

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ**
Профиль «Информационные технологии»
Командный кейс №1 «Подготовка фотошаблона»

1. Условия

Важной частью маршрута проектирования современных интегральных схем (ИС) в микроэлектронике является этап фотолитографии (рис. 1). Это процесс получения определенного рисунка на поверхности кремниевой пластины. Он начинается с нанесения на обрабатываемую поверхность тонкой фоточувствительной полимерной пленки (фоторезиста). Затем пленка засвечивается светом от специального источника через фотошаблон (маску) с заданным рисунком.

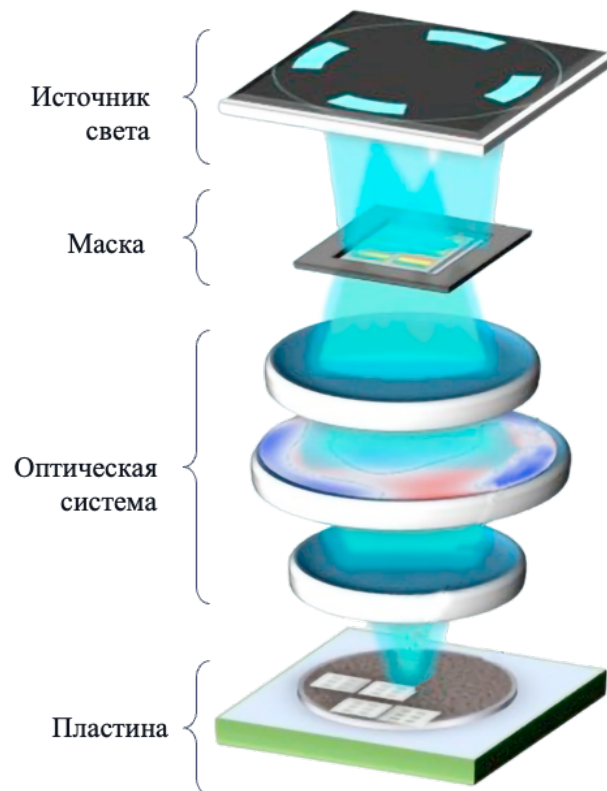


Рис. 1. Схема фотолитографии при производстве ИС

Ключевым фактором в указанном выше процессе является качество подготовки фотошаблона (рис. 2): из-за процессов дифракции и интерференции света структуры, проявленные на кремниевой подложке, искажаются. Чтобы этого избежать, фотошаблон модифицируется особым образом, приближая структуры к целевым. Такой процесс называется Optical Proximity Correction, или коррекция оптической близости.

МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ
Профиль «Информационные технологии»
Командный кейс №1 «Подготовка фотошаблона»



Рис. 2. Фотошаблон

На рисунках 3 и 4 схематично представлены результаты процесса фотолитографии с коррекцией оптической близости и без.



Рис. 3. Результат фотолитографии с коррекцией оптической близости



Рис. 4. Результат фотолитографии без коррекции оптической близости

Для подбора правильных алгоритмов модификации фотошаблона важно подготовить входные данные – структуры и их границы. Для решения этой задачи необходимо произвести высокоточную сегментацию (рис. 5) и выделение структур на общем фоне.

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ**
Профиль «Информационные технологии»
Командный кейс №1 «Подготовка фотошаблона»

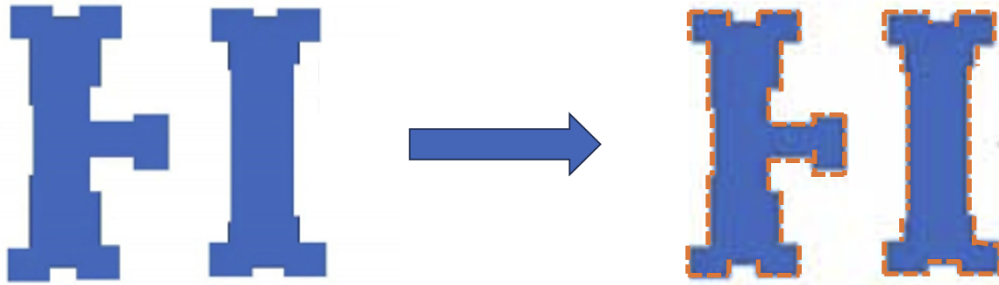


Рис. 5. Пример сегментации и выделения структур

2. Техническое задание

Разработать программное решение (приложение для Windows 10/Linux/macOS) сегментации, выделения и подсчета общего количества структур на фотошаблоне.

Функциональность программы:

- алгоритмы компьютерного зрения: сегментации, выделения и подсчета общего количества структур для формирования фотошаблона в формате изображения.

Входные данные:

- [фотошаблон](#) в формате JPG изображения 240x240 px.

Выходные данные:

- наложенные границы сегментов поверх входного изображения;
- CSV-файл с количеством выделенных сегментов на фотошаблоне (пример CSV-файла размещен по [ссылке](#)).

3. Рекомендации к выполнению

- использование системы контроля версий;
- ведение документации проекта (формат описания схем остается на усмотрение разработчиков);
- покрытие функционала модульным тестированием.

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ**
Профиль «Информационные технологии»
Командный кейс №1 «Подготовка фотошаблона»

4. Требования к документации

- Титульный лист (с указанием названия кейса и перечислением членов команды).
- Обоснование выбора языка программирования и используемых программных средств.
- Структурная и функциональная схемы программного продукта.
- Блок-схема работы основного алгоритма.
- Описание особенностей и аргументация выбранного типа СУБД.
- Схема базы данных.
- Программный код (ссылка на репозиторий), файл README должен включать:
 - краткое описание проекта;
 - инструкцию по установке/развертыванию;
 - ссылку на видеоролик.

5. Требования к видеоролику

- Видеоролик должен демонстрировать функционирование разработанного программного продукта в соответствии с регламентом испытаний.
- На видео или записи экрана необходимо продемонстрировать выполнение каждого испытания, описанного в регламенте, в соответствии с условиями.
- Видео должно однозначно подтверждать авторство участников (во время записи ролика необходимо четко произнести название команды, ФИО участников, номер школы, ФИО руководителя).
- Видеоролик необходимо разместить на стороннем видеохостинге («ВКонтакте», Rutube и др.)

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ**
Профиль «Информационные технологии»
Командный кейс №1 «Подготовка фотошаблона»

6. Регламент испытаний

1. Установка и запуск исполняемого файла приложения.
2. Выбор файла с фотошаблоном для дальнейшей сегментации.
3. Демонстрация визуализации результатов сегментирования для трех разных шаблонов.
4. Выгрузка результатов в выходное изображение и соответствующий CSV-файл для трех разных шаблонов.
5. Корректное завершение работы приложения на любых этапах работы.

7. Методические материалы

- <https://www.python.org>;
- <https://opencv.org/>;
- <https://pypi.org/project/opencv-python>;
- <https://github.com/AlexeyAB/darknet>;
- язык программирования Python;
- библиотека OpenCV для работы с компьютерным зрением;
- библиотека машинного обучения TensorFlow;
- детектор YOLO.