**📌 Car Black Box System - Project Notes**

**🛠️ Project Overview**

**Car Black Box System** is a **PIC16F877A-based system** that **logs events**, **displays real-time data**, and **allows log retrieval** using **UART communication**.

**📌 Features:**

✅ **LCD Display** - Shows time, speed, and gear position. ✅ **Keypad Input** - Used for password entry and menu navigation. ✅ **EEPROM Storage** - Stores logs (event, speed, time) permanently. ✅ **RTC (DS1307)** - Maintains real-time clock even after power-off. ✅ **UART Communication** - Allows log retrieval via serial interface. ✅ **Interrupt-Based Timer** - Used for countdowns and password blocking.

**📌 System Flow (प्रणालीचा प्रवाह)**

1️⃣ **System Initialization** (main.c)

* LCD, Keypad, ADC, Timer1, EEPROM, UART, I2C, RTC चालू करतो.

2️⃣ **Dashboard Display** (dashboard.c, clcd.c)

* LCD वर **वेळ, गियर स्थिती आणि वेग** दर्शवतो.

3️⃣ **Password Authentication** (pass\_menu.c, matrix\_keypad.c)

* **MK\_SW11 दाबल्यास**, पासवर्ड एंटर करण्यासाठी स्विच होईल.
* EEPROM मधून स्टोर्ड पासवर्ड तपासतो.
* **3 चुकीच्या एंट्री नंतर**, युजर **120 सेकंद ब्लॉक** होतो.

4️⃣ **Menu Navigation** (menu.c)

* युजर **मेणू पर्याय निवडतो:**
  1. **View Logs** - EEPROM मधून डेटा पाहतो.
  2. **Download Logs** - UART द्वारे डेटा ट्रान्सफर करतो.
  3. **Clear Logs** - EEPROM मधील डेटा डिलीट करतो.
  4. **Set Time** - RTC ची वेळ अपडेट करतो.
  5. **Change Password** - नवीन पासवर्ड सेट करतो.

5️⃣ **Event Logging** (ext\_eep.c, ds1307.c)

* **गियर बदलेल किंवा विशेष इव्हेंट होईल तेव्हा डेटा स्टोअर होतो** (वेळ, गती, घटना).

6️⃣ **Interrupt Handling** (isr.c, timer.c)

* **Timer1 ISR** ब्लॉकिंग टाइम आणि इव्हेंट टाइमिंग नियंत्रित करतो.

7️⃣ **Data Transfer (UART)** (uart.c, download\_log.c)

* UART द्वारे पीसी वर लॉग डाउनलोड करतो.

**📌 Step-wise Code Explanation**

**🔹 Step 1: Main Program (main.c)**

#include <xc.h>

#include "main.h"

void main(void) {

init\_config(); // Initialize all peripherals

while (1) {

get\_time(); // Fetch RTC Time

key = read\_switches(STATE\_CHANGE);

if (main\_f == DASHBOARD) {

display\_dashboard();

} else if (main\_f == PASSWORD) {

password();

} else if (main\_f == MENU) {

menu(key);

}

}

}

**🔹 Step 2: LCD Display (clcd.c)**

void clcd\_print(const unsigned char \*data, unsigned char addr) {

clcd\_write(addr, INSTRUCTION\_COMMAND);

while (\*data != '\0') {

clcd\_write(\*data++, DATA\_COMMAND);

}

}

**🔹 Step 3: Keypad Handling (matrix\_keypad.c)**

unsigned char scan\_key(void) {

unsigned char key = 0xFF;

for (unsigned char row = 0; row < 4; row++) {

PORTB = ~(1 << row);

for (unsigned char col = 0; col < 3; col++) {

if (!(PORTB & (1 << (col + 4)))) {

key = (row \* 3) + col + 1;

return key;

}

}

}

return key;

}

**🔹 Step 4: ADC for Speed Sensor (adc.c)**

unsigned short read\_adc(unsigned char channel) {

ADCON0 = (ADCON0 & 0xC3) | (channel << 2);

GO = 1;

while (GO);

return (ADRESH << 8) | ADRESL;

}

**🔹 Step 5: RTC Time Handling (ds1307.c)**

unsigned char read\_ds1307(unsigned char address) {

i2c\_start();

i2c\_write(SLAVE\_WRITE);

i2c\_write(address);

i2c\_rep\_start();

i2c\_write(SLAVE\_READ);

unsigned char data = i2c\_read();

i2c\_stop();

return data;

}

**🔹 Step 6: EEPROM Storage (ext\_eep.c)**

void write\_ext\_eep(unsigned char address, unsigned char data) {

i2c\_start();

i2c\_write(EEP\_WRITE);

i2c\_write(address);

i2c\_write(data);

i2c\_stop();

}

**🔹 Step 7: Interrupt Service Routine (isr.c)**

void \_\_interrupt() isr(void) {

if (TMR1IF) {

TMR1 = TMR1 + 3038;

if (++count == 80) {

count = 0;

tm--;

}

TMR1IF = 0;

}

}

**🔹 Step 8: UART Communication (uart.c)**

void putch(unsigned char byte) {

while (!TXIF);

TXREG = byte;

}

int puts(const char \*s) {

while (\*s) {

putch(\*s++);

}

return 0;

}

**📌 Conclusion**

✅ **Car Black Box System** वाहनाच्या लॉगिंग आणि डेटा स्टोरेज साठी एक विश्वासार्ह प्रणाली आहे. ✅ **ही प्रणाली वाहतुकीतील घटनांची माहिती स्टोअर करते आणि ती UART द्वारे ट्रान्सफर करण्यास सक्षम आहे.** ✅ **हे Embedded C, I2C, UART, LCD Interface आणि RTC Handling शिकण्यासाठी उत्तम प्रोजेक्ट आहे.** 🚀