ON-BOARD DIAGNOSTICS SCANNER

Supervisor: Dr. Samer Arandi

Prepared by: Ahmad Saqfalhait & Morad Abushamma

Hardware graduation project

Outline

- Background
- High Level View
- Protocols
- User Interface
- Storage Unit
- Hardware Implementation
- Similar System
- Limitations
- Standers
- PLC Project

Background

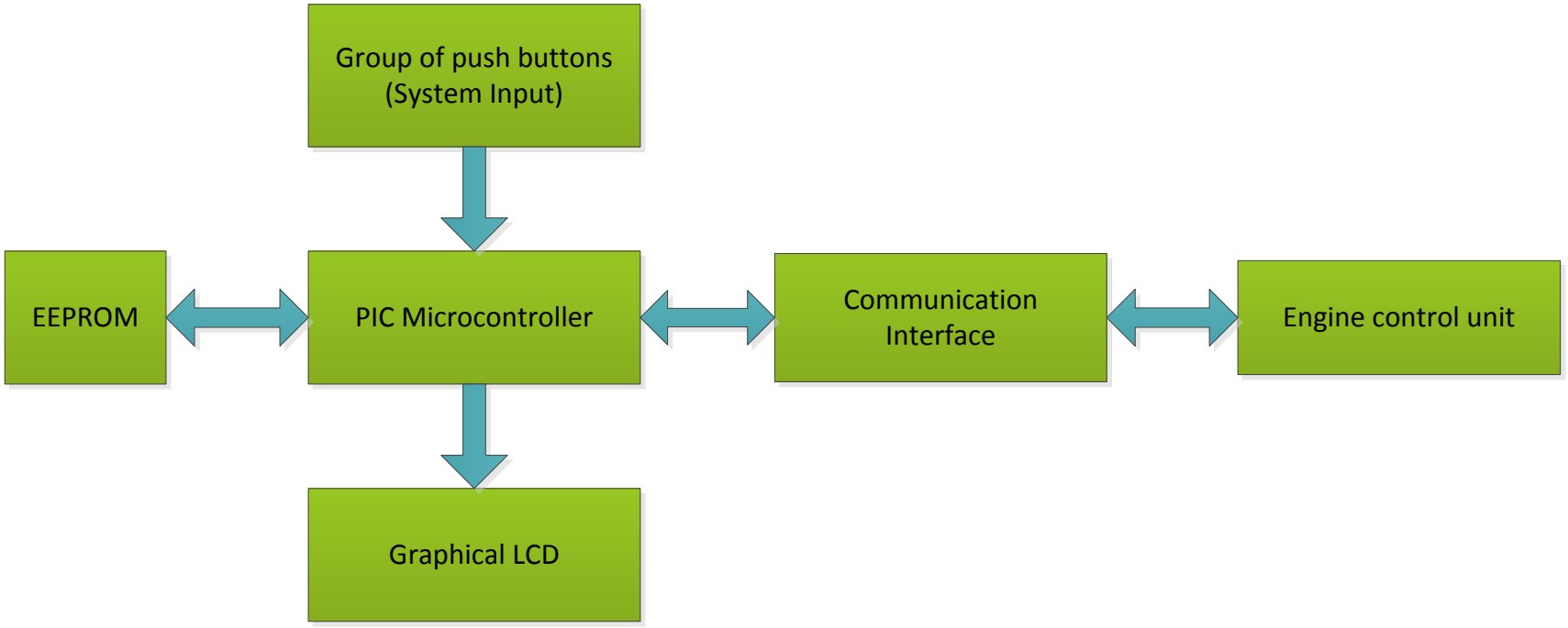
Nowadays cars are very essential element in people`s life, and it`s maintenance has a great consideration. In the early 1980s a new system that helps in car maintenance has been introduced which is on-board diagnostics system “OBD” a self-diagnostic and reporting system, It gives the repair technician and the car owner the ability to access the health information of a vehicle, sensors and many parameters, this system has been evolved and now it became a mandatory requirement in car industry.

Introduction

OBD scanner standalone device that will give the user the capability to request diagnostic data form the car through an interactive interface based on appropriate communication interface, storage unit and processing unit.

OBD scanner is OBD-II standard compliant so it supports almost all cars manufactured after 1996 and this device has three main functions: **view current and Freeze frame data, Read and clear generic and manufacturer specific diagnostic trouble codes** and **Retrieve vehicle information.**

High level view



Protocols and communication interface

A group of protocols should be implemented in the OBD scanner device in order to support all OBD-II compliant cars. Each protocol has physical, data link and application layer specifications.

The protocols are:

- ISO15765-4 (CAN-BUS) (mandatory for 2008+ cars)
- ISO14230-4 (KWP2000) (Very common protocol for 2003+)
- ISO9141-2 (ISO) (the oldest protocol)
- SAE J1850 (VPW) (GM vehicles)
- SAE J1850 (PWM) (Ford vehicles)

OBD II Cable

1	5	(Signal Ground)
2	4	(Chassis Ground)
3	6	(CAN High (J-2284)
4	7	(ISO 9141-2 K Line)
5	14	(CAN Low J-2284)
6	10	(J1850 Bus-)
7	2	(J1850 Bus+)
8	15	(ISO 9141-2 L Line)
9	16	(Battery Power)

ISO 9141

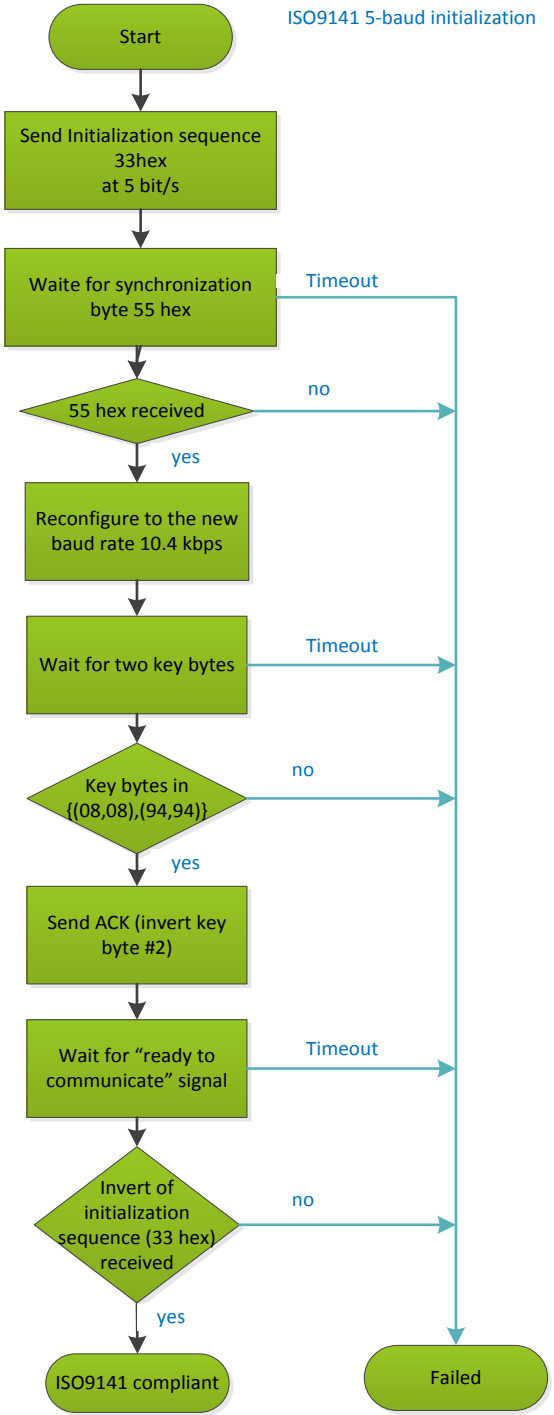
Message Format

	Header bytes			Data bytes							Tail
	Type	Target Address	Source Address	Byte #1	Byte #2	Byte #3	Byte #4	Byte #5	Byte #6	Byte #7	CS
Request	68	6A	F1	7 Data bytes (max)							
Response	48	6B	ECU Physical Address	7 Data bytes (max)							

Frame structure

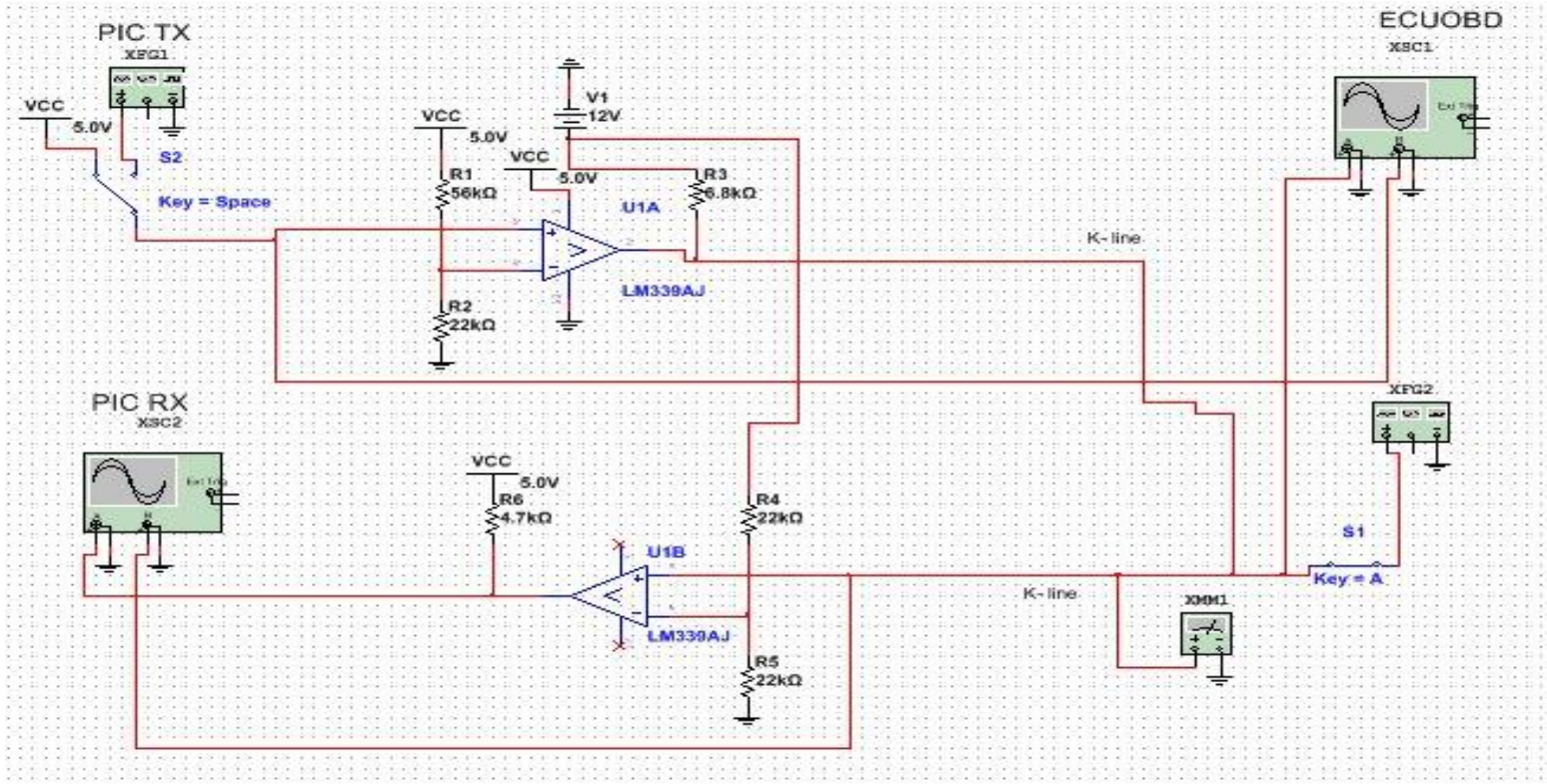
Start Bit "0"	bit 0	bit 1	bit 2	bit 3	bit 4	bit 5	bit 6	bit 7	Stop Bit "1"

ISO 9141 5-baud Initialization



ISO 9141 Interface

ISO9141 is an asynchronous serial data protocol with 10.4 baud rate, similar to RS232 in timing and frame structure and differs in signal levels with 12 V as logic 1 and 0 V as logic 0, one bidirectional line used to transmit and receive data known as K-Line.



ISO14230 (KWP2000)

Message Format

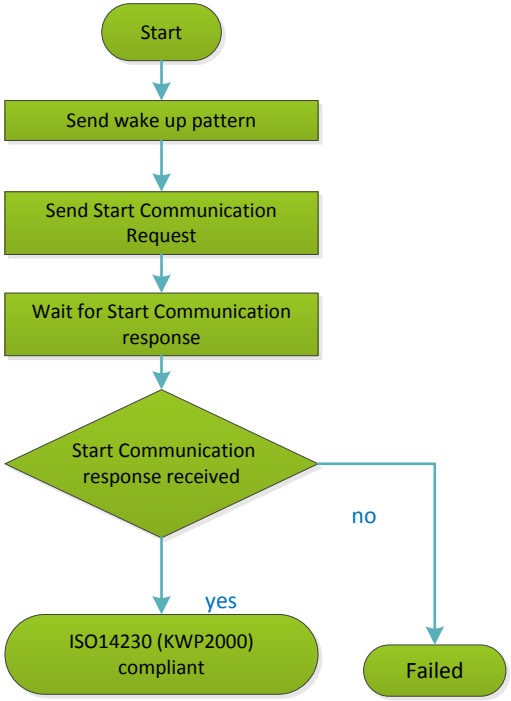
	Header bytes			Data bytes							Tail
	Format Byte	Target Address	Source Address	Byte #1	Byte #2	Byte #3	Byte #4	Byte #5	Byte #6	Byte #7	CS
Request	11LL LLLL	33	F1	7 Data bytes (max)							
Response	10LL LLLL	F1	ECU Physical Address	7 Data bytes (max)							

Frame structure

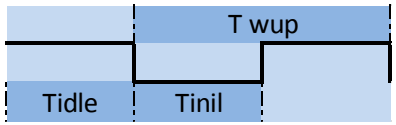
Start Bit "0"	bit 0	bit 1	bit 2	bit 3	bit 4	bit 5	bit 6	bit 7	Stop Bit "1"

KWP2000 5-baud initialization and hardware interface same ISO 9141

KWP2000 Fast Initialization



Wake up pattern



Tinil	Twup
25	50

Start communication Request

Header bytes			Data bytes	Tail
Format Byte	Target Address	Source Address	Byte #1	CS
C1	33	F1	81	66

Start communication positive response

Header bytes			Data bytes			Tail
Format Byte	Target Address	Source Address	Byte #1	Byte #2	Byte #3	CS
83	F1	10	C1	YY	ZZ	CS

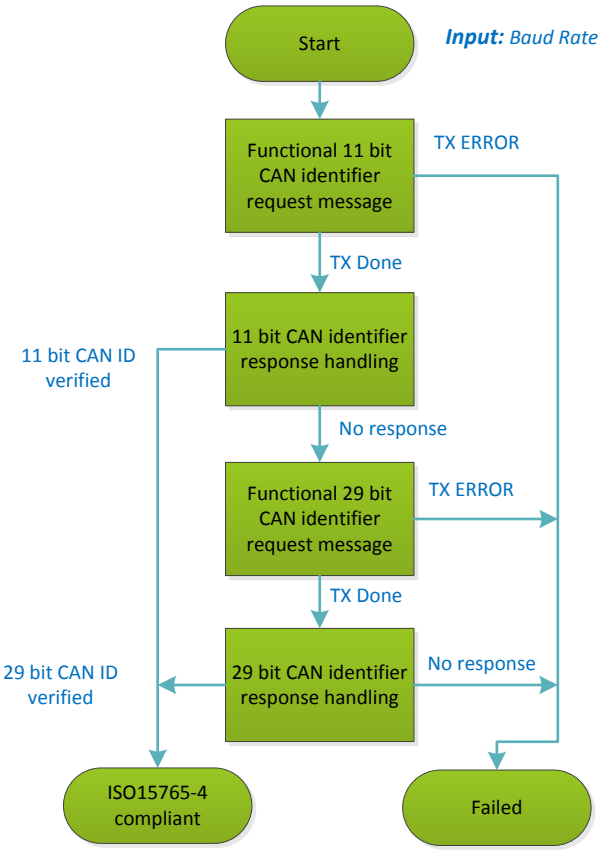
ISO15765 (CAN-BUS)

Versions

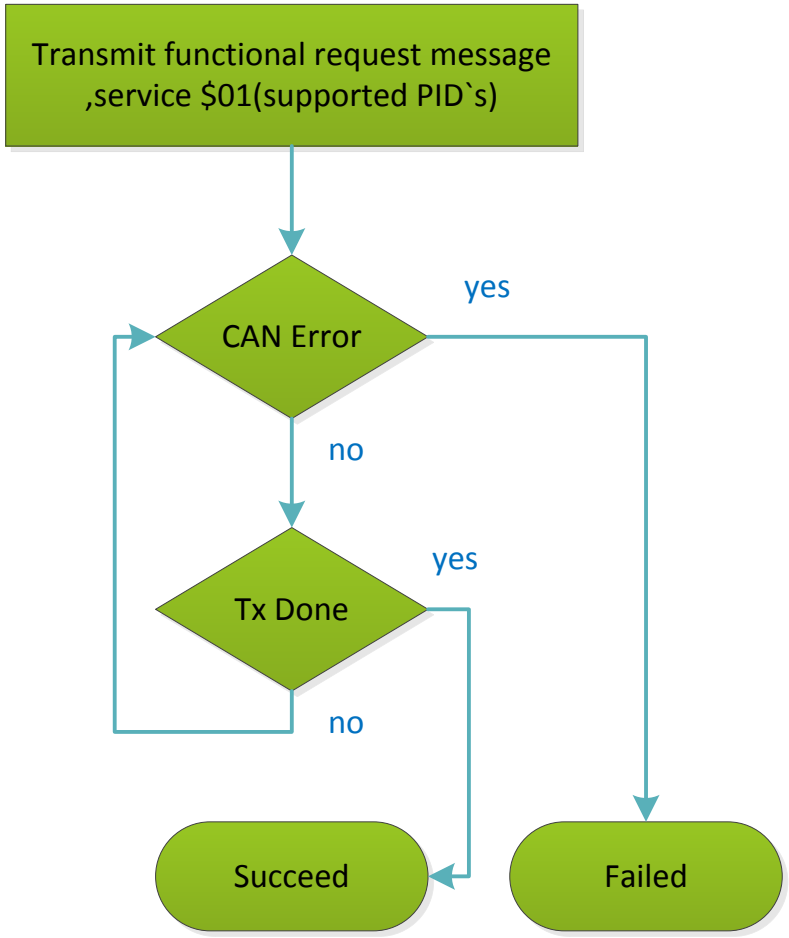
Version NO.	Identifier length	Baud rate
1	11 bit	500 Kbaud
2	29 bit	500 Kbaud
3	11 bit	250 Kbaud
4	29 bit	250 Kbaud

CAN Protocol is mandatory for 2008+ cars

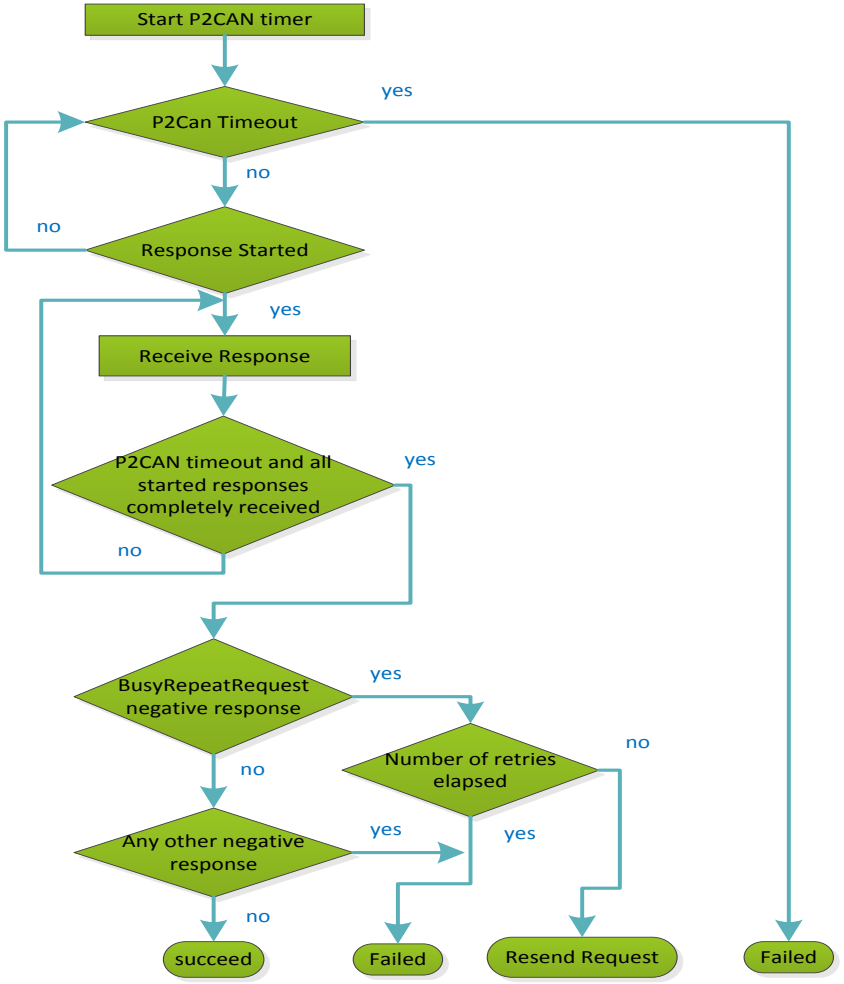
CAN Initialization



CAN REQUEST MESSAGE TRANSMISSION

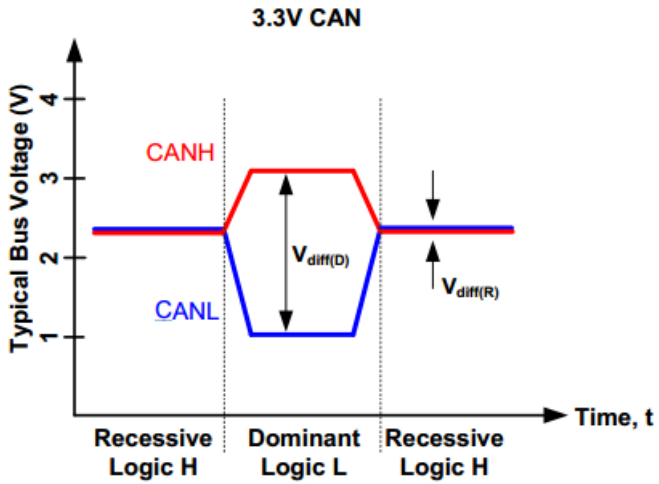


CAN RESPONSE HANDLING

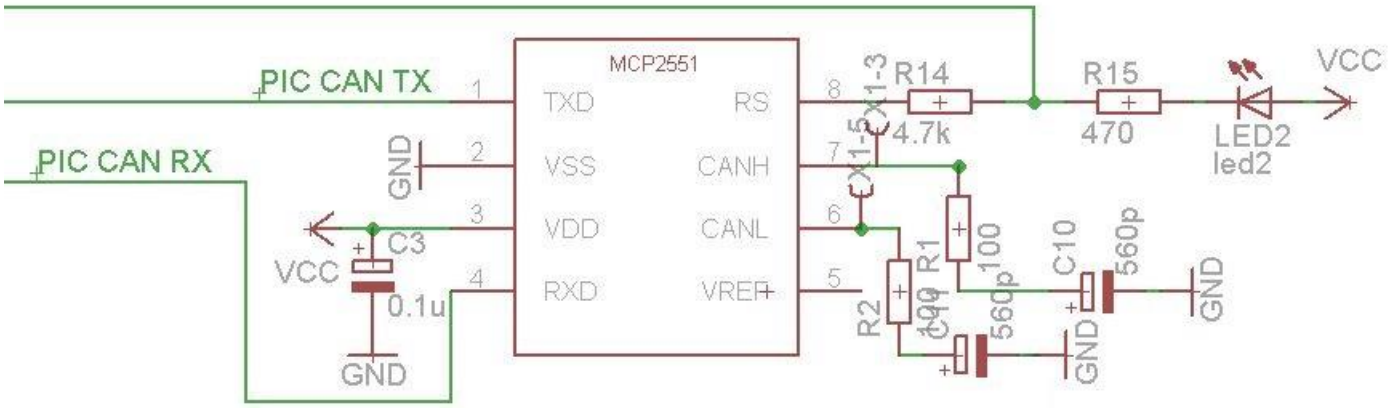


CAN Signaling specifications

- CAN Bus has two line CANH and CANL, logic is determined by both line level:
- Logic 1: Recessive or idle bus state: CANH and CANL signals are not driven
- Logic 0: Dominant or active bus state: CANH driven high while CANL driven low
- CANH signal voltage level: 3.5V
- CANL signal voltage level: 1.5V



Can Interface



Data inquiry

Message Format

Mode List

Mode (hex)	Description
1	Show current data
2	Show freeze frame data
3	Show stored Diagnostic Trouble Codes
4	Clear Diagnostic Trouble Codes and stored values
5	Test results, oxygen sensor monitoring (non CAN only)
6	Test results, other component/system monitoring (Test results, oxygen sensor monitoring for CAN only)
7	Show pending Diagnostic Trouble Codes (detected during current or last driving cycle)
8	Control operation of on-board component/system
9	Request vehicle information
0A	Permanent Diagnostic Trouble Codes (DTCs) (Cleared DTCs)

MODE 1 PID List

PID (hex)	bytes returned	Description
0	4	PIDs supported [01 - 20]
1	4	Monitor status since DTCs cleared. (Includes malfunction indicator lamp (MIL) status and number of DTCs.)
2	2	Freeze DTC
3	2	Fuel system status
4	1	Calculated engine load value
5	1	Engine coolant temperature
6	1	Short term fuel % trim—Bank 1
7	1	Long term fuel % trim—Bank 1
8	1	Short term fuel % trim—Bank 2
9	1	Long term fuel % trim—Bank 2
0A	1	Fuel pressure
0B	1	Intake manifold absolute pressure
0C	2	Engine RPM
0D	1	Vehicle speed
0E	1	Timing advance
0F	1	Intake air temperature
10	2	MAF air flow rate
11	1	Throttle position
12	1	Commanded secondary air status
13	1	Oxygen sensors present

OBD Error codes

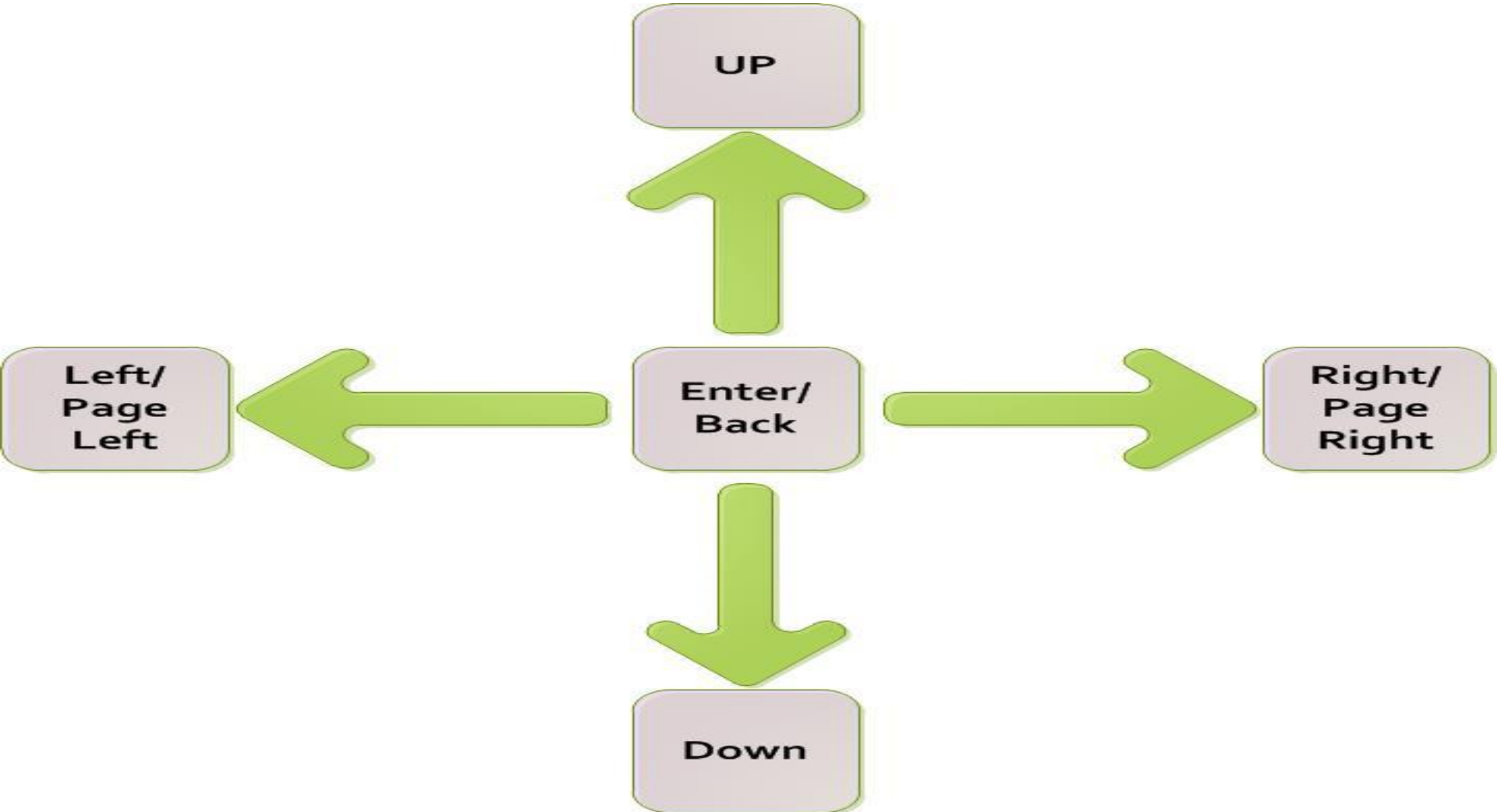
Code	Description
P1102	O2S Heating Circuit Bank 1 Sensor 1 Voltage Too Low/Air Leak
P1105	O2S Heating Circuit Bank 1 Sensor 2 Short To Positive
P1107	O2S Heating Circuit Bank 2 Sensor 1 Short To Positive
P1110	O2S Heating Circuit Bank 2 Sensor 2 Short To Positive
P1113	O2S Sensor Heater Resistance Too High Bank 1 Sensor 1
P1115	O2S Sensor Heater Circuit Short To Ground Bank 1 Sensor 1
P1116	O2S Sensor Heater Circuit Open Bank 1 Sensor 1
P1117	O2S Sensor Heater Circuit Short To Ground Bank 1 Sensor 2
P1118	O2S Sensor Heater Circuit Open Bank 1 Sensor 2
P1127	Long Term Fuel Trim B1 System Too Rich
P1128	Long Term Fuel Trim B1 System Too Lean
P1129	Long Term Fuel Trim B2 System Too Rich
P1130	Long Term Fuel Trim B2 System Too Lean
P1136	Long Term Fuel Trim Add. Fuel B1 System Too Lean
P1137	Long Term Fuel Trim Add. Fuel B1 System Too Rich
P1138	Long Term Fuel Trim Add. Fuel B2 System Too Lean

User Interface

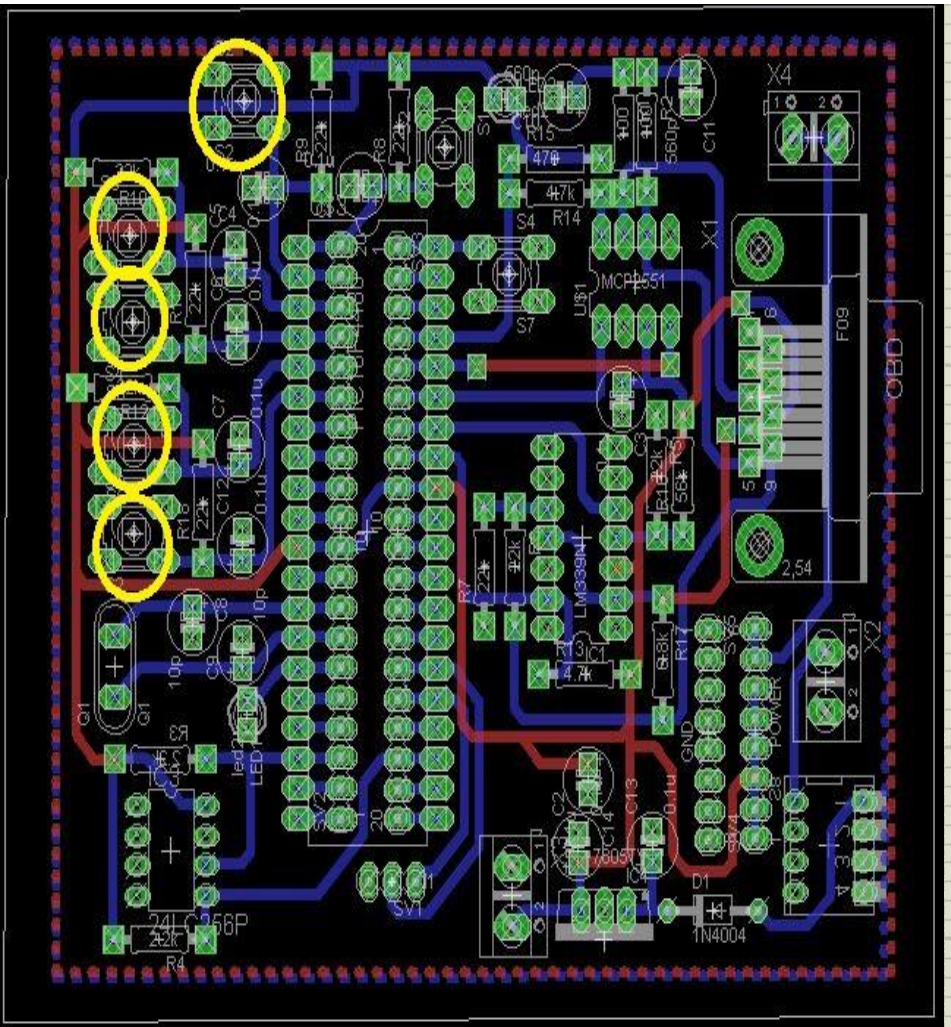
User interact with the device via:

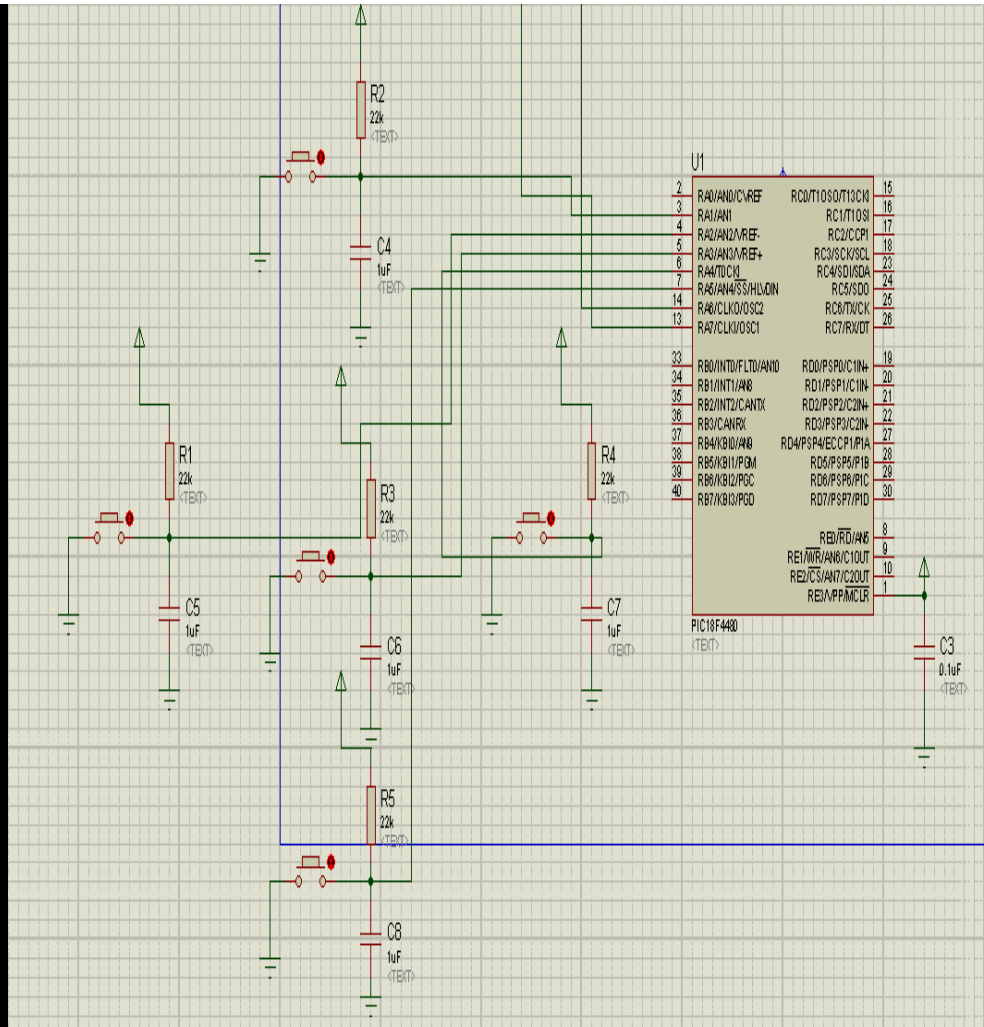
- Input: Push Button
- Output: GLCD

Input: push buttons



Input: push buttons – Proteus & PCB Design





Output: Graphical LCD

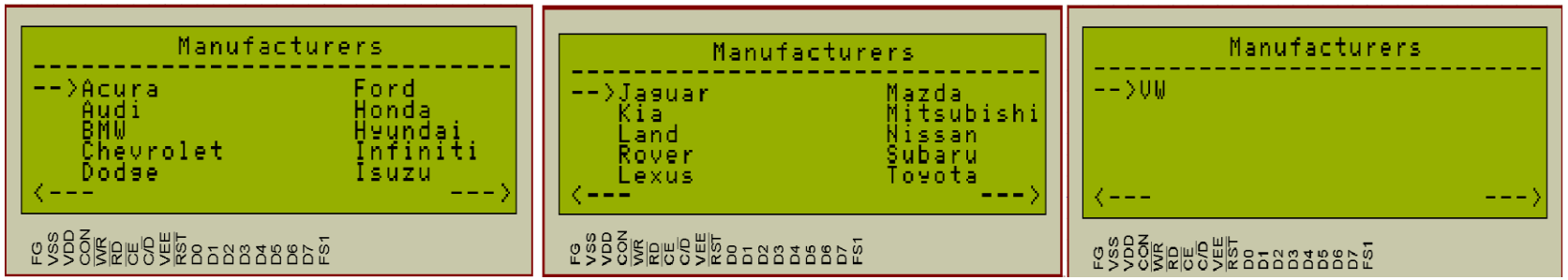
We have used GLCD (with t6963c controller) to present the output for the user.

Why this GLCD ?

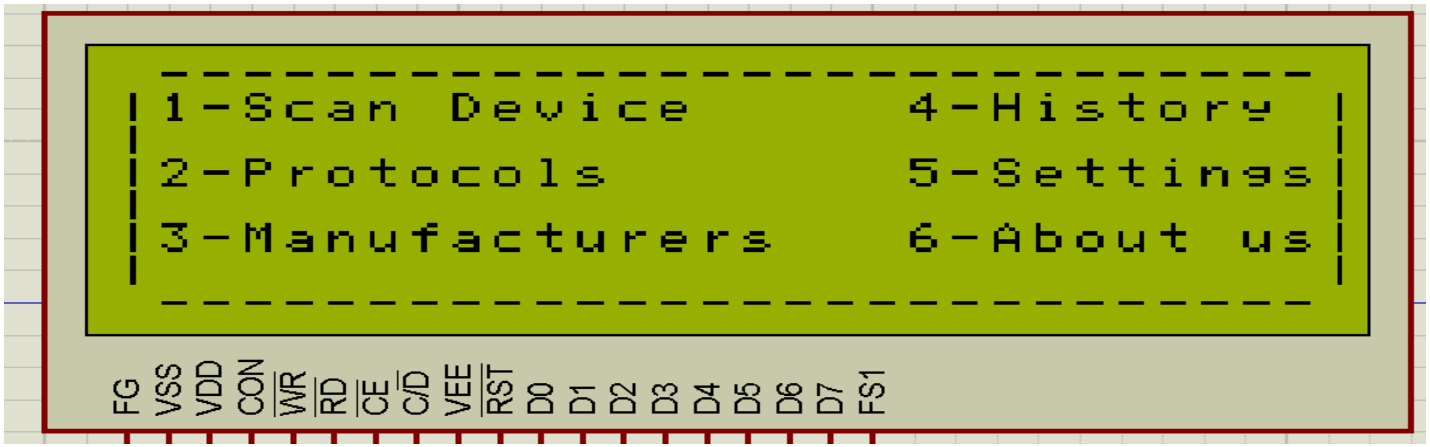
- It supports text and graphical modes
- It supports custom character – Arabic feature.
- Available Recourse.

Output: Graphical LCD - Software

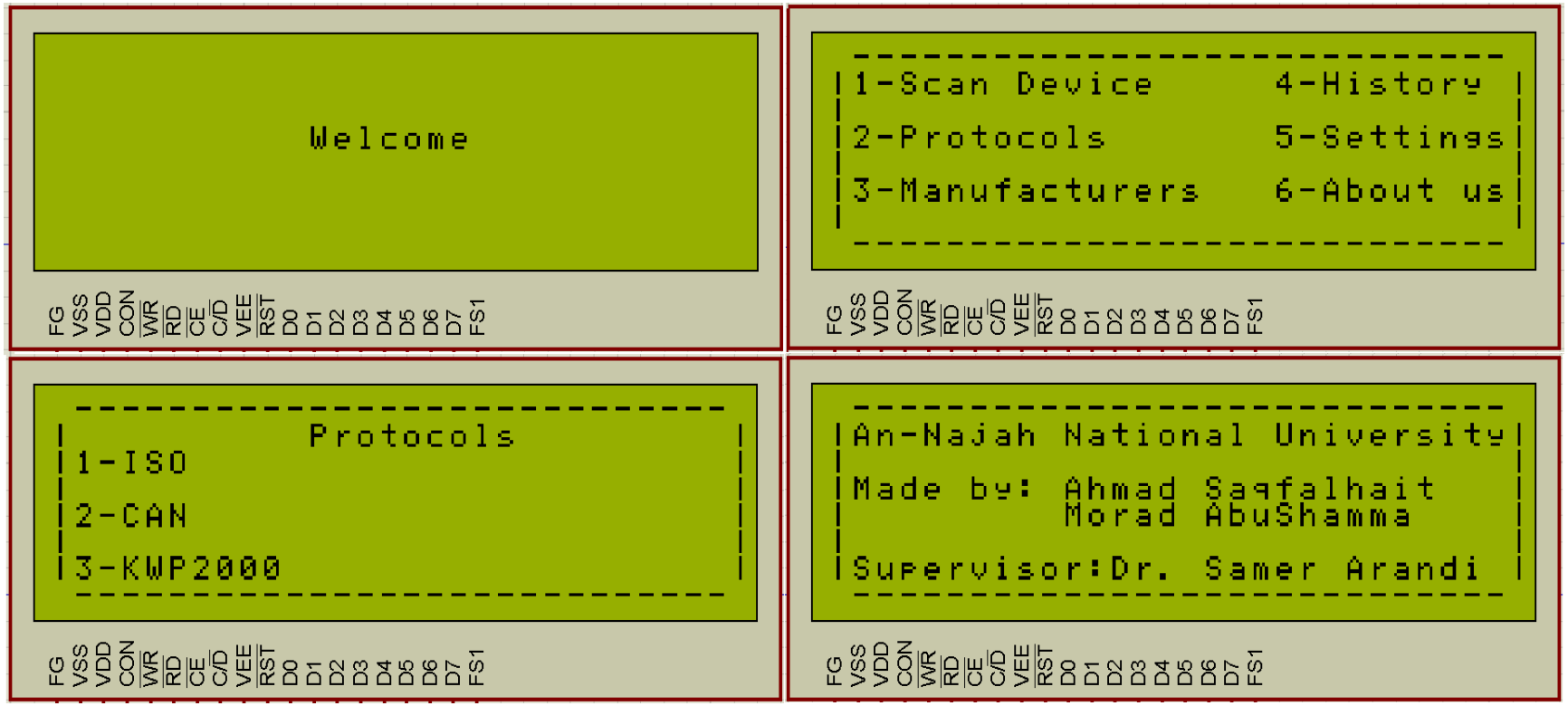
Paging and select functions.



Menu



Output: Graphical LCD - Software

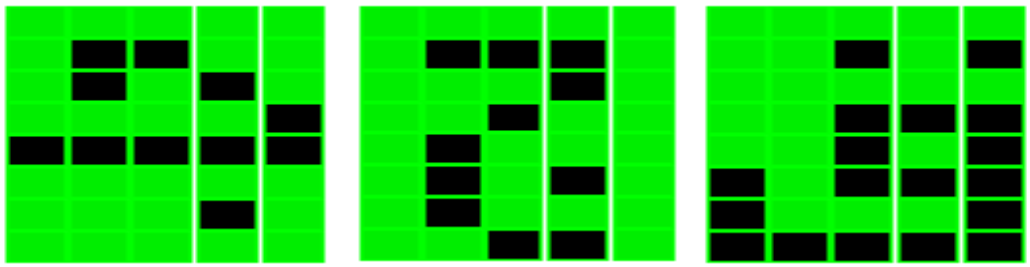


Output: Graphical LCD – Custom Char

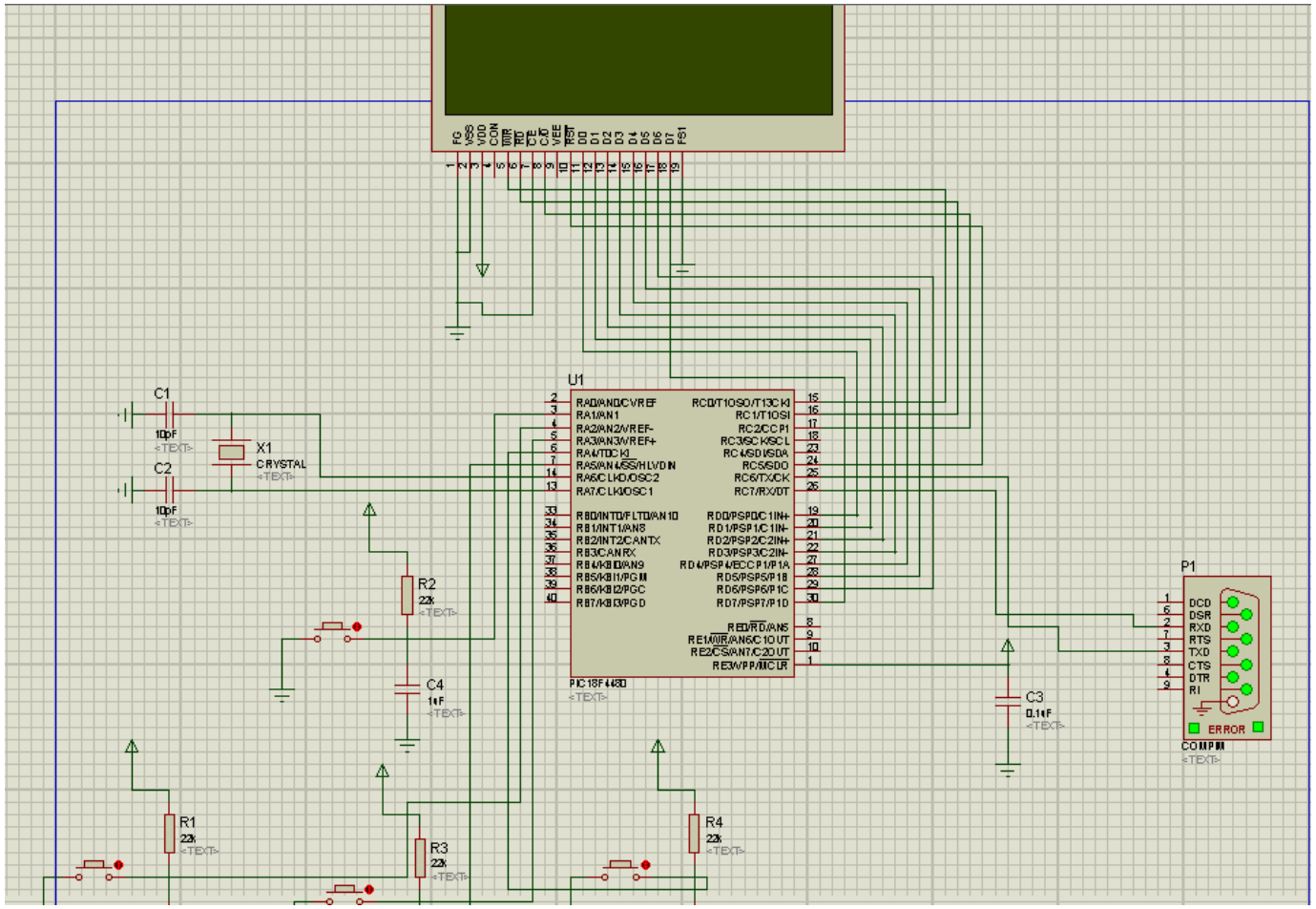
Process

- Generate array of hex values for every Arabic letter in all of its forms (connected and disconnected letter).
- Store characters' hex values at the EEPROM.
- Character's mapping address.

```
byte G_D[8] = {B00000,B01100,B01010,B00001,B11111,B00000,B00010,B00000};
byte G_C[8] = {B00000,B01110,B00010,B00100,B01000,B01010,B01000,B00110};
byte 8_D[8] = {B00000,B00101,B00000,B00111,B00101,B10111,B10001,B11111};
```



The whole system (input and output)



Storage unit EEPROM

- We used 24LC256 EEPROM.
- We stored 3 types of data:
  - Errors' codes.
  - Manufacturers.
  - History (last Scan result).
- Data stored at the Microcontroller's memory.
  - Main Sections Table

Sections	Address
Header	0Xxxxx– 0Xxxxx
Manufacturers and Codes - English	0Xxxxx– 0Xxxxx
Manufacturers and Codes - Arabic	0Xxxxx– 0Xxxxx
Description - English	0Xxxxx– 0Xxxxx
Description - Arabic	0Xxxxx– 0Xxxxx
History	0Xxxxx– 0Xxxxx

Storage unit EEPROM- Software

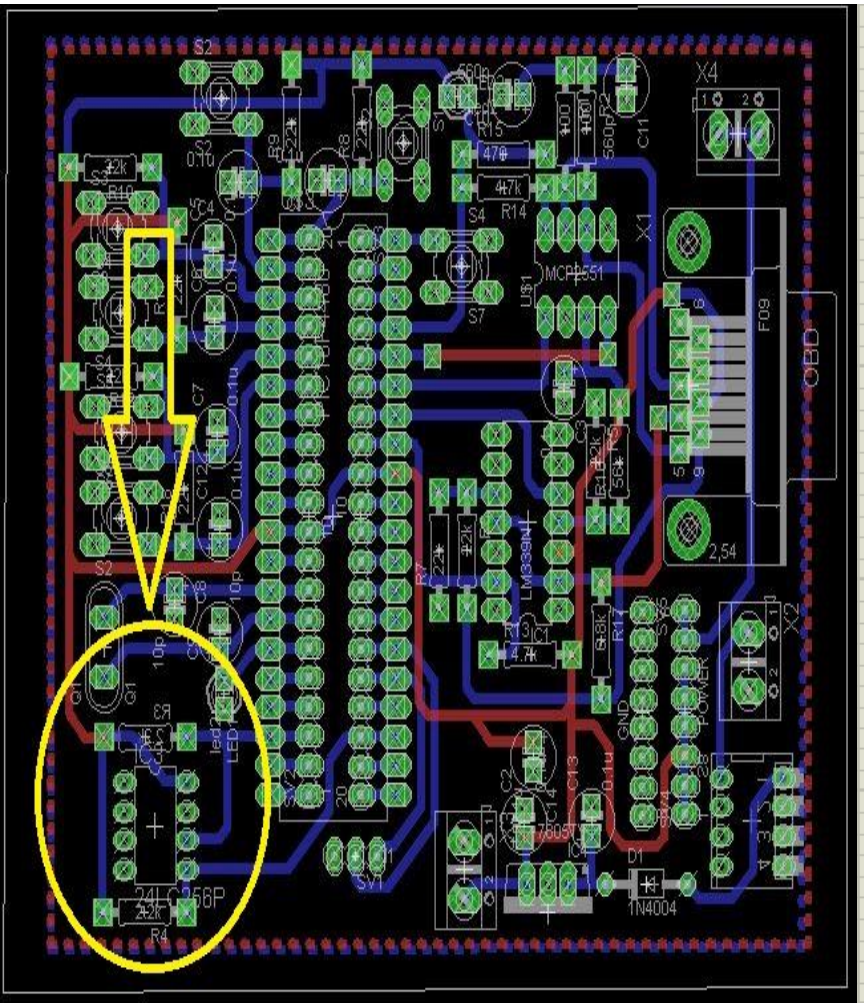
- Manufacturers Table:

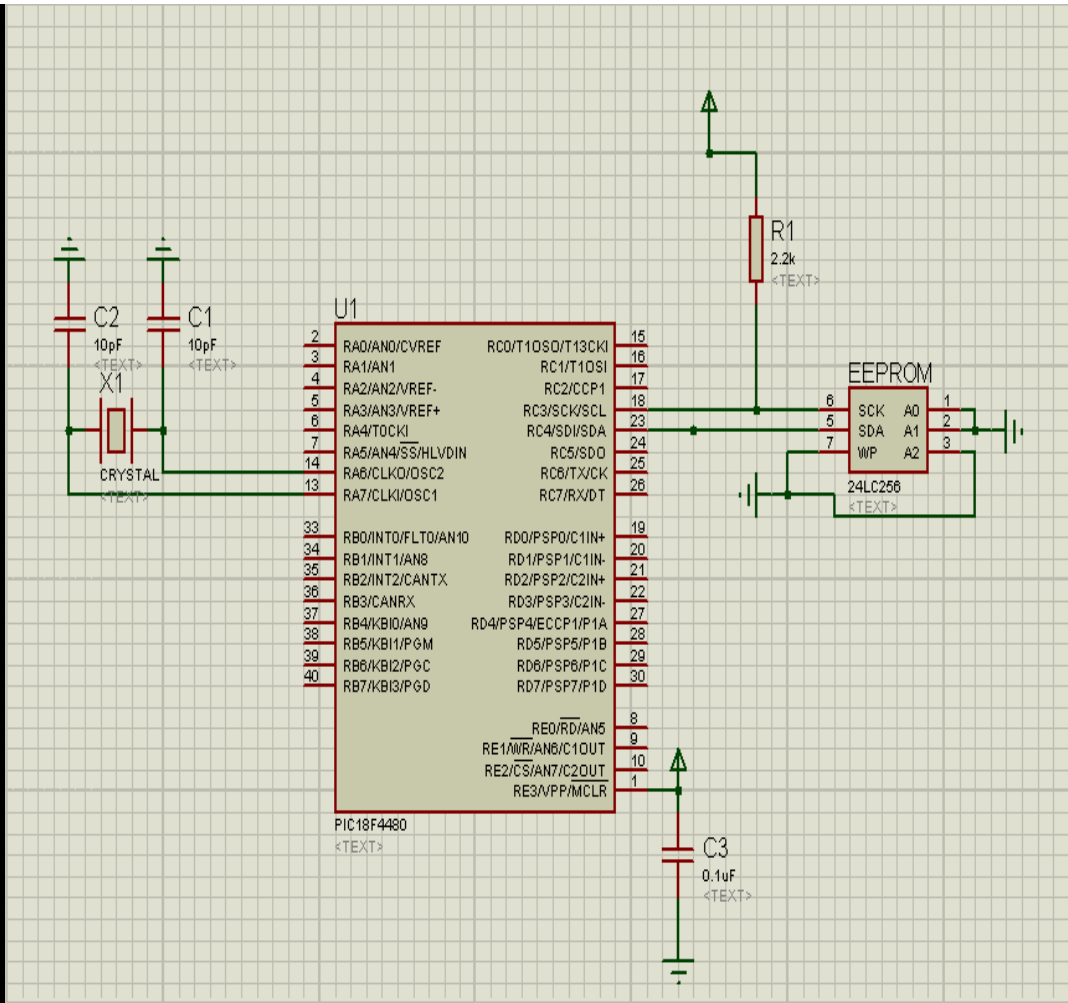
Manufacturer	Address
Acura	0Xxxxx
Audi	0Xxxxx
BMW	0Xxxxx
Chevrolet	0Xxxxx
Dodge / Chrysler / Jeep	0Xxxxx
Ford	0Xxxxx
etc	

- Segments Table:

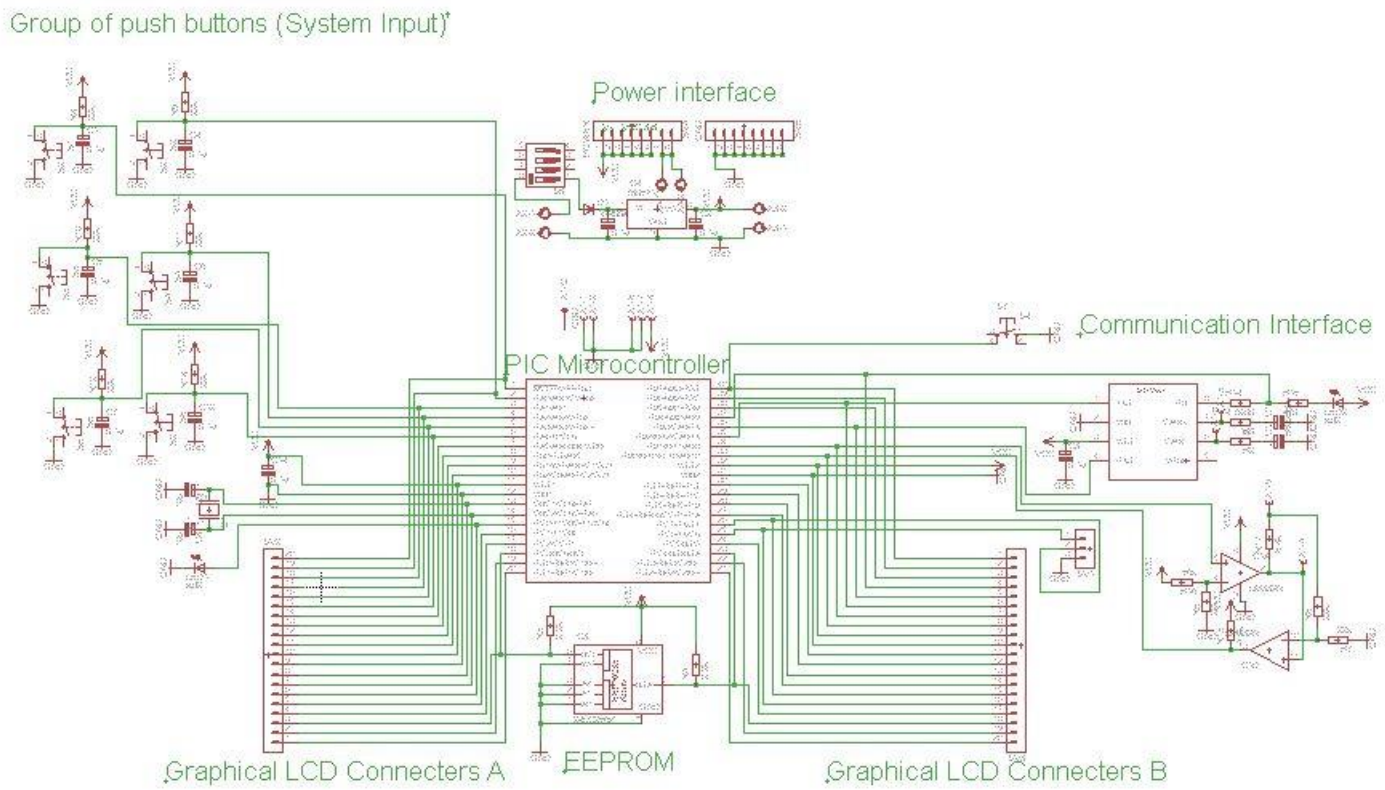
Segment	Type	Size
Error Code	Static, because all codes has the	5 bytes
Description Address	Static	2 bytes
Description Length	Static	3 bytes
Description – One Arabic char.	Static	2 byte
Description	Variable, assigned with its code	-

Storage unit EEPROM – Proteus & PCB Design

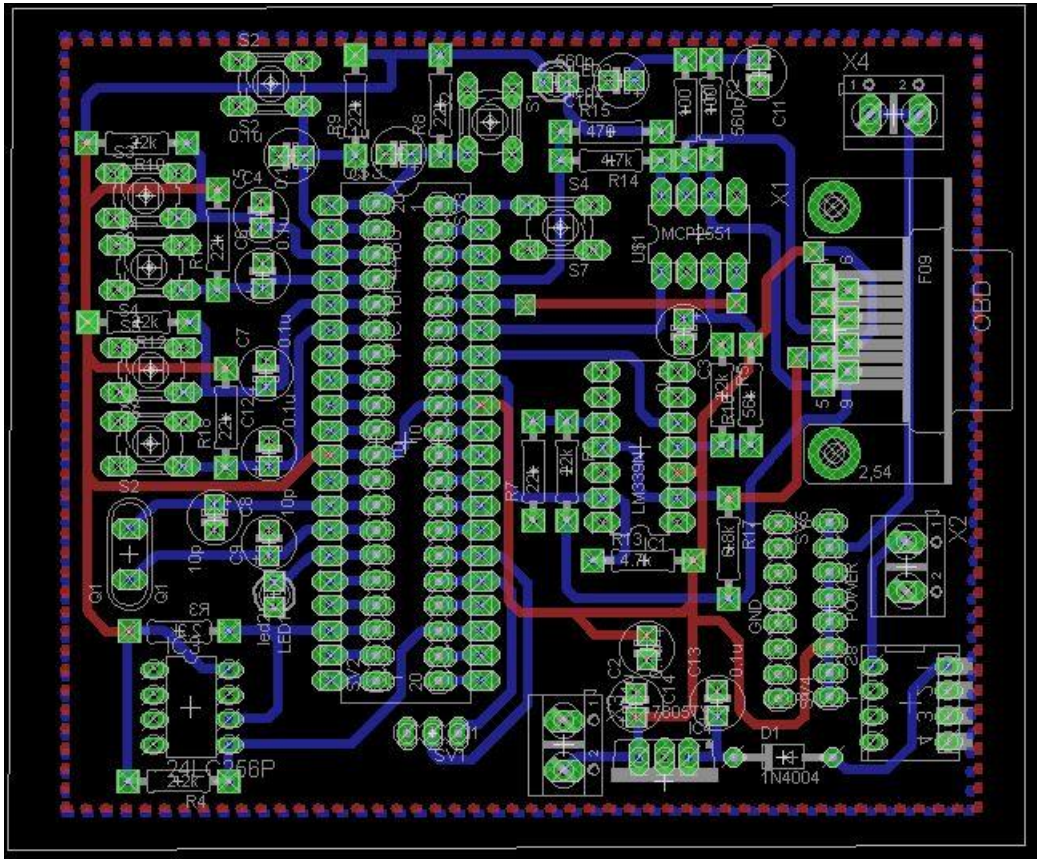




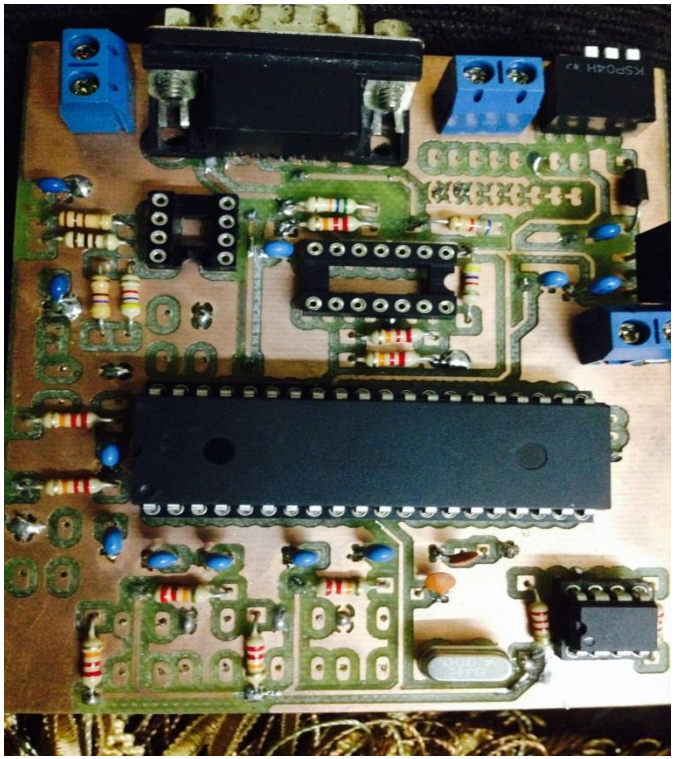
Hardware implementation – Eagle schematic

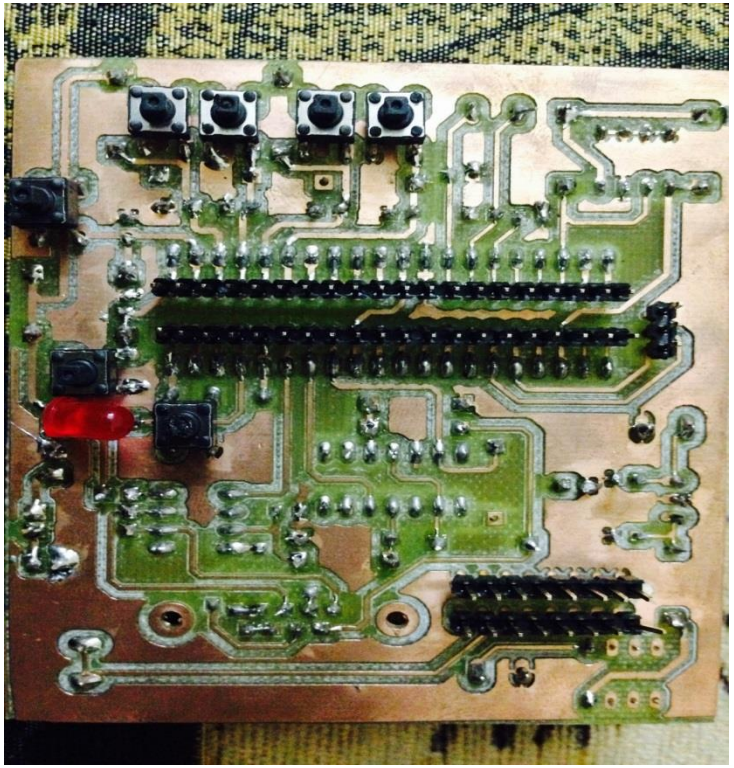


Hardware implementation – Eagle Printed circuit design



Hardware implementation – Screenshoots





Similar systems

D900

Standalone OBD scanner is connected to the OBD interface on car directly and supports the five main communication protocols.

Limitations

- Standards availability
  - On-board diagnostics system has five common communication protocols, this protocols specifics the communication process between the OBD scanner and the car ECU. These protocols are ISO standards, not available for free and the average price for each protocol 100-200$. This problem has been solved after a long search for free documents that describe the required details about each protocol.

- Testing environment
  - Our testing environment is cars, and for each protocol we need different type of cars, we could not find a car ECU emulator which will ease testing process, also we faced problems in finding people who will let us make our testing on their cars.
- Time
  - The available period for graduation project implementation which is four months is very adequate, but unfortunately we spent two and a half months on another project (implementing home automation system through home electrical network) but we have faced many problems that forced us to change the idea so we worked on OBD Scanner project for the half of the period.

Standards/Codes

- ISO 15765-4:2011
  - Road vehicles -- Diagnostic communication over Controller Area Network (DoCAN) -- Part 4: Requirements for emissions-related systems
- ISO 9141:1989
  - Road vehicles -- Diagnostic systems -- Requirements for interchange of digital information
- ISO 14230-4:2000
  - Road vehicles -- Diagnostic systems -- Keyword Protocol 2000 -- Part 4: Requirements for emission-related systems
- ISO 15031-5:2011
  - Road vehicles -- Communication between vehicle and external equipment for emissions-related diagnostics -- Part 5: Emissions-related diagnostic services

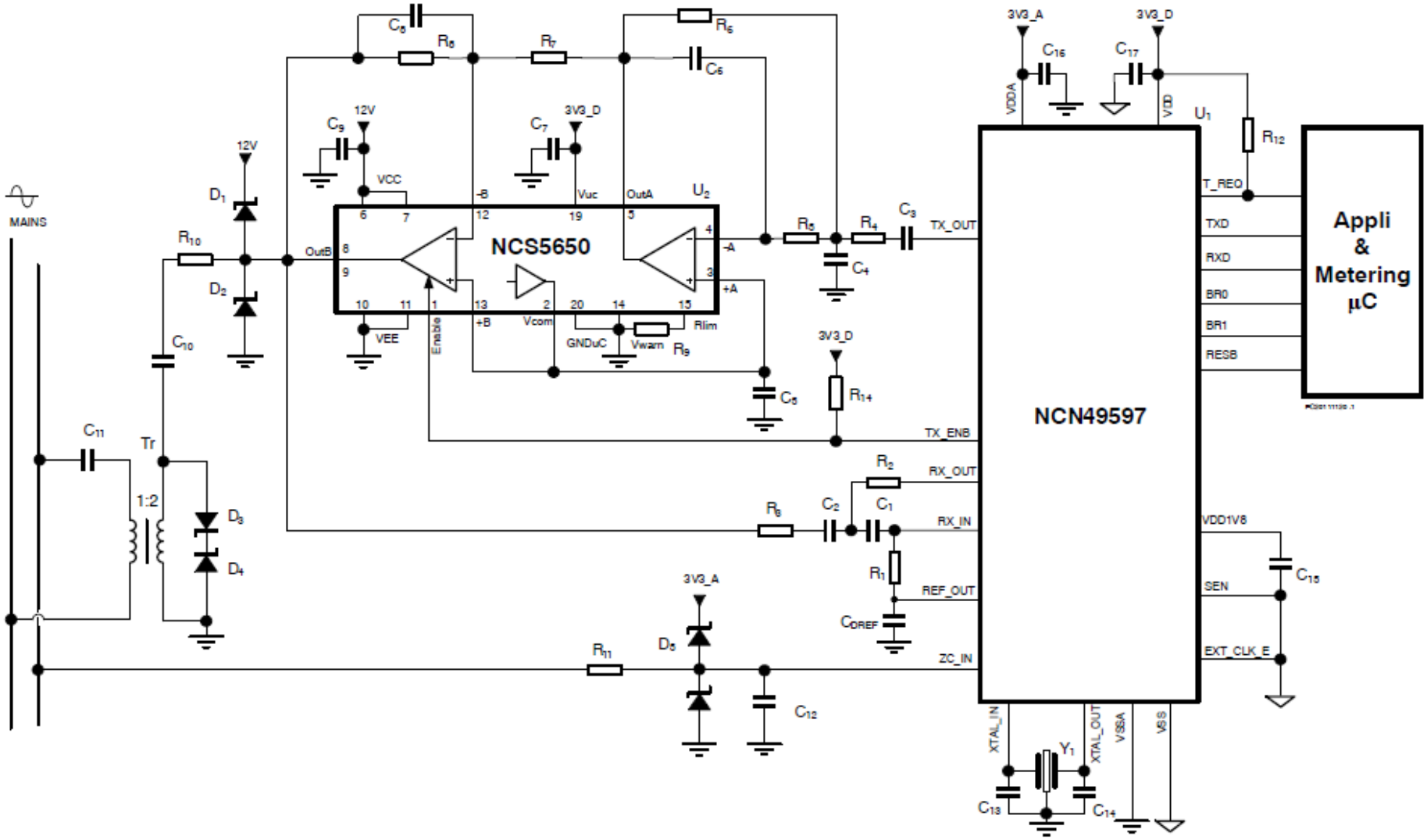
OBD simulator

PLC Project

- About
  - A remotely controlled electrical plug system which applies functionalities like (on & off) operations, timed operations and power readings. The brilliant idea here is the communication medium, its neither WIFI nor Bluetooth which always raise the costs; the electrical signals of electricity home network are used to communicate with plugs.

- Technology
  - Power line carrier communication

PLC Project-power line carrier modem



PLC Project - Problems

- Printed circuit design of NCN 49597
- Special circuit components availability
- High speed printing requirements

PLC Project-PLC evaluation board

- C2000 Power Line Modem Developer's Kit ($599.00)

Thanks