

# **Разработка интерфейса управления компьютером через нейроинтерфейс**

Евсеев Олег Александрович

Научный руководитель – ст. пр. Ханов А.Р. (СПбГУ)

Рецензент – к.т.н., доц. Воробьева А.А. (Университет ИТМО)

# Введение

- Понимание принципов работы мозга – актуальная тема
- Наибольшие успехи – электроэнцефалография
- Существуют устройства, снимающие ЭЭГ в домашних условиях – нейроинтерфейсы
  - 5–15 электродов, закрепляемых на поверхности головы
  - Регистрируют изменения электрических потенциалов
- Существуют решения для управления компьютером, но они недружелюбны к пользователям, не знакомым с предметной областью

# Цель работы

Целью данной работы является реализация прикладного решения, позволяющего осуществлять управление компьютером с использованием нейроинтерфейса Emotiv EPOC+.



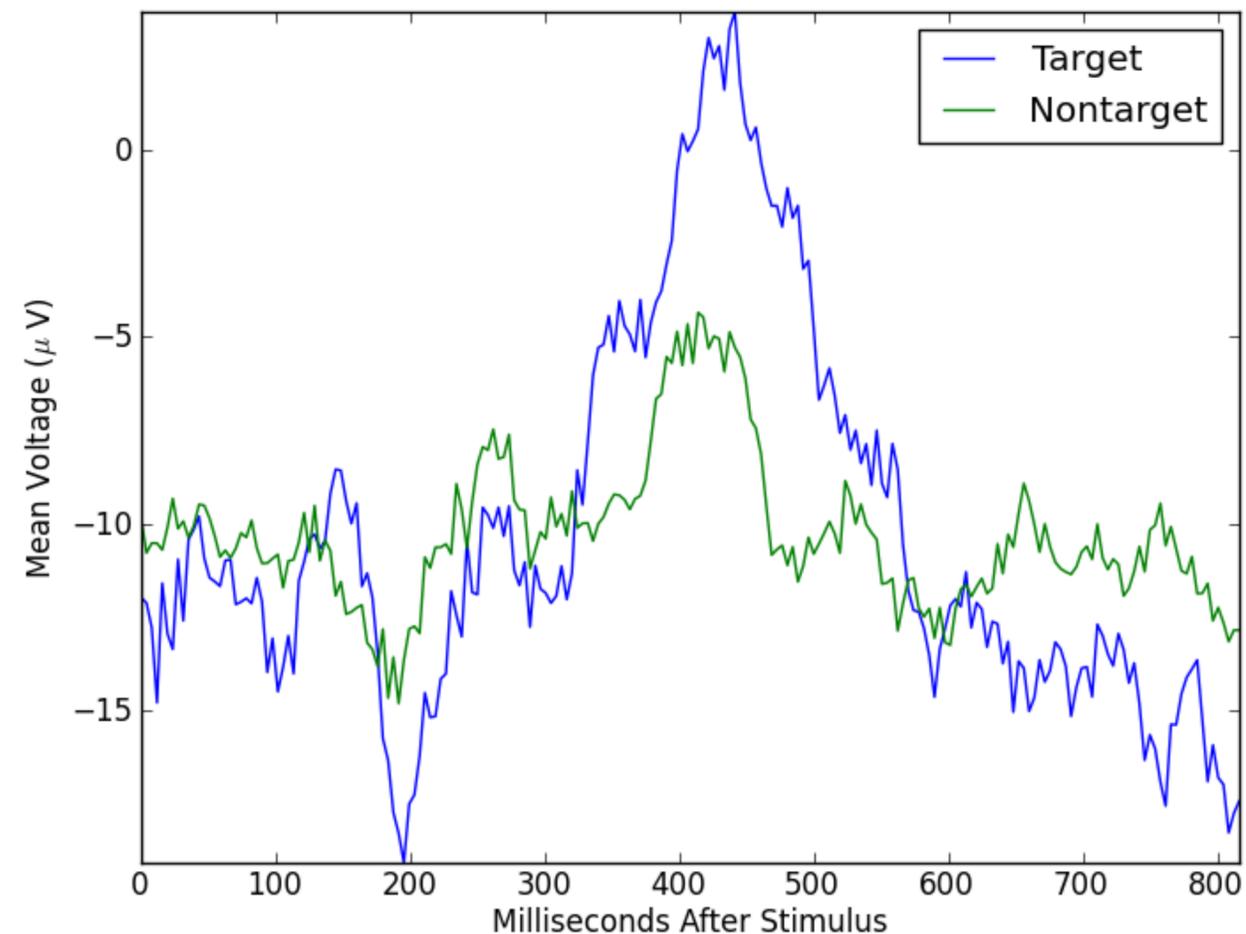
<https://www.emotiv.com/>

# Задачи

- Выработка требований
- Разработка алгоритма, способного извлекать из ЭЭГ нужную информацию
- Проектирование и реализация прототипа, позволяющего осуществлять взаимодействие с компьютером
- Апробация решения

# Что можно обнаружить в ЭЭГ?

- Ритмы головного мозга
- Потенциалы, связанные с событиями
  - P300 – реакция на редкий и релевантный стимул
  - Легко стимулировать («oddball-парадигма»)



Anderson, C. «P300 Waves for Single Subjects»

# Существующие решения

- P300 – хорошо исследованная тема
  - Есть ряд простых подходов, дающих хорошее качество обнаружения
- Варианты конструирования пользовательского интерфейса предлагаются в работе Ali Haider, Reza Fazel-Rezai. *Application of P300 Event-Related Potential in Brain-Computer Interface*
- Конкретные реализации обычно реализуют конкретные сценарии (чаще всего – ввод текста) и не взаимодействуют с ОС

# Решения для получения данных

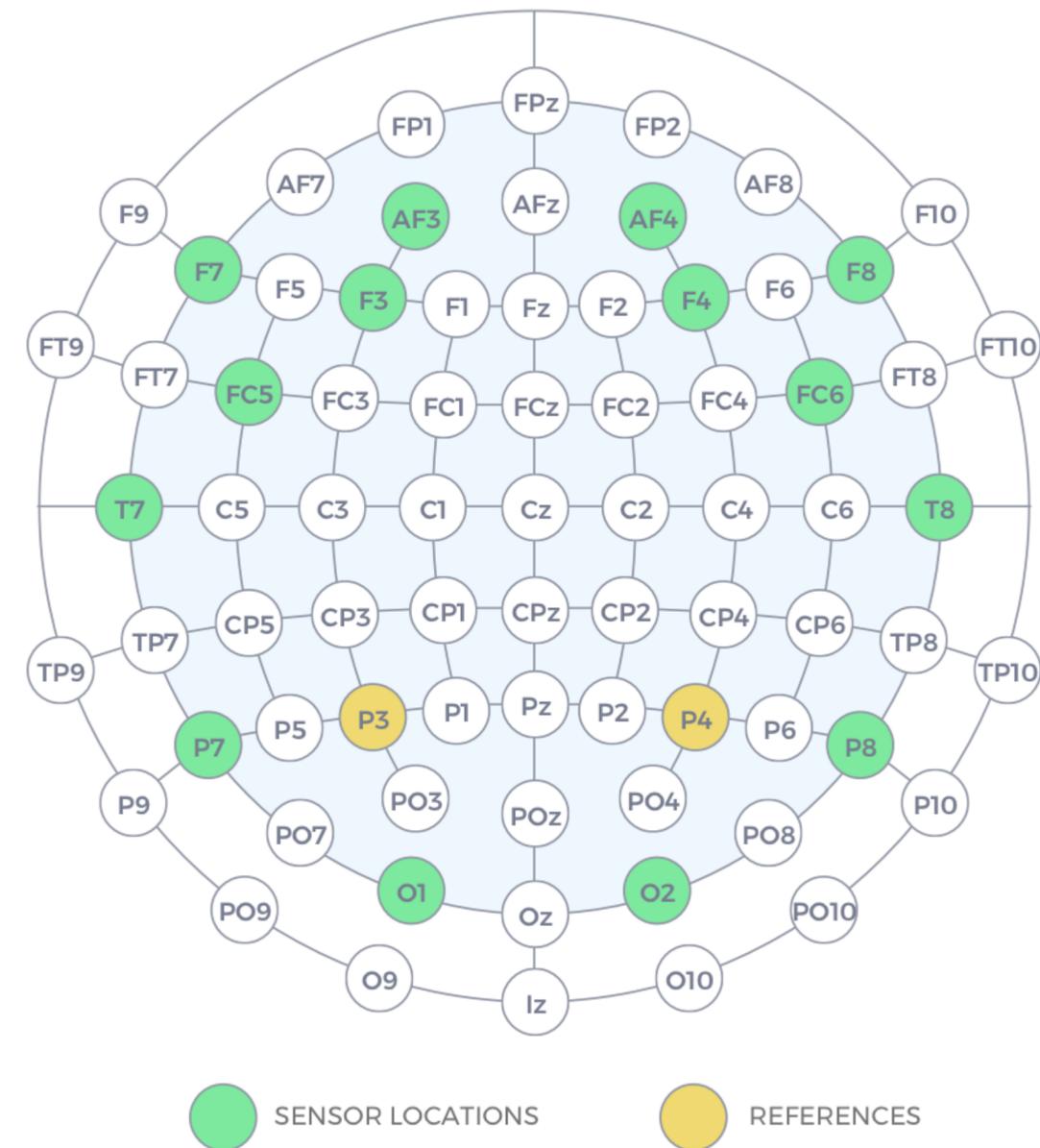
- Официальные (EmotivPRO, Emotiv Community SDK) – проприетарные
- EmoKit
  - Не поддерживается с 2017 года
  - Не умеет расшифровывать данные EPOC+ 2016 года выпуска и позже
- CyKit – актуальное решение
  - Только для Windows
  - Есть возможность работы в режиме TCP-сервера

# Требования

- Функциональные
  - Взаимодействие с ОС
  - Обработка сигнала в реальном времени
- Нефункциональные
  - Сигнализация об ошибках
  - Приемлемая точность распознавания
  - Простота использования
  - Целевая платформа – ОС Windows
  - Использование потенциала P300
  - Использование CyKit

# Описание данных

- 14 датчиков (2 референсных для аппаратной коррекции сигнала)
  - Разрешение – 16 бит (наименьший бит = 0.13 мкВ)
- Частоты 50 Гц и 60 Гц (помехи от сетей переменного тока) удаляются аппаратно
- Внутренняя частота дискретизации – 2048 Гц, понижается до 128 Гц или 256 Гц
- Счетчик текущего измерения (0–127/0–255)



<https://www.emotiv.com/>

# Алгоритм распознавания Р300



Параметры сессии:

- Количество повторений релевантного стимула
- TTI (target-to-target interval) – расстояние между стимулами
- Количество классов
  - Чем их меньше, тем больше вероятность релевантного стимула
  - Менее четко выраженный Р300

# Этап предобработки

- Полосовой фильтр 0.4 Гц – 30 Гц
- Подвыборка каждого второго второго измерения (понижение частоты дискретизации до 64 Гц)
- Удаление артефактов при помощи ICA
  - Артефакты обычно не коррелируют с сигналом
  - Удаляем несколько независимых компонент с наибольшей амплитудой

# Этап сегментации

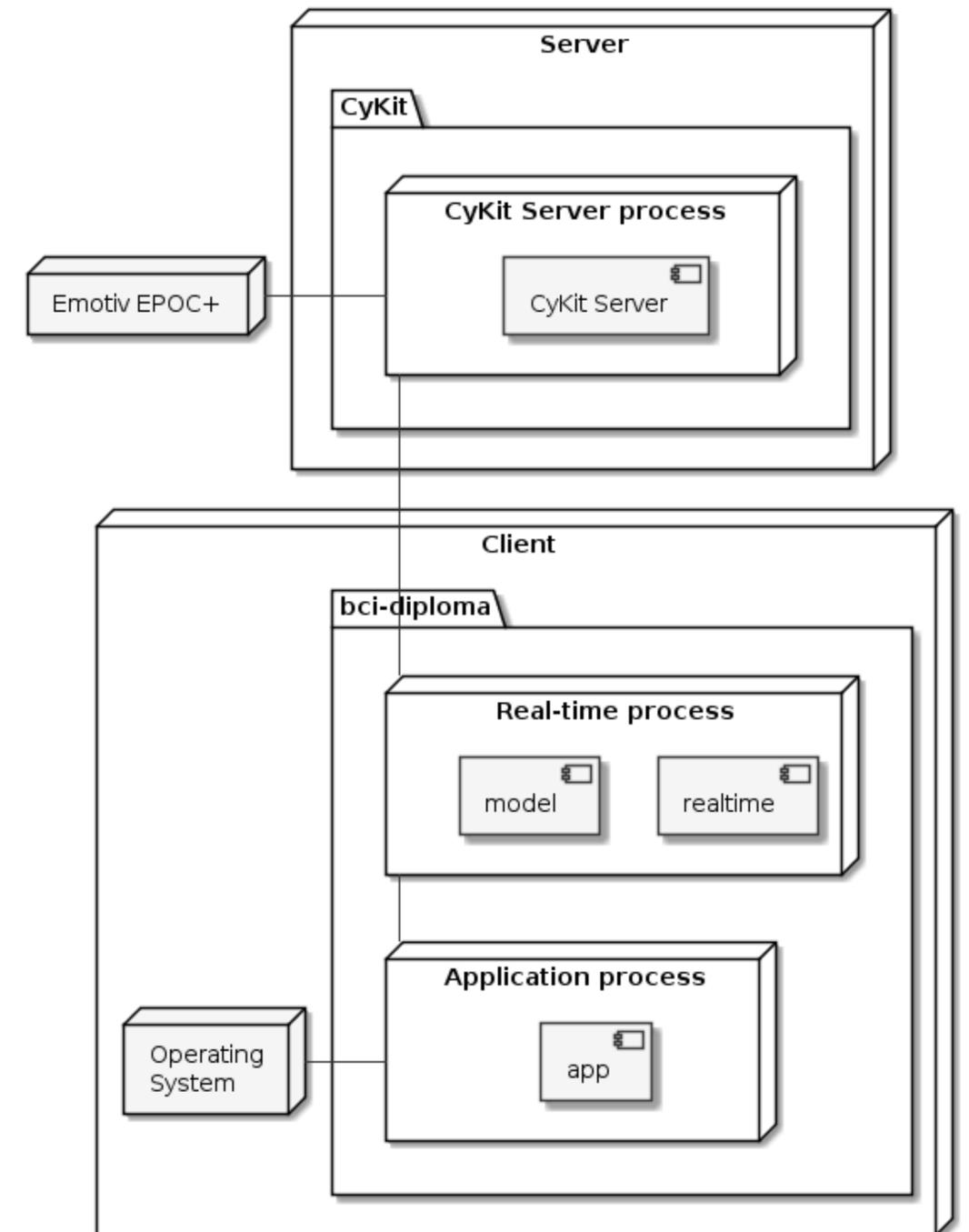
- Вырезается 1 с после стимула
- Отклики на стимулы одного класса усредняются
- Данные со всех каналов объединяются в один вектор признаков
- Полученный вектор признаков без изменений передается классификатору

# Опробованные классификаторы

- LDA
- Полносвязные нейронные сети
- SVM

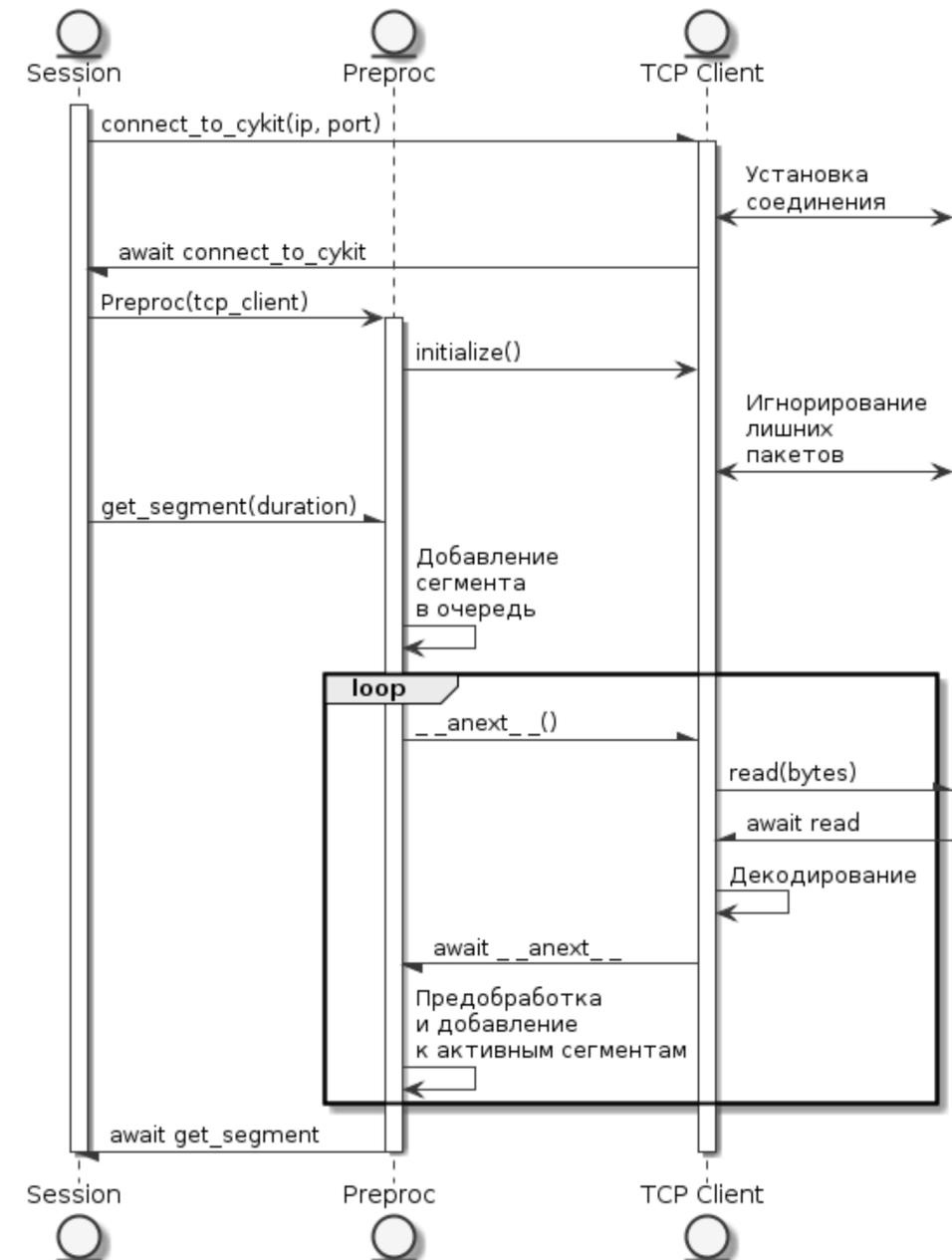
# Архитектура

- Сервер CyKit передает данные через TCP
- Разбиение логики решения на два процесса
  - Логика приложения (генерация стимулов, взаимодействие с ОС) – родительский
  - Логика реального времени (обработка сигнала, логика сессии) – дочерний
  - IPC на базе QTcpServer
- Асинхронные вызовы
  - app – Qt signals and slots
  - realtime – asyncio coroutines

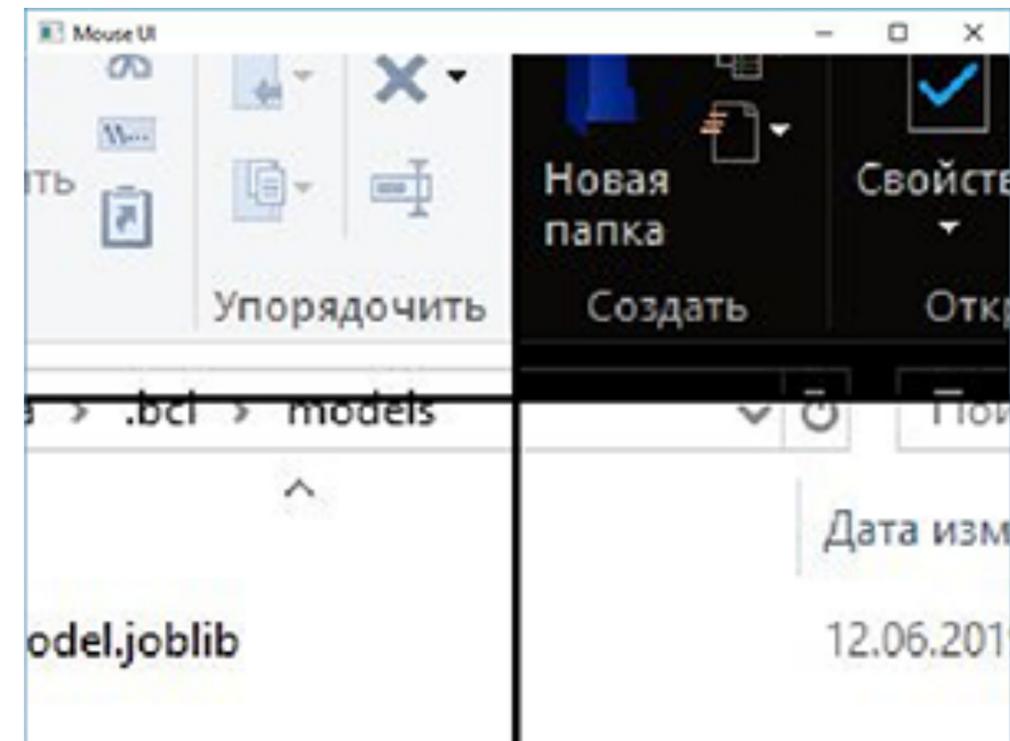
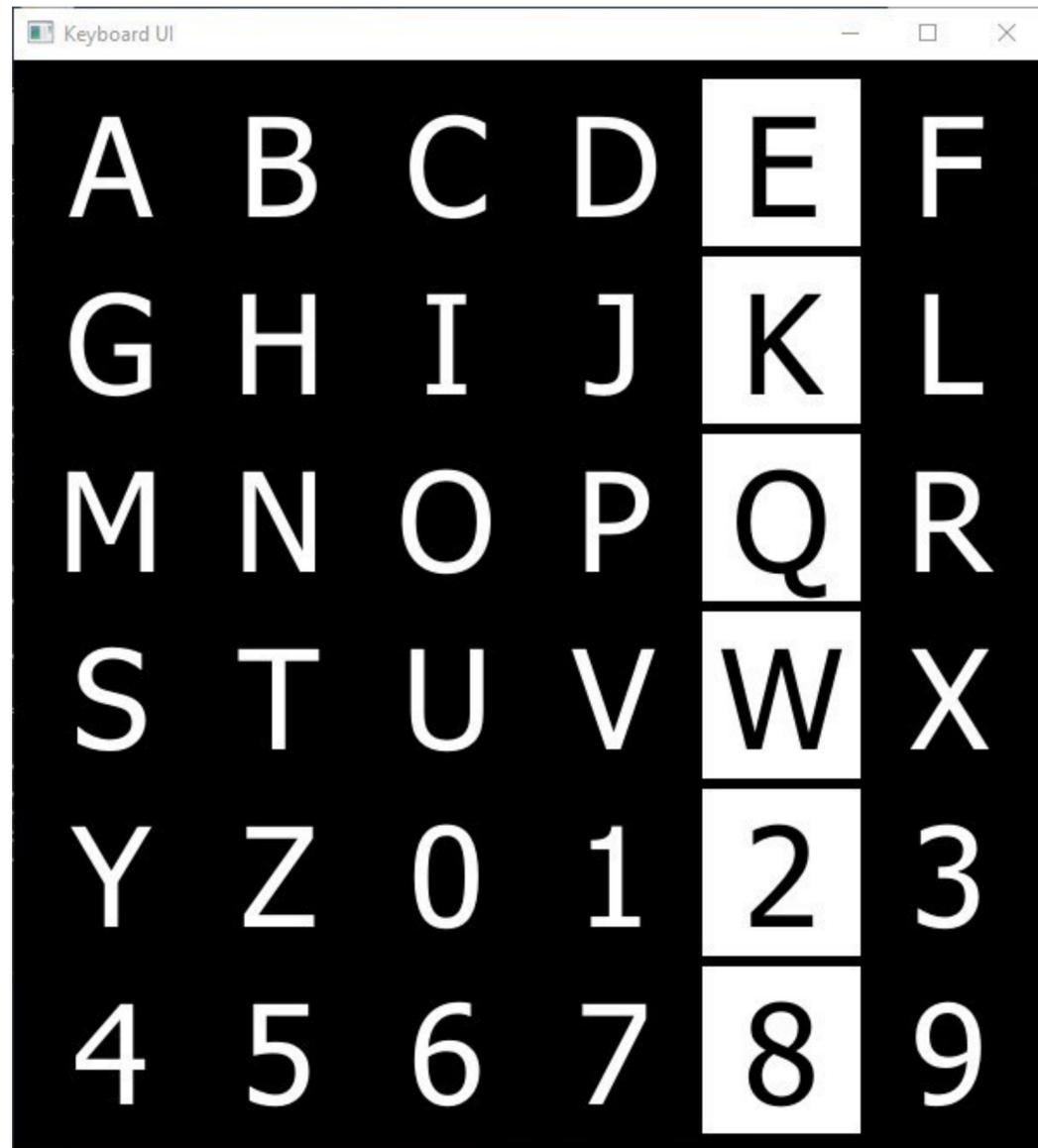


# Логика реального времени

- Логика сессии (класс `Session`)
  - Обеспечение интервалов между стимулами
  - Отправка сообщений родителю (генерация стимула, выбор класса)
  - Запросы сегментов
- Получение сегмента (класс `Preproc`)
  - Предобработка «на лету» сериями (`async for` над TCP-пакетами)
  - Асинхронная очередь сегментов
  - Игнорирование данных, если нет активных сегментов
- CPU-bound действия (взаимодействие с моделью) — в отдельном потоке



# Пользовательский интерфейс



# Реализация

- Язык разработки – Python 3.7
  - scipy
  - scikit-learn
  - Qt for Python (PySide2)
  - pywin32
- CyKit для получения данных ЭЭГ с устройства

# Апробация

- Задача – ввод текста при помощи P300, метрика – отношение правильных ответов к количеству испытаний (accuracy)
- Для каждого испытуемого имеются данные для обучения и данные для тестирования; accuracy усредняется по всем субъектам
- Алгоритм протестирован на данных BCI Competition III
  - Случайное гадание (математическое ожидание) – 2.8% accuracy
  - LDA + ICA, TTI = 100 мс, 5 повторений стимула – 56.5% accuracy
  - **LDA + ICA, TTI = 100 мс, 15 повторений стимула – 91.0% accuracy**
  - NN + ICA, TTI = 100 мс, 15 повторений стимула – 87.5% accuracy
  - SVM – менее 30% на необработанных дополнительно данных
- На своих данных (Emotiv EPOC+, два испытуемых) LDA + ICA – **57.5% accuracy**

# Результаты

- Осуществлен поиск решений, осуществляющих получение данных с Emotiv EPOC+
- Сформулированы требования к решению
- Реализован алгоритм распознавания P300, состоящий из фаз предобработки, сегментации и классификации
- Разработан прототип решения, позволяющего осуществлять ввод текста и позиционирование курсора на компьютере под управлением ОС Windows
- Произведена апробация с привлечением реальных испытуемых

Исходный код решения доступен на <https://github.com/oevseev/bci-diploma>.

# Ссылки

- BCI Competition III – <http://bbci.de/competition/iii/results/index.html>
- EmoKit – <https://github.com/openyou/emokit>
- CyKit – <https://github.com/CymatiCorp/CyKit>