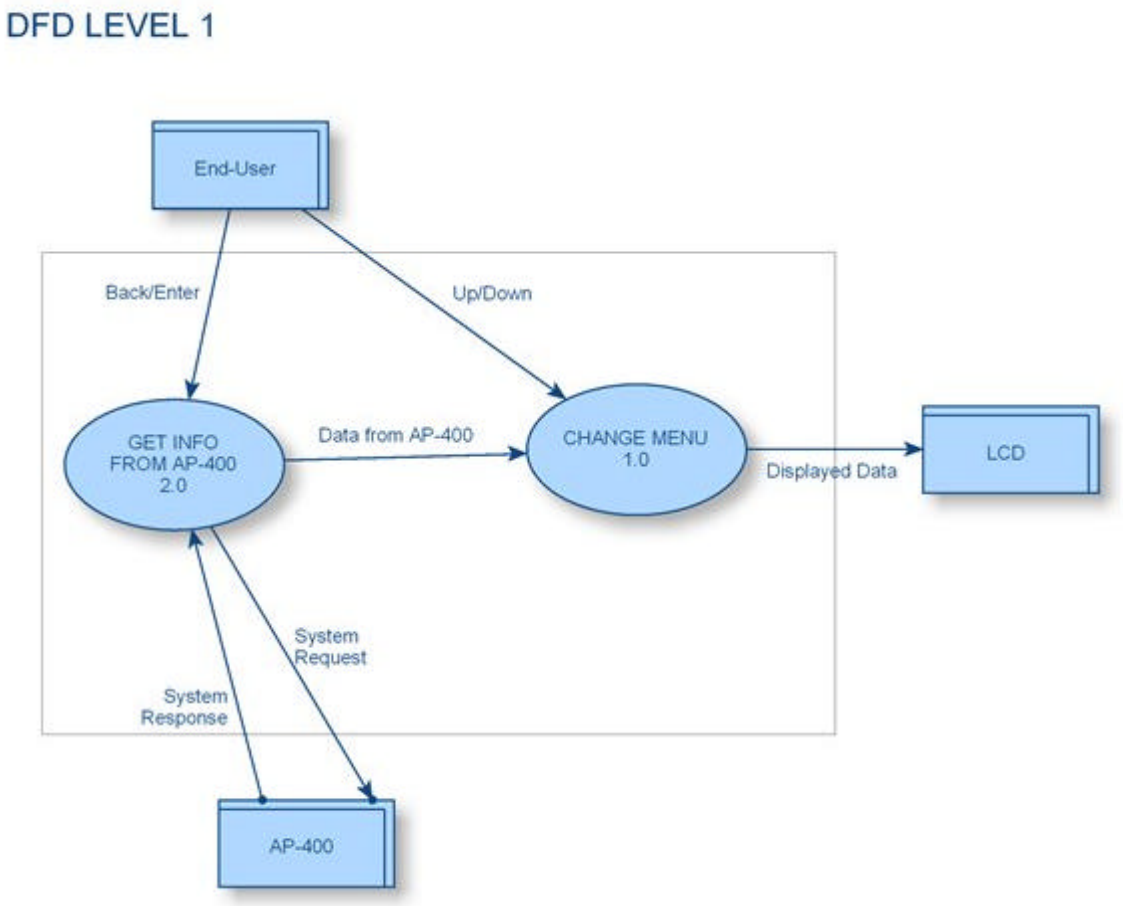


Figure 3.3

3.2.3 Data Flow Diagrams Level 2

In this level, insides of modules, which are not explained much more in level 1, are drawn with relations. Get info from AP-400 module and change menu module are drawn separately.

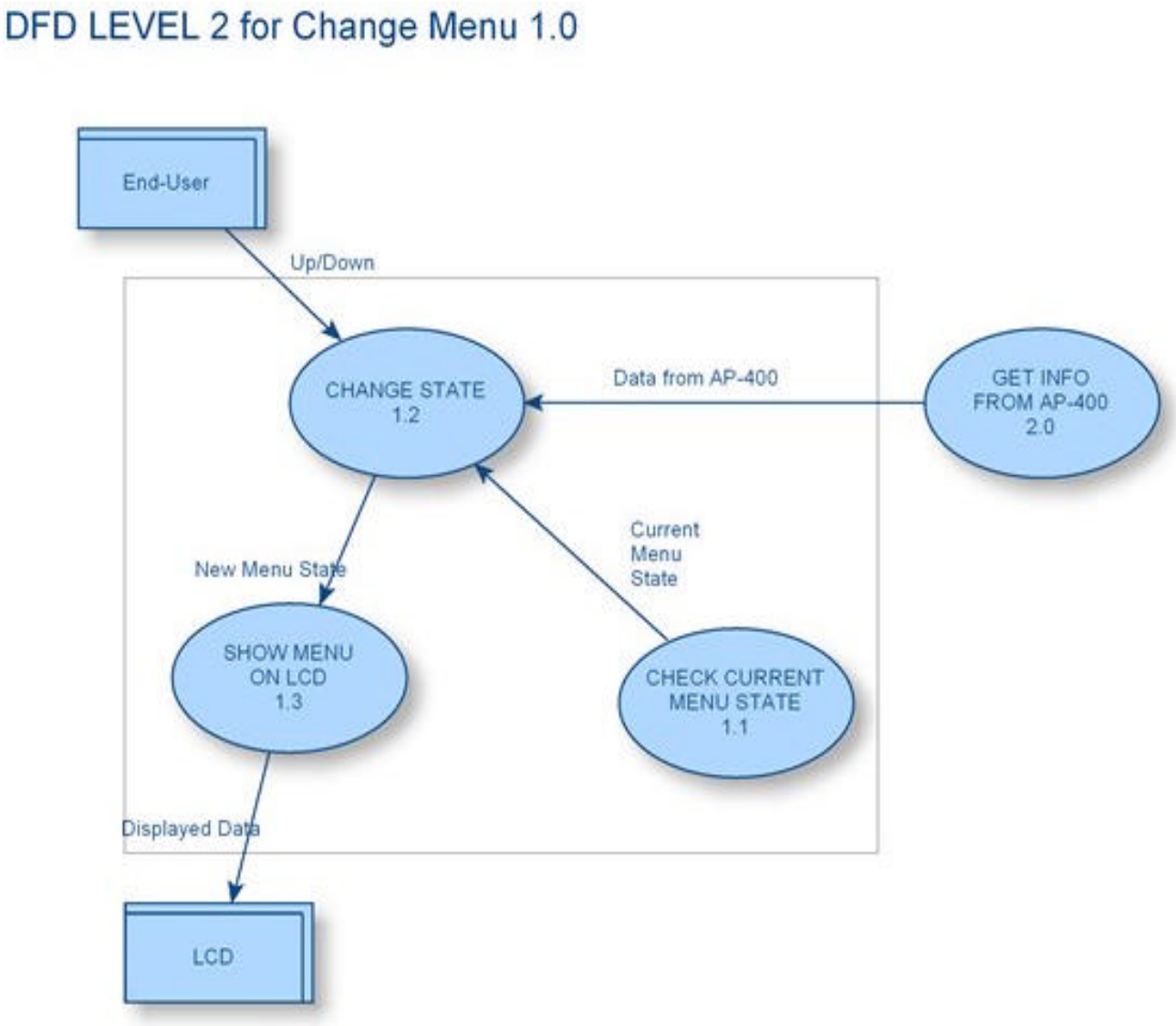


Figure 3.4

DFD LEVEL 2 for Get Info From AP-400 2.0

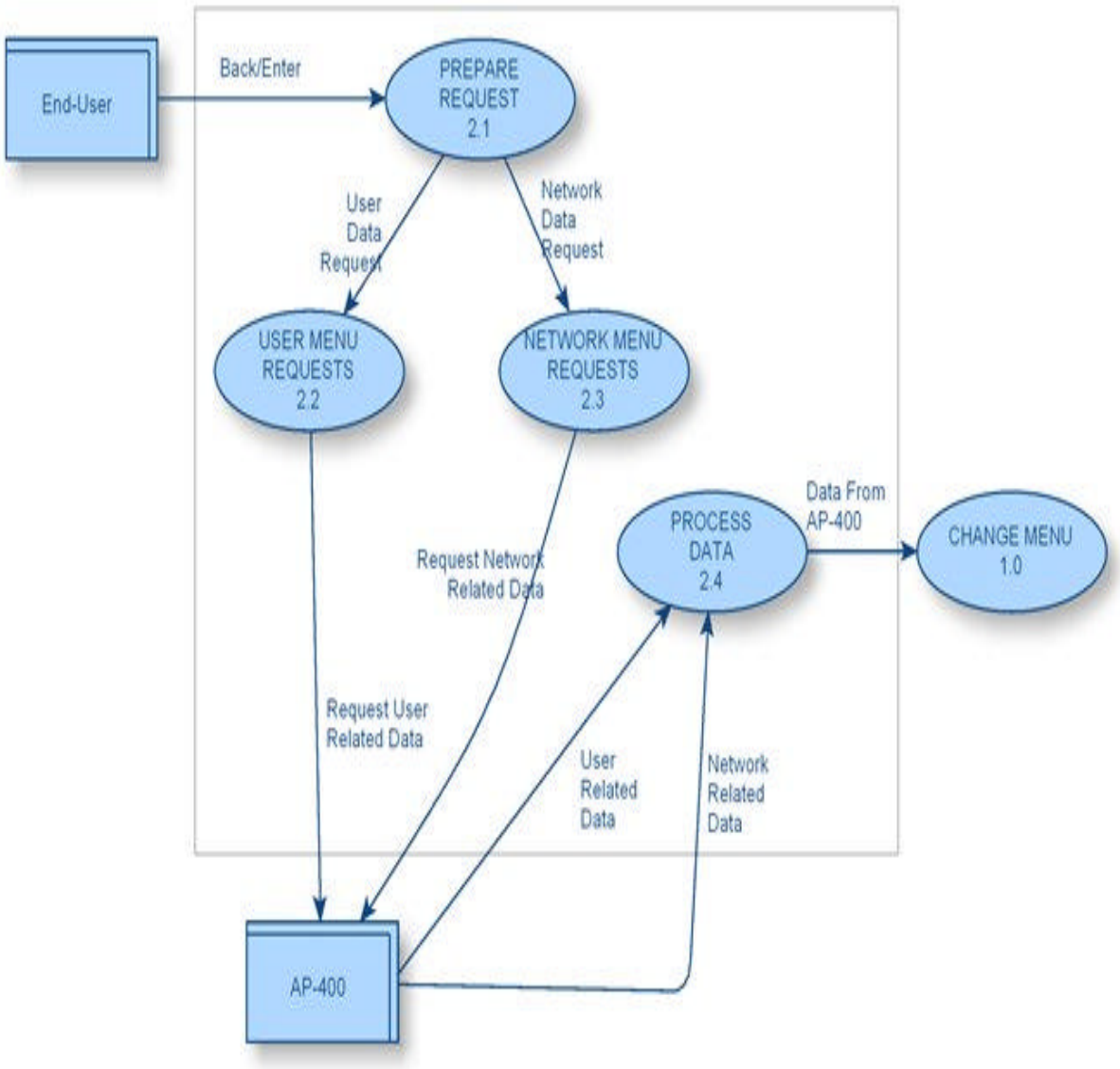


Figure 3.5

3.2.4 Data Flow Diagrams Level 3

In this level, insides of some modules, which are not explained much more in level 2, are drawn with relations. User menu requests module and network requests module are drawn separately.

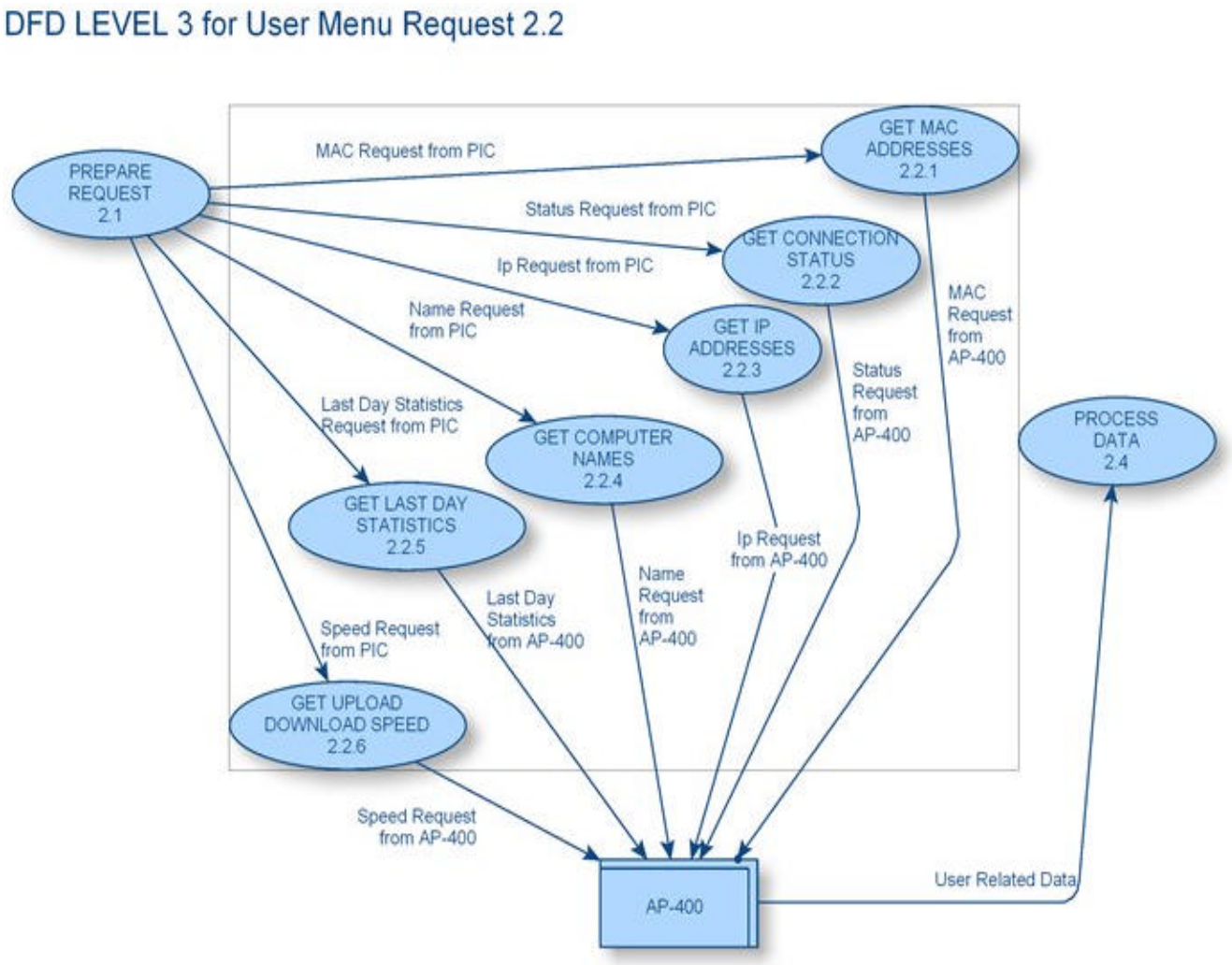


Figure 3.6

DFD LEVEL 3 for Network Menu Request 2.3

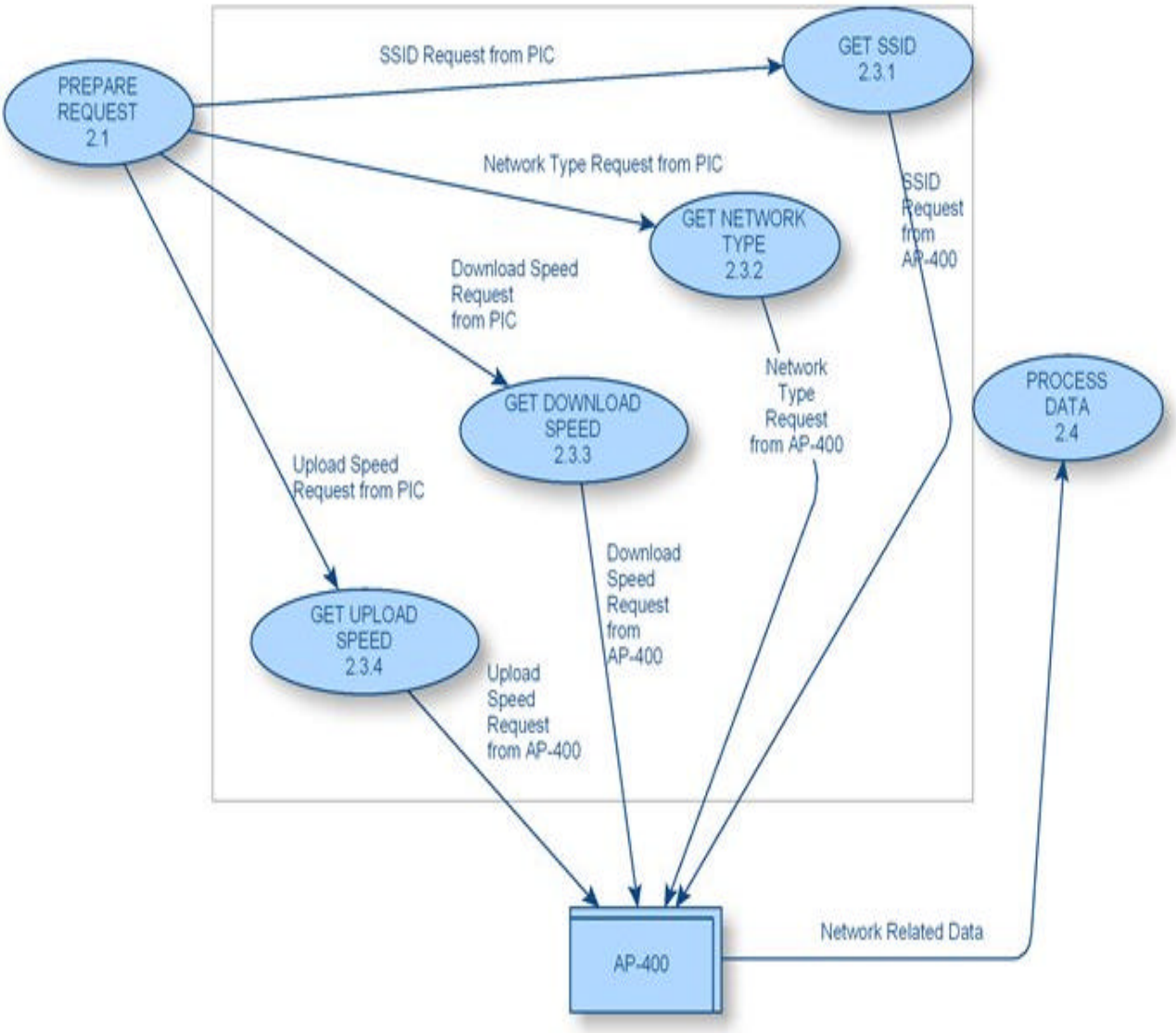


Figure 3.7

3.3DATA DICTIONARY

NAME	USER BUTTON REQUEST
FROM	END USER
TO	DRIVER BOARD 0.0
FORMAT	ELECTRICAL SIGNALS
DESCRIPTION	The signals that show which button pressed.

NAME	DISPLAYED DATA
FROM	DRIVER BOARD 0.0 CHANGE MENU 1.0 SHOW MENU ON LCD 1.3
TO	LCD
FORMAT	ELECTRICAL SIGNALS
DESCRIPTION	The processed data that includes the display information of display screen.

NAME	SYSTEM REQUEST
FROM	DRIVER BOARD 0.0 GET INFO FROM AP-400 2.0
TO	AP-400
FORMAT	RS232 SERIAL PORT COMMUNICATION
DESCRIPTION	Include requester information.

NAME	SYSTEM RESPONSE
FROM	AP-400
TO	DRIVER BOARD 0.0 GET INFO FROM AP-400 2.0
FORMAT	RS232 SERIAL PORT COMMUNICATION
DESCRIPTION	Include response of requester information.

NAME	UP/DOWN
FROM	END USER
TO	CHANGE MENU 1.0 CHANGE STATE 1.2
FORMAT	ELECTRICAL SIGNAL
DESCRIPTION	The signals that show which button pressed (up or down).

NAME	BACK/ENTER
FROM	END USER
TO	GET INFO FROM AP-400 2.0 PREPARE REQUEST 2.1
FORMAT	ELECTRICAL SIGNAL
DESCRIPTION	The signals that show which button pressed (back or enter).

NAME	DATA FROM AP-400
FROM	GET INFO FROM AP-400 2.0 PROCESS DATA 2.4
TO	CHANGE MENU 1.0 CHANGE STATE 1.2
FORMAT	ELECTRICAL SIGNAL
DESCRIPTION	Gathered data from AP-400.

NAME	CURRENT MENU STATE
FROM	CHECK CURRENT MENU STATE 1.1
TO	CHANGE STATE 1.2
FORMAT	ELECTRICAL SIGNAL
DESCRIPTION	Current menu state on LCD.

NAME	NEW MENU STATE
FROM	CHANGE STATE 1.2
TO	SHOW MENU ON LCD 1.3
FORMAT	ELECTRICAL SIGNAL
DESCRIPTION	New menu state will be displayed on LCD.

NAME	USER DATA REQUEST
FROM	PREPARE REQUEST 2.1
TO	USER MENU REQUESTS 2.2
FORMAT	ELECTRICAL SIGNAL
DESCRIPTION	Include requester information related to user.

NAME	NETWORK DATA REQUEST
FROM	PREPARE REQUEST 2.1
TO	NETWORK MENU REQUESTS 2.3
FORMAT	ELECTRICAL SIGNAL
DESCRIPTION	Include requester information related to network.

NAME	REQUEST USER RELATED DATA
FROM	USER MENU REQUESTS 2.2
TO	AP-400
FORMAT	RS232 SERIAL PORT COMMUNICATION
DESCRIPTION	Include requester information related to the user.

NAME	REQUEST NETWORK RELATED DATA
FROM	NETWORK MENU REQUESTS 2.3
TO	AP-400
FORMAT	RS232 SERIAL PORT COMMUNICATION
DESCRIPTION	Include requester information related to the network.

NAME	USER RELATED DATA
FROM	AP-400
TO	PROCESS DATA 2.4
FORMAT	RS232 SERIAL PORT COMMUNICATION
DESCRIPTION	Include response of requester information related to the user.

NAME	NETWORK RELATED DATA
FROM	AP-400
TO	PROCESS DATA 2.4
FORMAT	RS232 SERIAL PORT COMMUNICATION
DESCRIPTION	Include response of requester information related to the network.

NAME	MAC REQUEST FROM PIC
FROM	PREPARE REQUEST 2.1
TO	GET MAC ADDRESSES 2.2.1
FORMAT	ELECTRICAL SIGNAL
DESCRIPTION	Include requester MAC address information.

NAME	MAC REQUEST FROM AP-400
FROM	GET MAC ADDRESSES 2.2.1
TO	AP-400
FORMAT	RS232 SERIAL PORT COMMUNICATION
DESCRIPTION	Include requester MAC address information.

NAME	STATUS REQUEST FROM PIC
FROM	PREPARE REQUEST 2.1
TO	GET CONNECTION STATUS 2.2.2
FORMAT	ELECTRICAL SIGNAL
DESCRIPTION	Include requester connection status information.

NAME	STATUS REQUEST FROM AP-400
FROM	GET CONNECTION STATUS 2.2.2
TO	AP-400
FORMAT	RS232 SERIAL PORT COMMUNICATION
DESCRIPTION	Include requester connection status information.

NAME	IP REQUEST FROM PIC
FROM	PREPARE REQUEST 2.1
TO	GET IP ADDRESSES 2.2.3
FORMAT	ELECTRICAL SIGNAL
DESCRIPTION	Include requester IP address information.

NAME	IP REQUEST FROM AP-400
FROM	GET IP ADDRESSES 2.2.3
TO	AP-400
FORMAT	RS232 SERIAL PORT COMMUNICATION
DESCRIPTION	Include requester IP address information.

NAME	NAME REQUEST FROM PIC
FROM	PREPARE REQUEST 2.1
TO	GET COMPUTER NAMES 2.2.4
FORMAT	ELECTRICAL SIGNAL
DESCRIPTION	Include requester computer name information.

NAME	NAME REQUEST FROM AP-400
FROM	GET COMPUTER NAMES 2.2.4
TO	AP-400
FORMAT	RS232 SERIAL PORT COMMUNICATION
DESCRIPTION	Include requester computer name information.

NAME	LAST DAY STATISTICS REQUEST FROM PIC
FROM	PREPARE REQUEST 2.1
TO	GET LAST DAY STATISTICS 2.2.5
FORMAT	ELECTRICAL SIGNAL
DESCRIPTION	Include requester last day statistics information.

NAME	LAST DAY STATISTICS REQUEST FROM AP-400
FROM	GET LAST DAY STATISTICS 2.2.5
TO	AP-400
FORMAT	RS232 SERIAL PORT COMMUNICATION
DESCRIPTION	Include requester last day statistics information.

NAME	SPEED REQUEST FROM PIC
FROM	PREPARE REQUEST 2.1
TO	GET UPLOAD DOWNLOAD SPEED 2.2.6
FORMAT	ELECTRICAL SIGNAL
DESCRIPTION	Include requester upload download speed information.

NAME	SPEED REQUEST FROM AP-400
FROM	GET UPLOAD DOWNLOAD SPEED 2.2.6
TO	AP-400
FORMAT	RS232 SERIAL PORT COMMUNICATION
DESCRIPTION	Include requester upload download speed information.

NAME	SSID REQUEST FROM PIC
FROM	PREPARE REQUEST 2.1
TO	GET SSID 2.3.1
FORMAT	ELECTRICAL SIGNAL
DESCRIPTION	Include requester SSID information.

NAME	SSID REQUEST FROM AP-400
FROM	GET SSID 2.3.1
TO	AP-400
FORMAT	RS232 SERIAL PORT COMMUNICATION
DESCRIPTION	Include requester SSID information.

NAME	NETWORK TYPE REQUEST FROM PIC
FROM	PREPARE REQUEST 2.1
TO	GET NETWORK TYPE 2.3.2
FORMAT	ELECTRICAL SIGNAL
DESCRIPTION	Include requester network type information.

NAME	NETWORK TYPE REQUEST FROM AP-400
FROM	GET NETWORK TYPE 2.3.2
TO	AP-400
FORMAT	RS232 SERIAL PORT COMMUNICATION
DESCRIPTION	Include requester network type information.

NAME	DOWNLOAD SPEED REQUEST FROM PIC
FROM	PREPARE REQUEST 2.1
TO	GET DOWNLOAD SPEED 2.3.3
FORMAT	ELECTRICAL SIGNAL
DESCRIPTION	Include requester download speed information.

NAME	DOWNLOAD SPEED REQUEST FROM AP-400
FROM	GET DOWNLOAD SPEED 2.3.3
TO	AP-400
FORMAT	RS232 SERIAL PORT COMMUNICATION
DESCRIPTION	Include requester download speed information.

NAME	UPLOAD SPEED REQUEST FROM PIC
FROM	PREPARE REQUEST 2.1
TO	GET UPLOAD SPEED 2.3.4
FORMAT	ELECTRICAL SIGNAL
DESCRIPTION	Include requester upload speed information.

3.4STATE TRANSITION DIAGRAM

Menu structure (Figure 3.8):

When the user initiates the device, a welcoming message appears while the embedded operating system in AP-400 is loading. The message will be determined later on and will be open to changes.

- User initiates the device
- Welcome message appears
- Main menu:

User will select one of the functions/menus below:

- Users:

The device user will select a machine with up/down buttons, will confirm with enter button to view information about the selected device.

IP address, MAC address, device name and current connection speeds are some of data we plan to show on LCD.

Back button will exit to user menu and another back button will move the user to the main menu.

- Networks:

If selected, the menu will display current networks in the environment.

- Statistics:

The menu will show some statistics about current network in which the device is connected.

User will be able to see some information like how many people are connected to the device, their average connection speeds, for how many hours/days the network is active.

- Options menu:

User will select a submenu to get in the selected device options, will confirm the submenu with enter.

- Sound menu: If selected, sound effects for the device will be enabled. If not, sound will be disabled.

- Brightness: Brightness of the LCD will be adjusted using the menu. User will confirm selected brightness with enter button or will cancel it by just exiting the menu without confirming.

- Language menu: User will be able to change the language of LCD interface by using the menu. Currently, we plan to use two languages: Turkish and English.

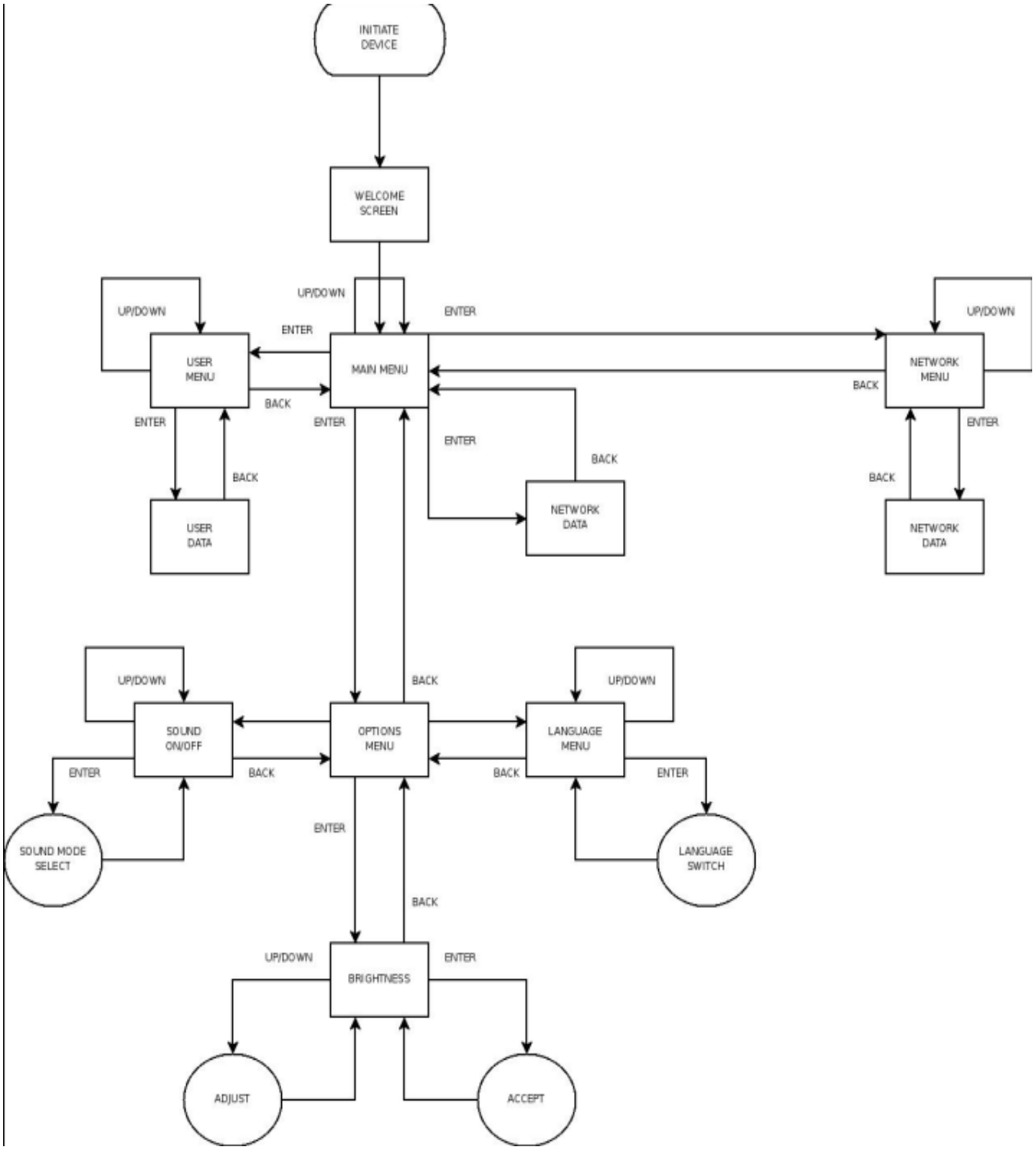


Figure 3.8

4. TESTING

During the development of the project, a continuous testing approach will be applied to ensure the device and software quality. Since the project steps are dependent of the steps beforehand to be completed, we will need to be sure that, for all phases of the project, the previous phases work fault free. If we did not have a testing procedure for all steps, next steps would include the errors of previous steps and at a time, we would not be able to continue the project and it would halt. In order to prevent that, we made a well-formed testing procedure for both hardware and software parts in all phases of the project.

Testing procedure is composed of two steps, namely hardware testing and software testing.

4.1 HARDWARE TESTING

The project is not a software-only project and includes hardware. Thus, we have a series of well-formed hardware testing procedures.

Hardware testing is composed of two parts which are comprehensively explained below.

4.1.1 Device Requirements Testing

Device requirement testing includes the testing of the parts necessary for the proper functionality of the new module.

- The module and the original circuit will be getting the necessary power from the already existing power source of AP-400. During the steps where we include a new device element to the system, we will check for the possible power shortage.
- We will need to add the other circuit elements (resistor, capacitor, MAX3378 etc.) to the system for the LCD module processor to function properly. These elements should not have any defects to prevent them from functioning properly.

Cumulative working of the circuit elements with other circuit elements may also cause unforeseeable results. Elements will be tested not only before the addition to the module, but also after the addition to the circuit, in order to be able to detect such kind of failures.

4.1.2 Device Testing

Device testing includes the testing procedure of checking the functionalities main devices; LCD, AP-400 processor, module processor and buttons, specifically.

4.1.2.1 Liquid Crystal Display (LCD)

LCD is the main-role device of the project. Thus, testing of LCD device is an important concept for the whole project.

First testing approach for the LCD is for the device functionality. Device should work without any failure.

As LCD will be used for in nearly all phases of the project, we guess that, in case of a failure of the device, we will easily be able to identify it.

Second testing approach for LCD is for the characters and graphics that will be shown on the device.

We will check for all characters that might occur on the screen and check for graphics as we show them on the screen. For both the characters and graphics, there should not be any distortion on the screen.

LCD and its view will be tested thoroughly since it is the main purpose of this project.

4.1.2.2 AP-400 Processor

Main processor testing primarily includes the testing of the old system functionalities after the addition of the new module.

For the addition of the new module to the system, we will need to change/add some executables to the old system to obtain the data requested. We will check the system, whenever we want to acquire new kinds of information from the system, on how other functionalities and the performance will be affected.

4.1.2.3 Module Processor

Even a circuit element should be tested before adding into a system and module processor will be, too. Since the device will be used for all the steps of the development, we will, most probably, be able to detect any hardware failure as soon as it occurs.

4.1.2.4 Buttons

Buttons should be tested in order to have right inputs to the user.

User clicks should be taken as appropriate inputs (click number should increase once every 0.5-1 seconds, not 0.05 seconds for example).

Button functionality will be tested as the project progresses. We will most probably be able to detect any malfunctioning buttons easily.

4.2SOFTWARE TESTING

4.2.1 Software Testing for the Main Processor

The software for communicating with new processor, having directions from module processor (about what to send to the user), which process the information (preparing the requested data) and send it back, will be tested for functionality, reliability and efficiency.

During gathering data from main processor, we do not expect any difficulties, since the procedure will be quite easy such that some shell scripts is enough to gather the necessary information fom the device. We will be able to test our scripts as we write them on the embedded Linux.

We will need some testing about the software to transmit requests from module processor and to transmit prepared data to the module processor through RS232 port. When the new module is prepared, we will be able to test the transmission protocols.

4.2.2 Software Testing for the Module Processor

The software to communicate with main processor, LCD and buttons will be tested under this category.

During the transmission of a button click to module processor, most probably, the buttons will send more than one button clicks to the processor. Filtering of these false clicks will be done on the software level. These clicks will be tested thoroughly.

After checking proper input to the device, we will need to send proper requests to main processor. As the project progresses, as we write the code about these requests, we will be checking them.

After the creation of request and acquisition of the response, the device will send the information to the LCD. We will be testing LCD software as we write the code. We plan to use some software about the LCD communication and viewing part of the project.

Another testing mechanism for module processor is that, we should check our codes before and after adding them to the system for any inconsistencies with the already existing ones.

Finally, software testing includes checking the accuracy of the information gathered from main processor and the data shown on LCD. We cannot show wrong IP addresses to the users, for example.

5. CODING STANDARTS

5.1 FILE ORGANIZATION

File names must include the content of the file clearly and in the file name the first letter of the all words must be capital.

5.2 VARIABLE NAMES

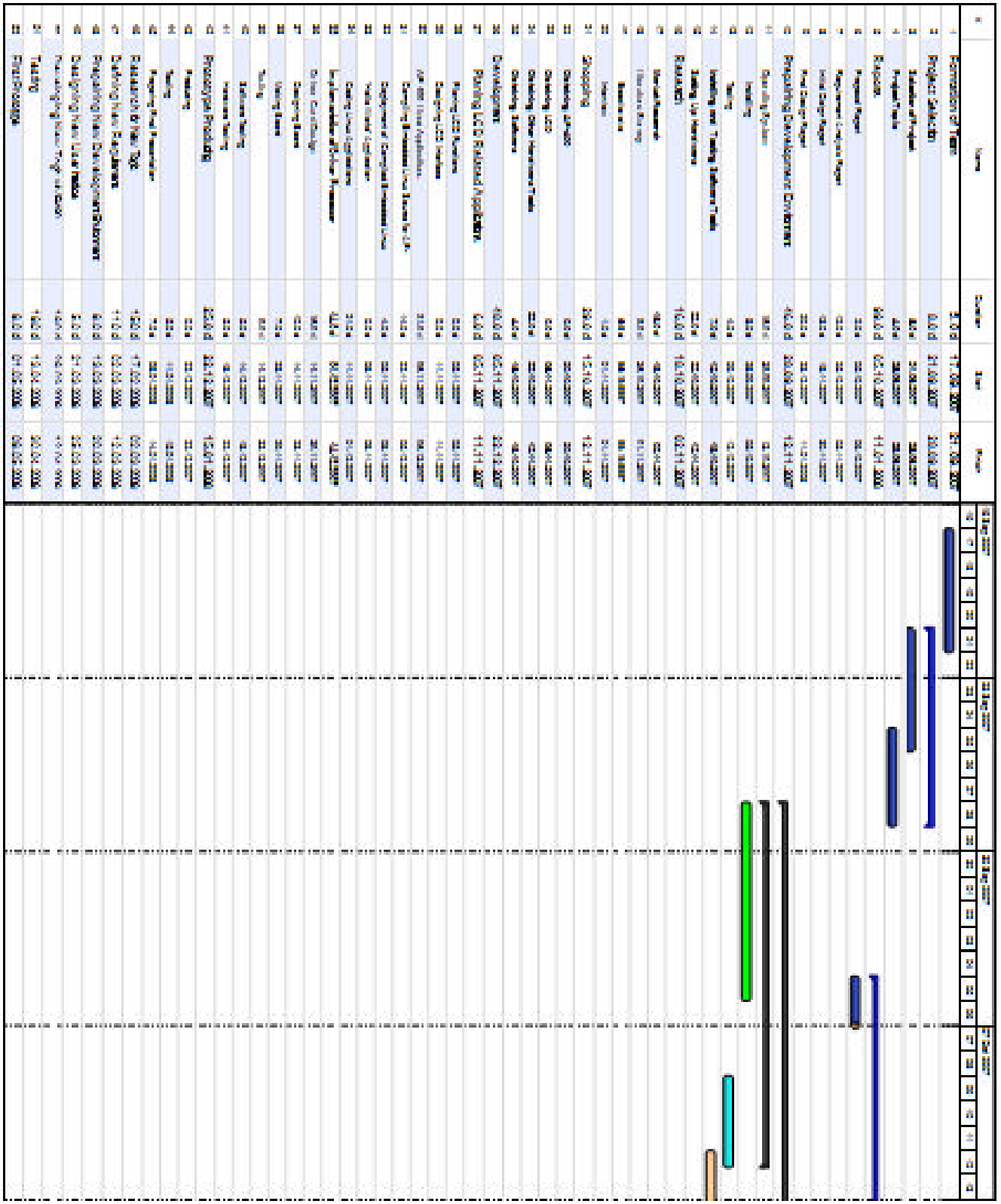
Variables names should include the content of the variable. Types of variables should be also indicated. For example; pointers must start with letter ‘p’ and a char variable must start with ‘c’. Constant variables should have capital letters.

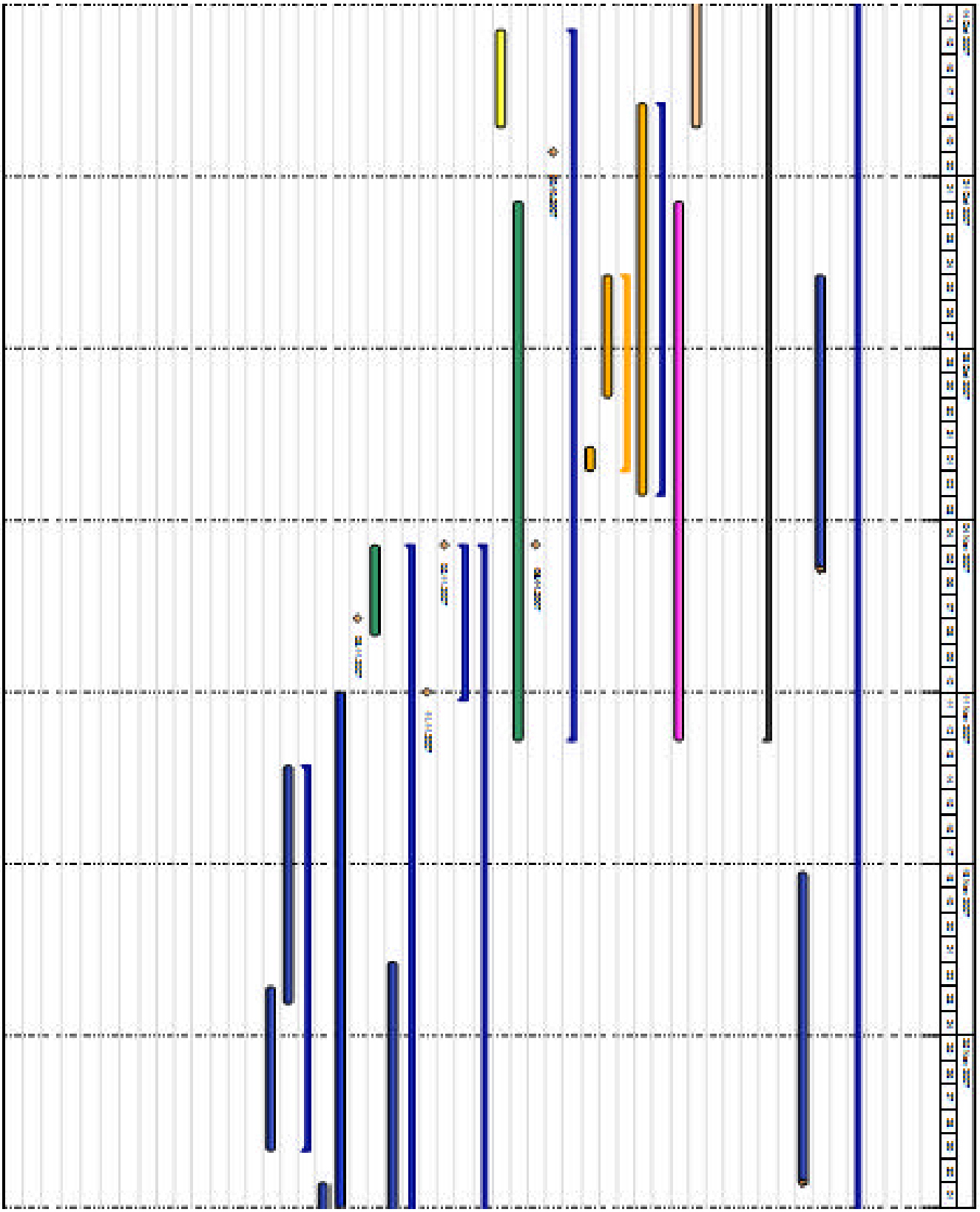
5.3 MODULE COMMENTS

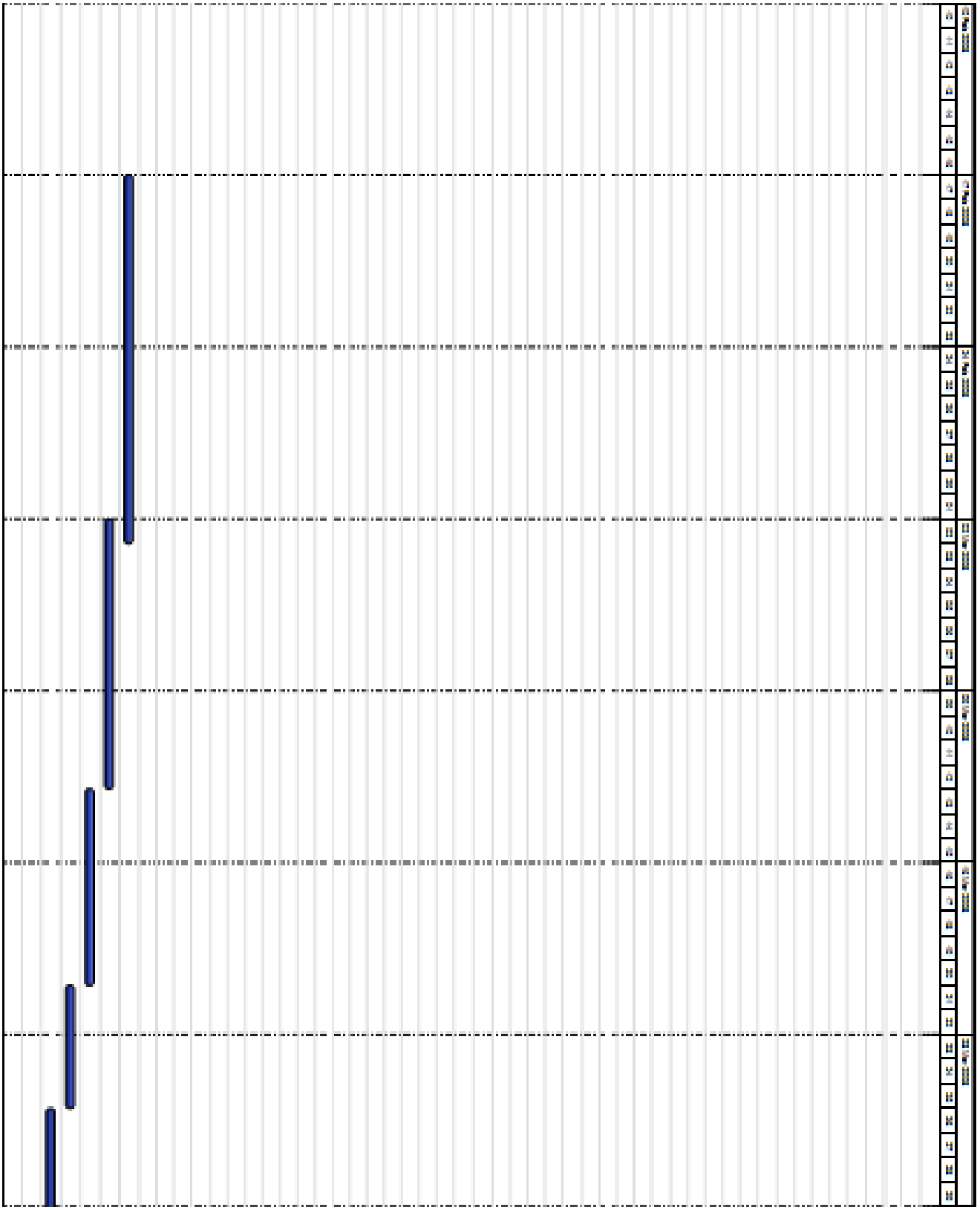
Module comments will provide us to get information about module role. The structure of our module comments is mentioned below:

```
/*
File: Filename
Description: The role of module in the all system is indicated here.
Coder: coder name
Written Date: 03/12/2007
Notes: If there is any note about the code, write here.
*/
```

6. GANTT CHART







7. CONCLUSION

During the initial design process, we overviewed all parts of the project and we gained almost all the information that we need. In this report, we gave information about our initial design process. In the introduction part, we mentioned problem definition in detailed; also we determined design constraints and the design requirements for an embedded software and hardware development in our project. How the design of user interface will be was figured in user interface part and some examples for submenus were given. Architectural design of the system and its components were dealt with in next part. The general structure of driver board also exists in this part. Data flow diagrams (DFD) are revised and drawn with more depth (up to third level). Data dictionary of DFDs are also renewed according to new version of DFDs. A general state transition diagram with menu structure is the last part of third section.

Testing is an important issue for such a software-hardware project. Therefore, a section is composed of software and hardware testing of each component in the project. Coding and file naming must be in a standard in order to easy readability and debugging. At fifth section, these standards were explained. Gantt chart of the whole process is also added to the report as the sixth part.

8. REFERENCES

Booch G., Rumbaugh J. & Jacobson I. (1999). *The Unified Modeling Language User Guide.* Addison – Wesley

Dogru, A. (2006). *Component Oriented Software Engineering.* The Atlas Publications.

Pressman Roger S.(2005). *Software Engineering: A practitioner's Approach.* Mc Graw Hill.