Aayush Gupta

309601

Vehicle onboard computer:

Introduction:

The vehicle on-board computer is designed to provide essential information about the vehicle's internal temperature, outside temperature, current speed, and current and averaged fuel consumption. The device utilizes the EFM8BB10F2A microcontroller as the main control unit, along with other components such as the LM317 variable power supply, LM35 temperature sensors, 1602A LCD display, and the STN1110 OBD to UART interpreter.

Thus, the aim of this project is to read the date via OBD2 interface and display the data on LCD display.

Hardware specifications and their working:

1. **EFM8BB10F2A Microcontroller:** The microcontroller serves as the brain of the system, controlling and coordinating the various components. It reads sensor data, performs calculations, and manages the display and communication interfaces.

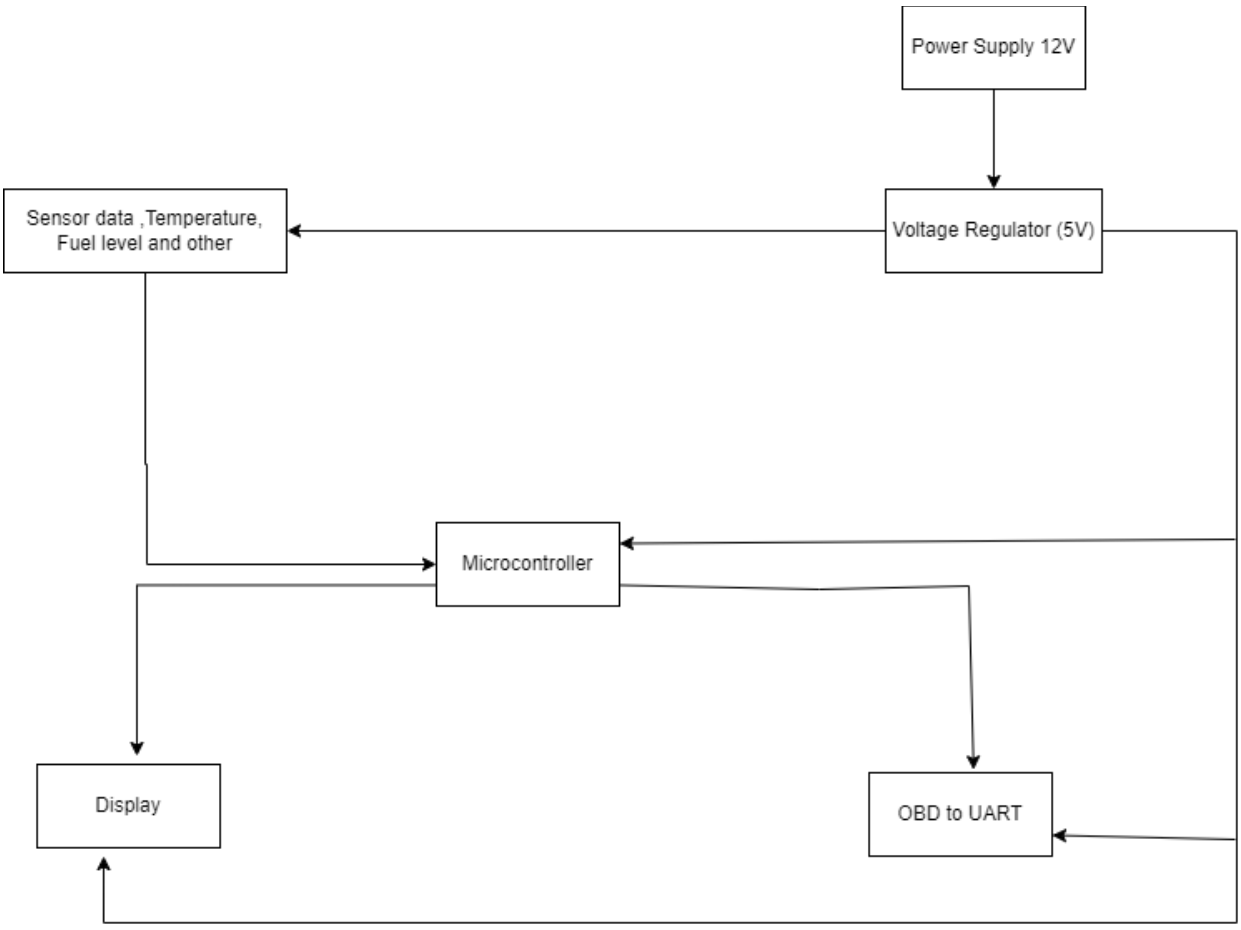
2. LM317 Variable Power Supply: The LM317 is a voltage regulator that provides variable output voltage (1.25V to 37V) to the components in the system. It ensures that the microcontroller and other components receive the appropriate voltage for their operation.

3. LM35 Temperature Sensors: The LM35 sensors are used to measure the internal and outside temperatures of the vehicle. They convert temperature into analog voltage readings, which are then read by the microcontroller to calculate and display the temperature values. These sensors output voltage is linearly proportional to the Celsius temperature.

4. 1602A LCD Display: The LCD 1602A display is a 16x2 character LCD module that can display alphanumeric characters and symbols. IT is powered by built in backlight with low power consumption. As for compatibility with the EFM8BB10F2A, the microcontroller can be interfaced with the LCD 1602A display using a simple 8-bit parallel interface.

5. STN1110 OBD to UART Interpreter: The STN1110 module serves as an interface between the microcontroller and the vehicle's OBD-II system. It is designed to provide bidirectional half-duplex communication with the vehicle’s On-Board Diagnostic System (OBD-II). And Our microcontroller that has a built-in UART interface, which makes it compatible with the STN1110. The UART interface.

6. Block diagram:



In summary, the microcontroller controls and coordinates the system, the LM317 provides stable power, the LM35 sensors measure temperature, the 1602A LCD display provides visual feedback, and the STN1110 module enables communication with the vehicle's OBD-II system. Together, these components work together to measure and display internal temperature, outside temperature, current speed, and fuel consumption in vehicle on-board computer.

Algorithm for the project –

1. Initialize the microcontroller and set up the necessary GPIO pins for interfacing with the components, including the LCD display.
2. Initialize the UART communication module of the microcontroller to communicate with the STN1110 OBD to UART interpreter. Set the baud rate and other necessary parameters.
3. Initialize the LCD display module, configure the necessary control and data pins, and setting up the display parameters such as cursor position, backlight, and other settings.
4. Continuously loop the following steps:

a. Send an appropriate OBD-II command to the STN1110 module using the UART interface to request specific vehicle data. This could include commands to retrieve vehicle speed, fuel consumption, or other desired parameters.

b. Receive the response from the STN1110 module through the UART interface. Parse and extract the relevant data from the received response.

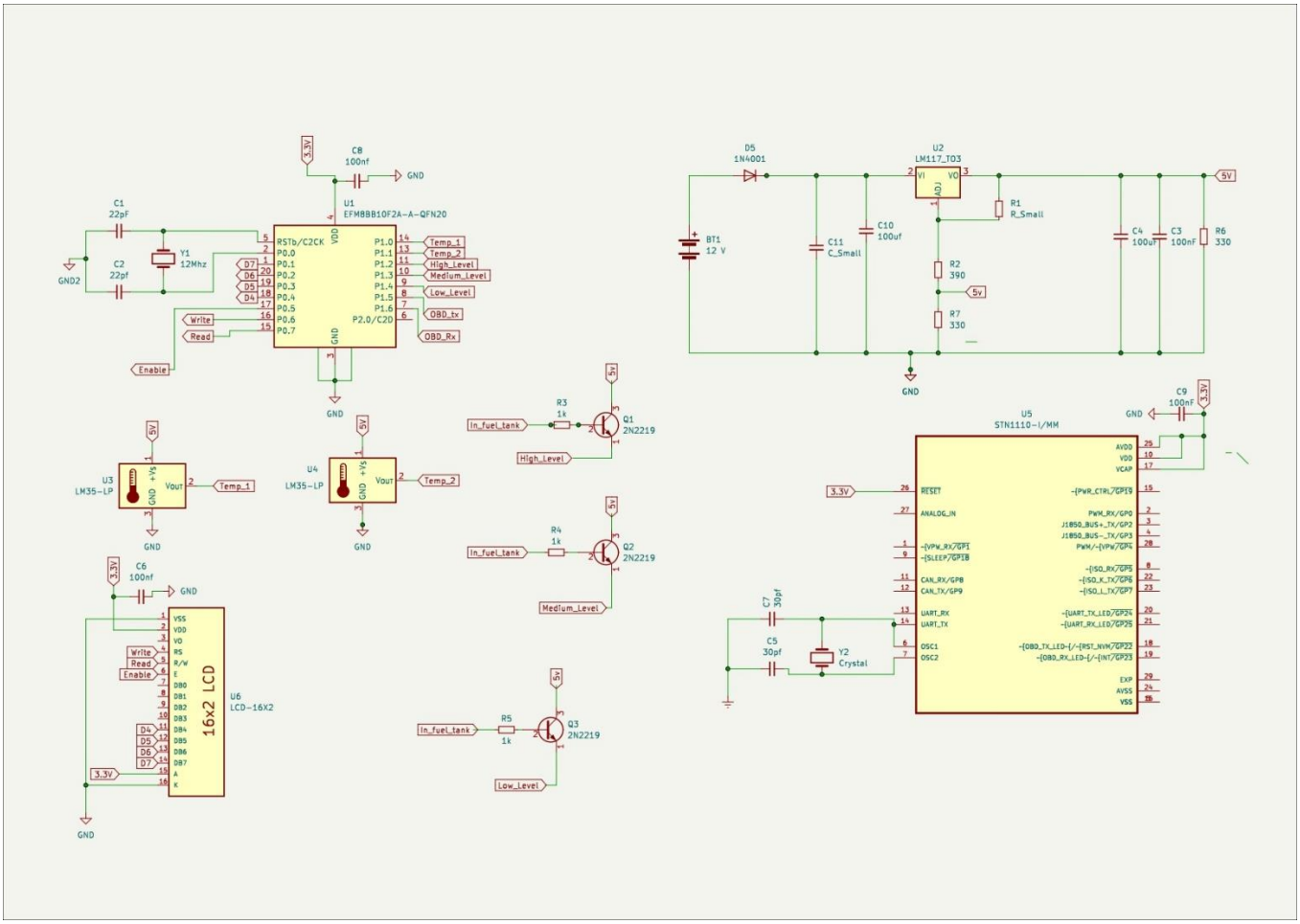
c. Perform any necessary calculations or conversions on the data obtained to obtain the desired units or formats. For example, converting fuel consumption values from liters per hour to miles per gallon.

d. Update the content of the LCD display with the retrieved and processed data. This involves writing the appropriate characters or symbols to the LCD module at the desired positions.

e. Delay for a certain period to control the refresh rate of the display and the rate of data retrieval. This delay allows for smoother operation and prevents overwhelming the microcontroller or the display.

5. End the loop and the program execution when desired or when a specific condition is met.

Schematic:


C8
100nf
GND
C1
22pF
C2
22pF
GND2
Y1
12Mhz
U1
EFM8BB10F2A-A-QFN20
RSTb/C2CK
P0.0
P0.1
P0.2
P0.3
P0.4
P0.5
P0.6
P0.7
P1.0
P1.1
P1.2
P1.3
P1.4
P1.5
P1.6
P2.0/C2D
VDD
GND
Temp_1
Temp_2
High_Level
Medium_Level
Low_Level
OBD_tx
OBD_Rx
Write
Read
Enable
D7
D6
D5
D4
D5
1N4001
U2
LM117_TO3
VI
VO
ADJ
BT1
12 V
C11
C_Small
C10
100uf
R1
R_Small
R2
390
R7
330
C4
100uF
C3
100nF
R6
330
5V
GND
U3
LM35-LP
U4
LM35-LP
+Vs
VOUT
GND
Temp_1
Temp_2
R3
1k
Q1
2N2219
In_fuel_tank
High_Level
R4
1k
Q2
2N2219
Medium_Level
R5
1k
Q3
2N2219
Low_Level
C6
100nf
U6
LCD-16X2
16x2 LCD
VSS
VDD
VO
RS
R/W
E
DB0
DB1
DB2
DB3
DB4
DB5
DB6
DB7
A
K
3.3V
C9
100nF
U5
STN1110-I/MM
AVDD
VDD
VCAP
RESET
ANALOG_IN
-{VPW_RX/GPI
-{SLEEP/GPIB
CAN_RX/GPB
CAN_TX/GP9
UART_RX
UART_TX
OSC1
OSC2
-{PWR_CTRL/GPI9
PWM_RX/GP0
J1850_BUS+_TX/GP2
J1850_BUS-_TX/GP3
PWM/-{VPW/GP4
-{ISO_RX/GP5
-{ISO_K_TX/GP6
-{ISO_L_TX/GP7
-{UART_TX_LED/GP24
-{UART_RX_LED/GP25
-{OBD_TX_LED-{/-{RST_NVM/GP22
-{OBD_RX_LED-{/-{INT/GP23
EXP
AVSS
VSS
C7
30pf
C5
30pf
Y2
Crystal