https://download.geofabrik.de/asia/kazakhstan.html.

Отлично — давай отправлять данные с ESP32 прямо в **Google Sheets** через Google Apps Script и Webhook. Это бесплатный способ логгировать любые данные в Google Таблицу **без серверов и баз данных**.

## ✅ Что ты получишь

ESP32 будет:

1. Каждые 5 минут забирать уровень пробок из JSON.
2. Отправлять этот уровень и метку времени в Google Таблицу.
3. Всё хранится в Google Sheets, можно строить графики и анализ.

## 📁 Шаг 1: Подготовь Google Таблицу

1. Создай новый Google Sheet: [https://sheets.new](https://sheets.new/)
2. Назови, например: Almaty Traffic Log
3. В **первую строку** добавь заголовки:

Time, Level

## 🧩 Шаг 2: Напиши Google Apps Script

1. В таблице нажми Extensions → Apps Script
2. Удали всё и вставь:

function doPost(e) {

var sheet = SpreadsheetApp.getActiveSpreadsheet().getSheetByName("Лист1");

var data = JSON.parse(e.postData.contents);

var time = new Date();

var level = data.level;

sheet.appendRow([time, level]);

return ContentService.createTextOutput("OK");

}

1. Нажми **Deploy → New deployment**
2. Нажми **Select type → Web app**
3. Установи:
   * Execute as: **Me**
   * Who has access: **Anyone**
4. Нажми **Deploy**, разреши доступ → скопируй URL вида:

https://script.google.com/macros/s/AKf.../exec

## 🔐 Шаг 3: Код ESP32 для отправки в Google Sheets

Вот доработанный код с отправкой в Google:

#include <WiFi.h>

#include <HTTPClient.h>

const char\* ssid = "YOUR\_WIFI\_SSID";

const char\* password = "YOUR\_WIFI\_PASSWORD";

const char\* traffic\_url = "https://raw.githubusercontent.com/yourusername/traffic-data/main/almaty-traffic.json";

const char\* google\_webhook\_url = "https://script.google.com/macros/s/AKf.../exec"; // вставь свой URL

unsigned long interval = 5 \* 60 \* 1000;

unsigned long lastCheck = 0;

void setup() {

Serial.begin(115200);

WiFi.begin(ssid, password);

while (WiFi.status() != WL\_CONNECTED) delay(500);

Serial.println("WiFi connected.");

}

void loop() {

unsigned long now = millis();

if (now - lastCheck >= interval || lastCheck == 0) {

lastCheck = now;

if (WiFi.status() == WL\_CONNECTED) {

HTTPClient http;

http.begin(traffic\_url);

int httpCode = http.GET();

if (httpCode == 200) {

String payload = http.getString();

Serial.println("Traffic JSON: " + payload);

int idx = payload.indexOf("level");

int colon = payload.indexOf(":", idx);

int end = payload.indexOf("}", colon);

String levelStr = payload.substring(colon + 1, end);

levelStr.trim();

int trafficLevel = levelStr.toInt();

Serial.print("Traffic level: ");

Serial.println(trafficLevel);

// Отправляем в Google Sheets

HTTPClient httpPost;

httpPost.begin(google\_webhook\_url);

httpPost.addHeader("Content-Type", "application/json");

String jsonPayload = "{\"level\": " + String(trafficLevel) + "}";

int postCode = httpPost.POST(jsonPayload);

Serial.print("POST response: ");

Serial.println(postCode);

httpPost.end();

}

http.end();

}

}

delay(1000);

}

## ✅ Что можно улучшить

* Добавить **обработку ошибок / offline-режим**.
* Поддержка **SD-карты для бэкапа**.
* График на Google Sheets: просто вставь Line Chart по столбцу Level.

Хочешь — могу ещё автоматизировать создание графика, Telegram-бота с уведомлениями, или систему на ESP32 с дисплеем.

ID

AKfycbwJ3I56O3ciOp6uDYiHC1E70mrcyABTBYZbk1ArKzafGYB2FzGMfXGMn4C9Pwk4XkeO  
  
  
Deployment  
https://script.google.com/macros/s/AKfycbwJ3I56O3ciOp6uDYiHC1E70mrcyABTBYZbk1ArKzafGYB2FzGMfXGMn4C9Pwk4XkeO/exec