



**БОЛЬШИЕ ВЫЗОВЫ**

ВСЕРОССИЙСКИЙ КОНКУРС  
НАУЧНО-ТЕХНОЛОГИЧЕСКИХ ПРОЕКТОВ



**Региональный конкурс научно-технологических проектов**

**«Большие вызовы»**

**Свердловская область 2022–2023 учебный год**

# Устройство слежения за состоянием водителя «NSControl»

**Направление: Умный город и безопасность**

**Тип: Прикладной**

**Автор: Андреев Алексей Сергеевич**

**Город, школа, класс: г.Каменск-Уральский, Лицей №9, 10 класс**

**Научный руководитель проекта: Киселева Ирина Анелидовна, педагог дополнительного образования МБУ ДО «Центр дополнительного образования»**



**ПОБЕДИТЕЛЯМ – «СИРИУС»!**

**#большиевызовы**

# Актуальность

- **Транспортный поток:** увеличение численности и разновидностей.
- **Водители:** возраст, стаж, опыт.
- **Ежегодно** попадают в ДТП в мире 350 тысяч людей (24%), в РФ около 45 тысяч людей (27%).
- **Устройство для отслеживания:** патентов немного, функционал не оптимален.

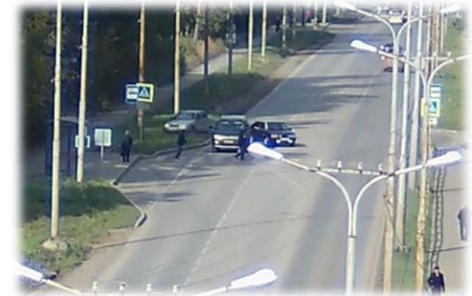


# Цель

Создание прототипа устройства слежения за состоянием водителя, который можно использовать в любом транспортном средстве

# Задачи

1. Изучение литературы и интернет-источников.
2. Выбор аппаратного обеспечения.
3. Разработка и конструирование устройства.
4. Выбор программного обеспечения.
5. Создание программного кода.
6. Тестирование устройства.
7. Анализ результатов и корректировка.





# Предмет, объект исследования

Объект: устройство обнаружения засыпания человека за рулем и оповещение.

Предмет: аппаратное и программное обеспечение (микрокомпьютер Nvidia Jetson Nano, Visual Studio Code, библиотеки распознавания лиц и объектов.)

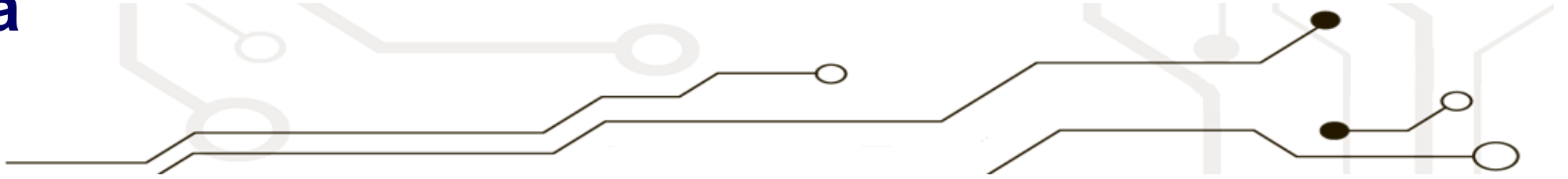
## Методы исследования

Наблюдение, сравнение, анализ, моделирование, прототипирование, конструирование, программирование, тестирование.

## Гипотеза

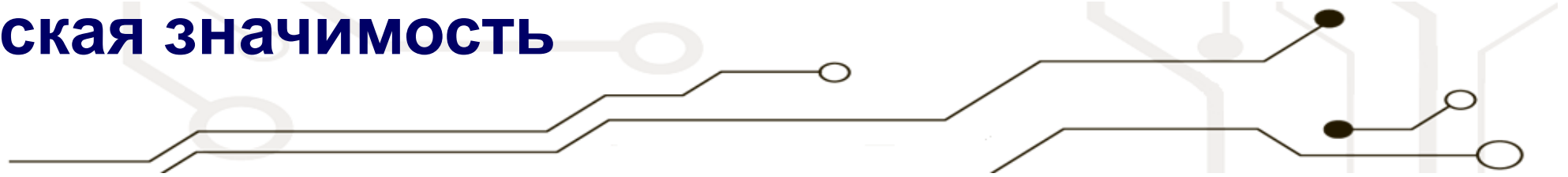
Созданное устройство не позволит водителю уснуть или отвлечься за рулём, что должно уменьшить количество ДТП.

# Новизна



- **Возможность** установки в любое транспортное средство.
- **Логирование** скорости движения и местоположения с помощью GPS модуля.
- Организация **обратной** связи на базе GSM модуля
- **Возможность** выноса камеры в зону, определяемую по желанию водителя.
- Использование **сенсорных** кнопок в качестве элемента управления.
- Угол обзора **220°**.

# Практическая значимость



- Способность **отслеживания** момента засыпания или несосредоточенности водителя за рулём и **оповещения** звуковым сигналом.
- Возможность применения в **большинстве** транспортных средств: железнодорожном, водном, автомобильном, воздушном.
- Способность **отслеживания** местоположения и скорости транспортного средства.
- Возможность **автоматически** оповещать экстренную службу, если водитель не реагирует на звуковой сигнал и продолжает движение.

# Аналоги устройств

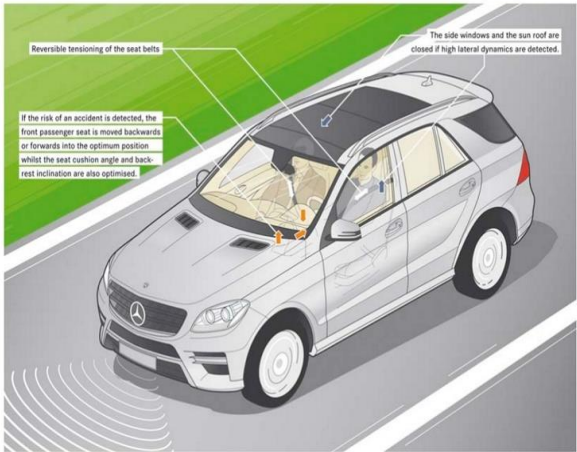
Dunobil Insomnia



Антисон



Attention Assist





# Аналоги устройств

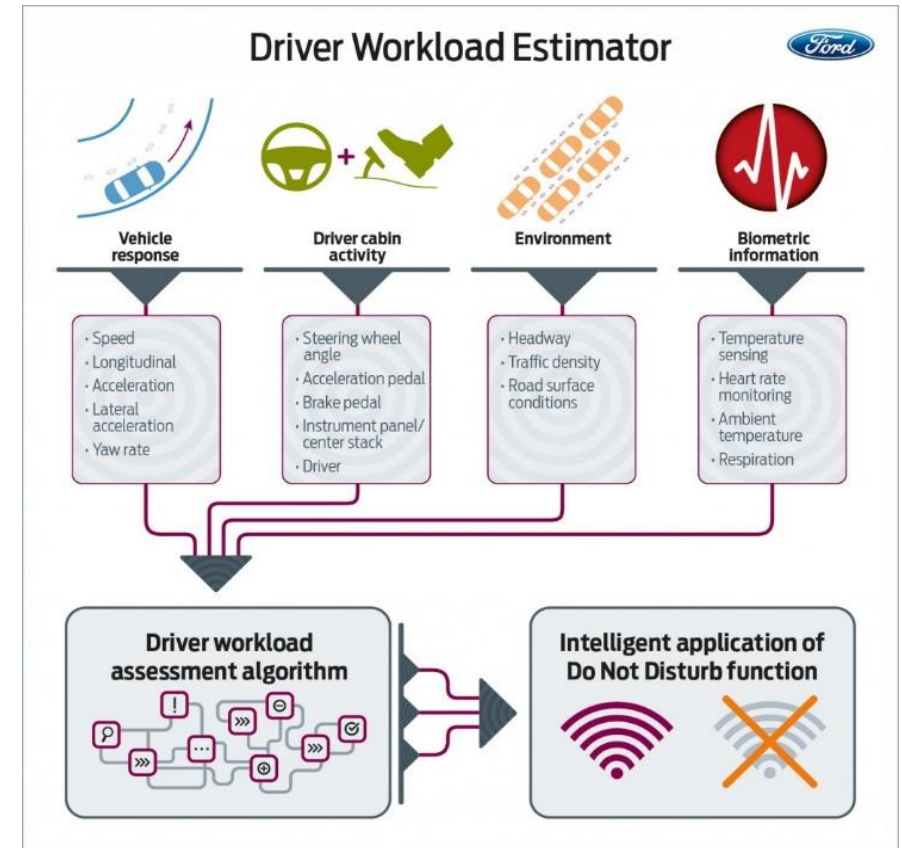
## STEER



## Sleep Alert



## Driver Workload Estimator





# Подведение итогов



**Сравнительный анализ аналогов** (критерии: точность определения засыпания, возможность работы от аккумулятора, логирование данных GPS, оповещение, работа в ночное время, цена и т.д.)

**Выводы:** устройства имеют возможность определять момент засыпания и оповещения об этом водителя, работают в достаточно большом диапазоне температур и освещённости, имеют удобный способ установки и эксплуатации.

**Личный опыт:** работа с Lego Mindstorms EV3, Arduino; изучение схемотехники, программирования на языках Python и C++, моделирования и прототипирования.



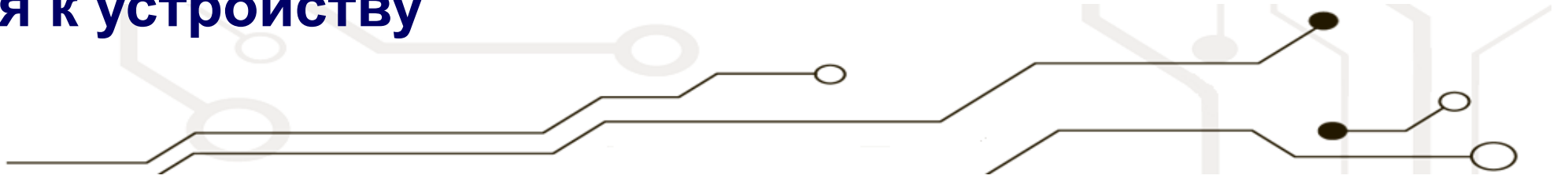
**NSControl» No Sleep Control**

# Этапы работы

1. Изучение интернет источников по проблеме.
2. Выбор оборудования для реализации устройства.
3. Создание набросков и чертежей.
4. Установка OS и драйверов.
5. Загрузка и настройка зависимостей и библиотек.
6. Создание программного кода для инициализации камеры, обработки изображения, обнаружения и распознавания лиц, работы GPS модуля, динамиков, кнопок.
7. Моделирование корпуса и частей крепления устройства.
8. Печать готовых моделей на 3Д принтере.
9. Тестирование устройства. Анализ и корректировка.



# Требования к устройству

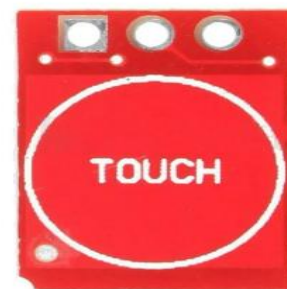
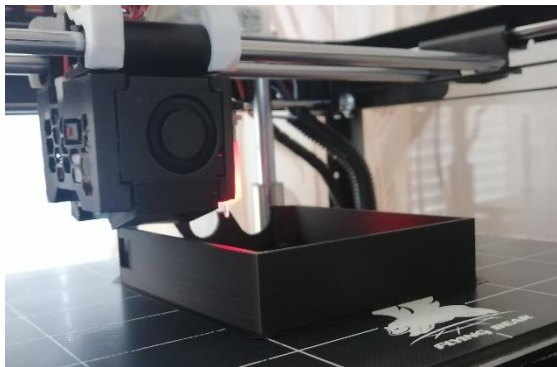


- Не наносить вред человеку.
- Точно определять засыпание.
- Должно иметь возможность установки в любом транспортном средстве.
- Работать в ночное время.
- Должно быть удобным в эксплуатации.



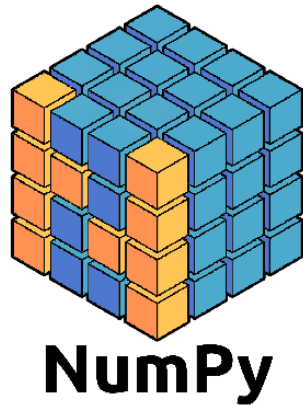
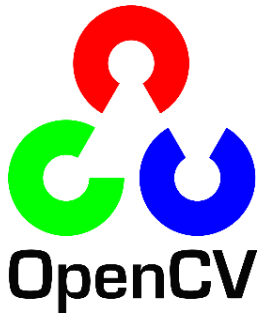


# Аппаратное обеспечение



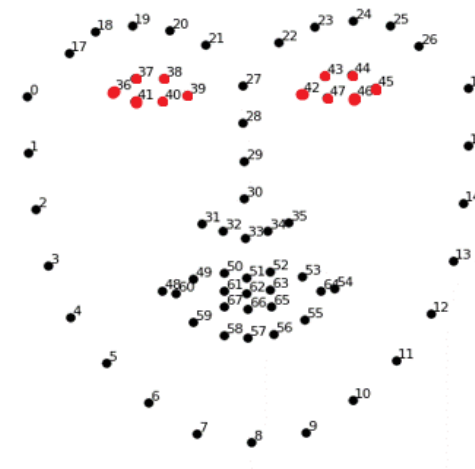
# Программное обеспечение

- JetPack SDK (пакет драйверов L4T, OS Linux, библиотеки CUDA-X и API) для общей работы устройства.
- Библиотеки OpenCV, Numpy, Matplotlib служат для обработки исходного изображения с камеры.
- Библиотека Dlib для отслеживания точек лица, направления взгляда.



# Программное обеспечение

- Visual Studio Code для разработки программного кода на языке Python.

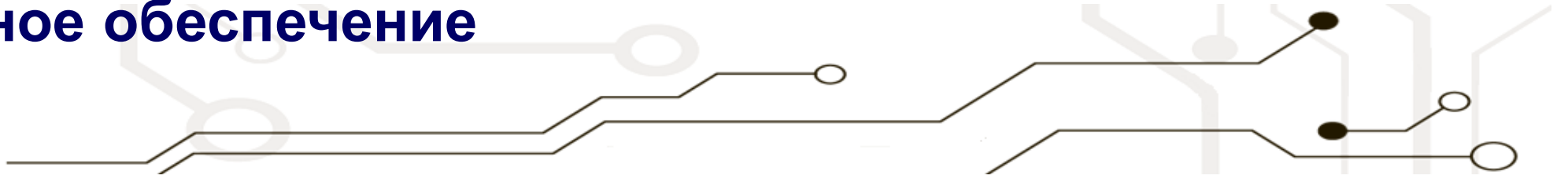


## Пример программного кода

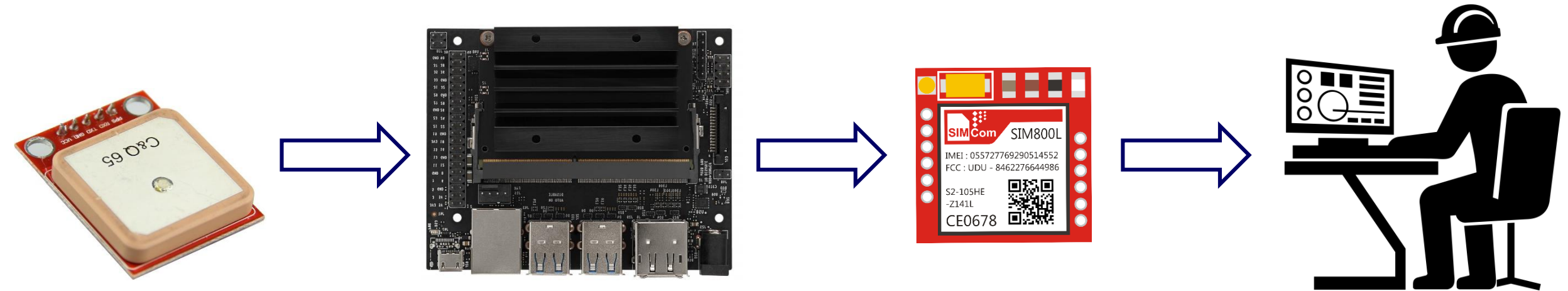
```
38 def get_blinking_ratio(eye_points, facial_landmarks):
39     left_point = (facial_landmarks.part(eye_points[0]).x, facial_landmarks.part(eye_points[0]).y)
40     right_point = (facial_landmarks.part(eye_points[3]).x, facial_landmarks.part(eye_points[3]).y)
41
42     center_point = midpoint(facial_landmarks.part(eye_points[1]), facial_landmarks.part(eye_points[2]))
43     center_bottom = midpoint(facial_landmarks.part(eye_points[5]), facial_landmarks.part(eye_points[4]))
44
45     #eyes_line = cv2.line(frame, left_point, right_point, (255, 0, 0), 2)
46     #vert_line = cv2.line(frame, center_point, center_bottom, (255, 0, 0), 2)
47
48     eyes_line_lenght = hypot((left_point[0] - right_point[0]), (left_point[1] - right_point[1]))
49     vert_line_lenght = hypot((center_point[0] - center_bottom[0]), (center_point[1] - center_bottom[1]))
50
51     ratio = eyes_line_lenght / vert_line_lenght
52     return ratio
```



# Программное обеспечение



- Библиотека GPSTools для функционирования GPS модуля.
- Библиотека SIM800L для работы GSM модуля.

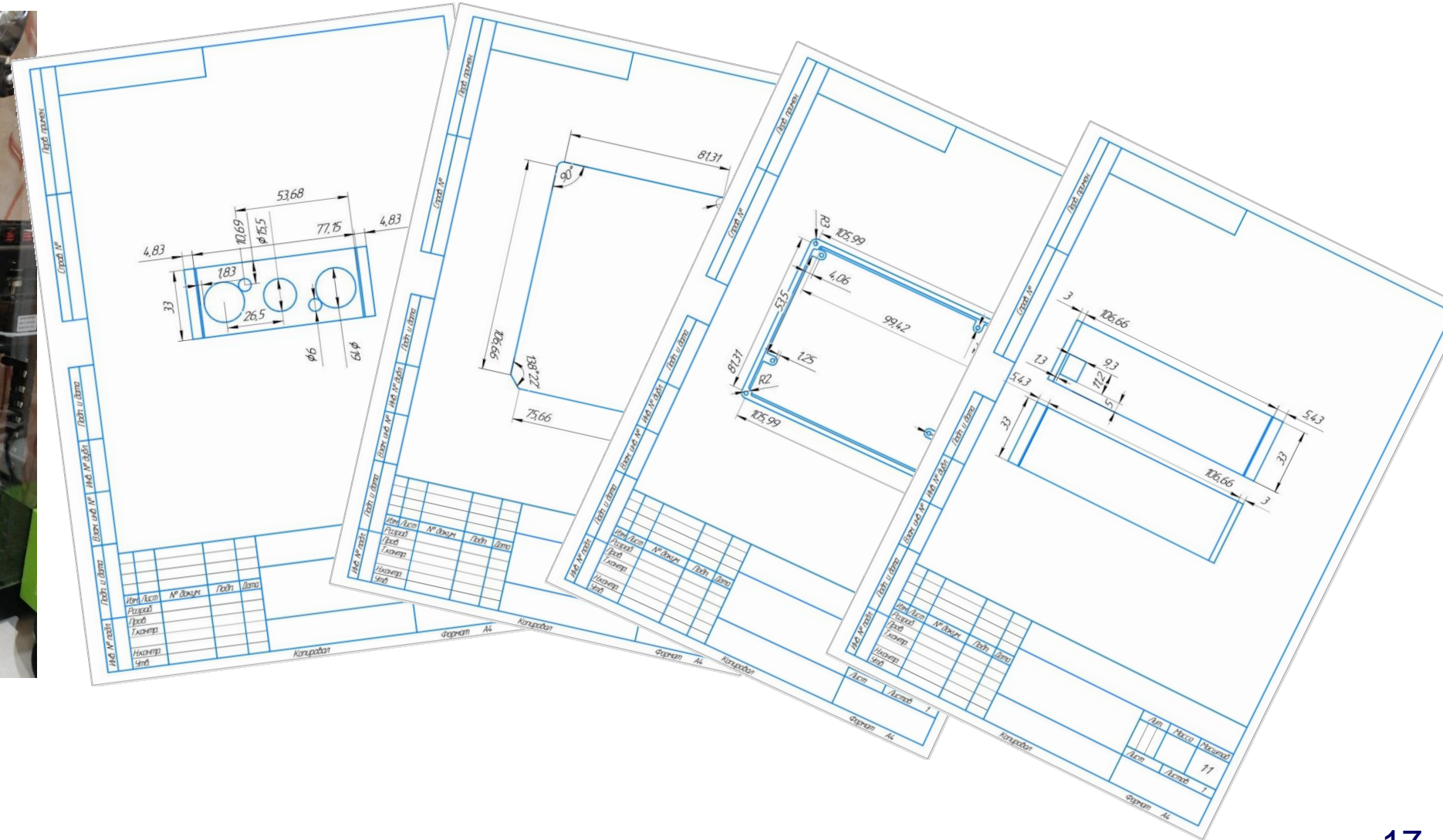


# Программное обеспечение

- Программы КОМПАС-3D и Autodesk Inventor для создания чертежей и моделей.
- Программа EasyEDA для составления схем.

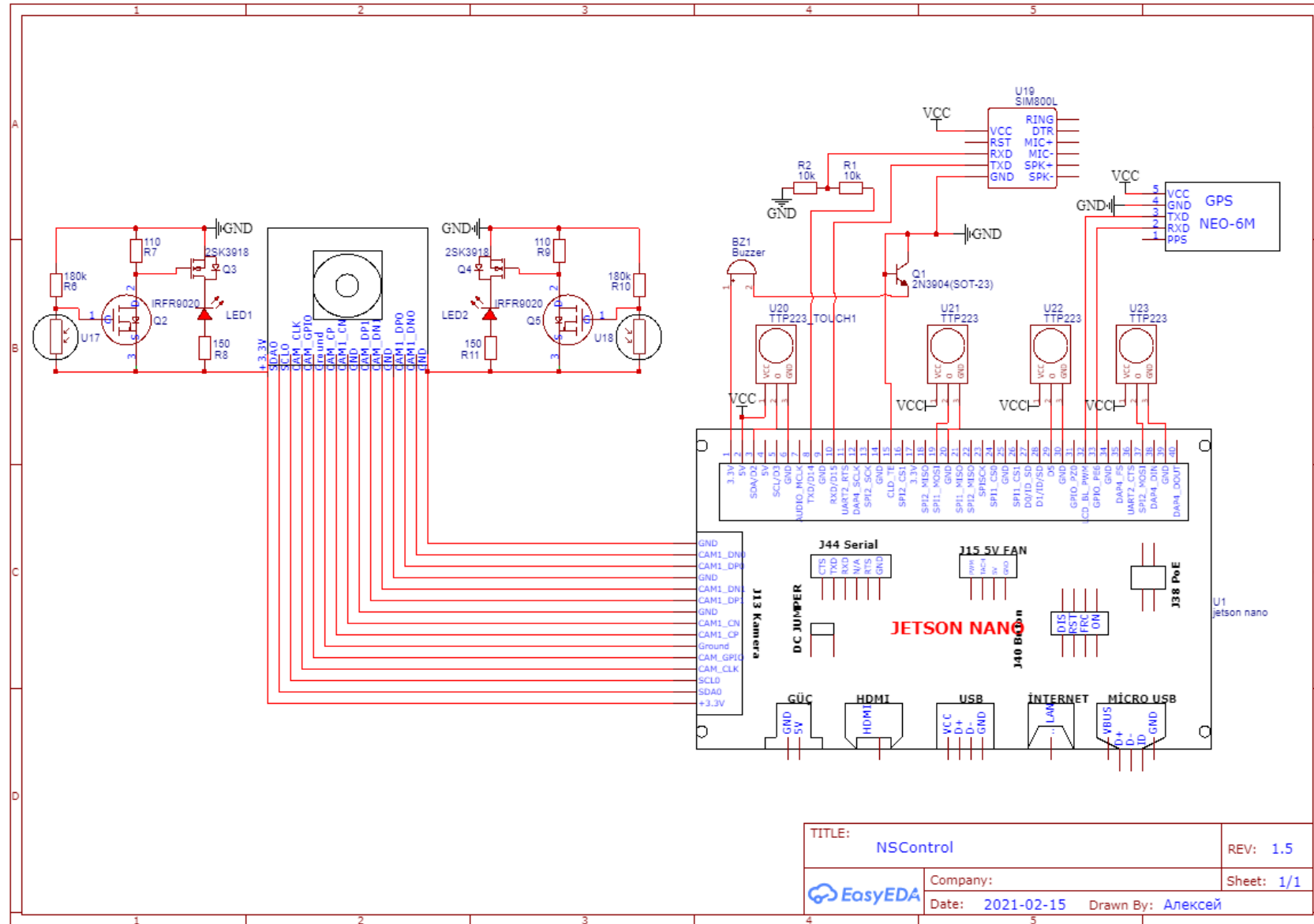


# Чертежи в 3Д Компасе

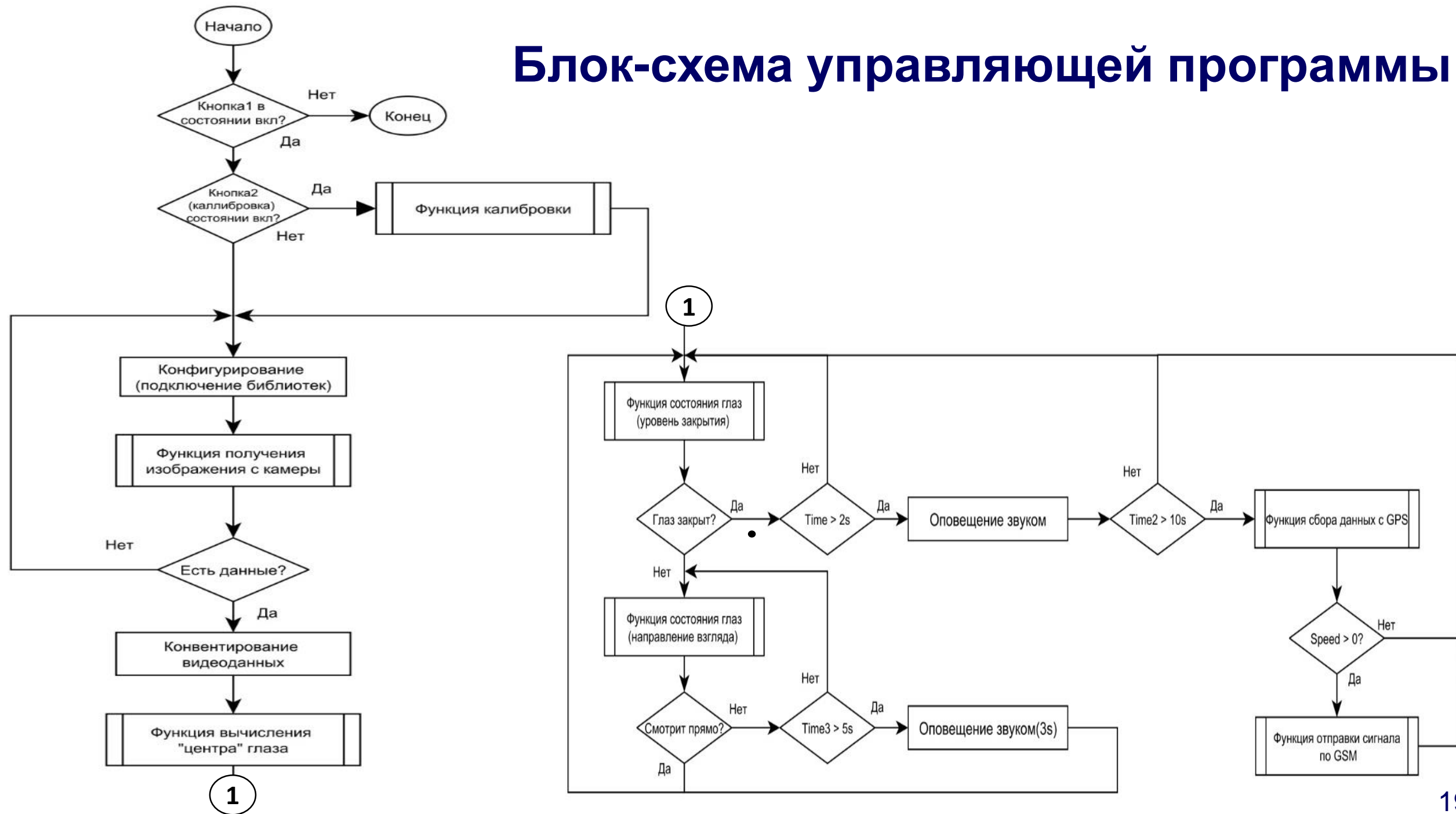




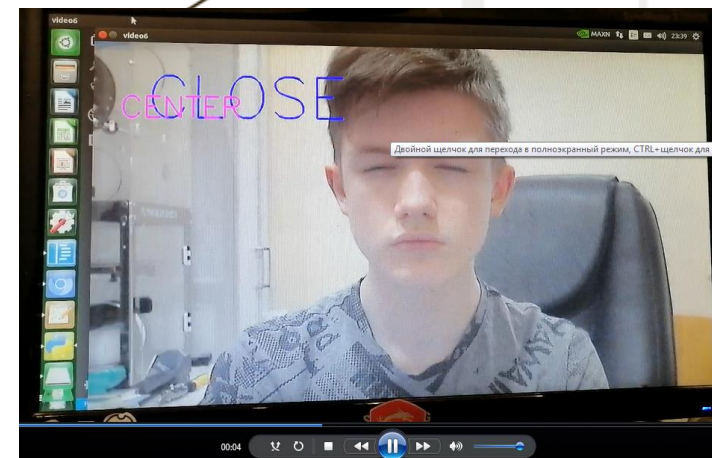
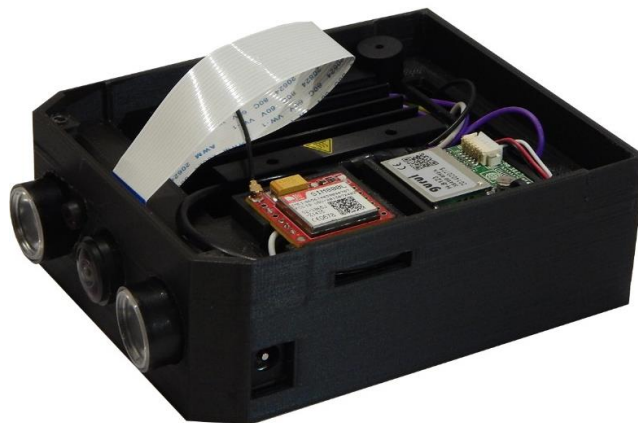
# Принцип работы



# Блок-схема управляющей программы



# Тестирование



← NSControl

19:44

Внимание, транспортное средство A000AA 196, водитель не реагирует на предупреждения. (Скорость 75 км/ч, координаты GPS 56.41814187892487, 61.930322530619826)

Внимание, транспортное средство A000AA 196, водитель не реагирует на предупреждения. (Скорость 65 км/ч, координаты GPS 56.414590949350114, 61.930296797783406)

Сейчас



# Заключение

## Сравнительный анализ

Таблица 2. Сравнительный анализ Dunobil Insomnia, Антисон, STEER, Sleep Alert, NSControl

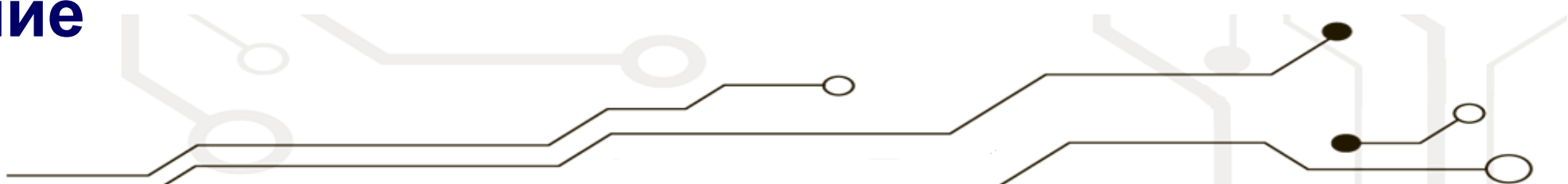
Критерий/Название аналога	<u>Dunobil Insomnia</u>	<u>“Антисон”</u>	STEER	Sleep Alert	<u>NSControl</u>
Точность определения засыпания.	Точно	Неточно	Неточно	Точно	<b>Точно</b>
Работа от аккумулятора	Нет	Да	Да	Да	<b>В перспективе</b>
Время работы от аккумулятора	-	3 – 4 недели.	1 – 2 недели	6 – 7 часов	<b>Не проверено</b>
Возможность использования в любом ТС	Да	Да	Да	Да	<b>Да</b>
<u>Логирование</u> данных GPS, отправка диспетчеру	Нет	Нет	Нет	Нет	<b>В перспективе</b>
Оповещение	Звуковой сигнал	Звуковой сигнал	Электрический разряд	Вибросигнал	<b>Звуковой сигнал</b>
Работа в ночное время	Да	Да	Да	Да	<b>Да</b>

# Заключение



**Перспектива** – уменьшение размеров устройства, реализация процесса автоматического обучения на базе сверточной нейронной сети, оптимизация программного кода.

# Заключение



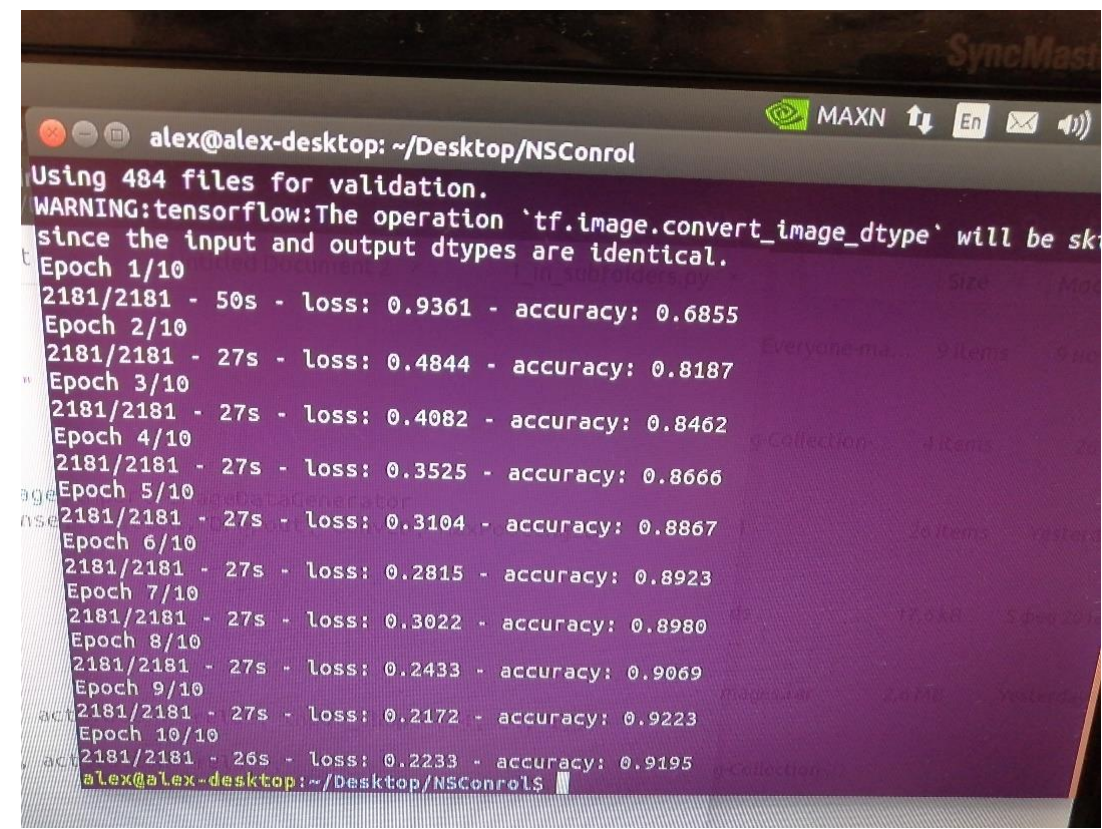
```
import os

os.environ["TF_CPP_MIN_LOG_LEVEL"] = "2"

import tensorflow as tf
import tensorflow.keras
import tensorflow.keras.layers
import tensorflow.keras.preprocessing.image
import tensorflow.keras.preprocessing.image.ImageDataGenerator
import tensorflow.keras.layers.Dense, tensorflow.keras.layers.Flatten, tensorflow.keras.layers.Dropout, tensorflow.keras.layers.Conv2D, tensorflow.keras.layers.MaxPooling2D

height = 24
width = 24
channel_size = 2

model = keras.Sequential([
    Conv2D(18, (3,3), padding='same', activation='relu', input_shape=(24, 24, 1)),
    MaxPooling2D((2, 2), strides=2),
    Conv2D(36, (3,3), padding='same', activation='relu'),
    MaxPooling2D((2, 2), strides=2),
    Flatten(),
    Dense(128, activation='relu'),
    Dense(10, activation='softmax')
])
```



***Спасибо за внимание!***

