**main.cpp**

// Подключение библиотек

#include <WiFi.h>

#include <esp\_now.h>

#include <Wire.h>

#include <Adafruit\_MAX6675.h>

#include <PN532\_I2C.h>

#include <PN532.h>

// Ввод MAC-адреса мастер-устройства

#define CHANNEL 1

uint8\_t masterAddress[] = {0x0, 0x0, 0x0, 0x0, 0x0, 0x0}; // Заменить MAC-адресом мастер-устройства

// Настройка подключения датчиков

const int vibrationSensorPin = 33; // Подключение датчика вибрации (акселлерометра)

const int wearSensorPin = 34; // Подключение датичка износа

const int oilPressureSensorPin = 35; // Подключение датчика давления масла

Adafruit\_MAX6675 thermocouple(13, 12, 14); // Подключение датчика MAX6675 с термопарой: SCK - пин 13, CS - пин 12, SO - пин 14

// Настройка NFC

PN532\_I2C pn532\_i2c(Wire);

PN532 nfc(pn532\_i2c);

// Обратная функция ESP-NOW

void OnDataRecv(const uint8\_t\* macAddr, const uint8\_t\* data, int dataLen) {

// Получение данных с мастер-устройства при настройке взаимодействия

}

// Основной цикл

void setup() {

Serial.begin(115200);

// Настройка режимов пинов для датчиков

pinMode(vibrationSensorPin, INPUT);

pinMode(wearSensorPin, INPUT);

pinMode(oilPressureSensorPin, INPUT);

// Настройка ESP-NOW

if (esp\_now\_init() != ESP\_OK) {

Serial.println("ESP-NOW initialization failed");

return;

}

esp\_now\_register\_recv\_cb(OnDataRecv);

esp\_now\_add\_peer(masterAddress, ESP\_NOW\_ROLE\_CONTROLLER, CHANNEL, NULL, 0);

}

void loop() {

// Чтение данных датчиков

float temperature = thermocouple.readCelsius();

int vibrationData = digitalRead(vibrationSensorPin);

String wearData = readWearSensor();

int oilPressureData = analogRead(oilPressureSensorPin);

String readNFCModule() {

String result = "Некорректная установка"; // Значение при ошибке

nfc.begin();

uint8\_t success;

uint8\_t uid[] = { 0, 0, 0, 0, 0, 0, 0 }; // Буфер для хранения идентификационного номера NFC-метки

uint8\_t uidLength; // длина идентификационного номера по стандарту ISO14443A

success = nfc.readPassiveTargetID(PN532\_MIFARE\_ISO14443A, uid, &uidLength);

if (success) {

result = "Установлен корректно"; // Сообщение при корректной установке

}

return result;

}

String nfcData = readNFCModule();

// Создание сообщения

String payload = "Температура: " + String(temperature) + " °C\n";

payload += "Вибрация: " + String(vibrationData) + "\n";

payload += "Износ: " + wearData + "\n";

payload += "Давление масла: " + String(oilPressureData) + "\n";

payload += "Датчик положения: " + nfcData;

// Отправка данных на мастер-устройство посредством ESP-NOW

esp\_now\_send(masterAddress, (uint8\_t\*)payload.c\_str(), payload.length());

delay(5000); // Настройка ожидания

}

String readWearSensor() {

int sensor