

**ИНФОРМАТИКА, ВЫЧИСЛИТЕЛЬНАЯ ТЕХНИКА И УПРАВЛЕНИЕ**

УДК 004.93'1

DOI 10.52928/2070-1624-2024-43-2-2-9

**ПРИНЦИПЫ ПРОЕКТИРОВАНИЯ И РАЗРАБОТКИ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ  
ДЛЯ ОБОРУДОВАНИЯ КОНТРОЛЯ КАЧЕСТВА ТОПОЛОГИЧЕСКИХ СТРУКТУР  
В МИКРОЭЛЕКТРОНИКЕ**

*д-р техн. наук, доц. С. М. АВАКОВ*  
(ОАО «Планар», Минск);

*канд. техн. наук, доц. А. А. ВОРОНОВ, канд. техн. наук В. В. ГАНЧЕНКО*  
(Объединенный институт проблем информатики НАН Беларуси, Минск)

*Изложены принципы создания программного обеспечения для установок, применяемых в микроэлектронной промышленности для проверки качества топологических структур, играющих ключевую роль в технологическом процессе производства установок контроля критических размеров и автоматического поиска дефектов. Предложен оригинальный алгоритм выбора функции резкости и проведены экспериментальные исследования, подтверждающие его эффективность для получения качественных исходных изображений топологии микросхем. Это обеспечивает высокую точность выполнения измерений установкой контроля критических размеров с использованием системы технического зрения и, как результат, повышает процент выхода годных изделий в микроэлектронике.*

**Ключевые слова:** СБИС, архитектура программного обеспечения систем технического зрения, установки контроля критических размеров в микроэлектронике.

**Введение.** Постоянно растущие требования к надежности, быстродействию и компактности элементов и узлов электронных изделий требуют непрерывной разработки и совершенствования микроэлектронной цифровой техники. Современные средства разработки электронных схем направлены на сокращение времени освоения и производства новых изделий. Использование установок автоматического обнаружения дефектов и контроля критических размеров способствует снижению стоимости цифровой аппаратуры при ее массовом производстве. Такую возможность обеспечивают системы технического зрения (СТЗ), которые являются составной частью современной технологии проектирования и производства интегральных схем (ИС). В отечественной литературе отсутствуют исследования по разработке обобщенных архитектур программного обеспечения установок, применяемых в комбинированной технологии, позволяющей в едином производственном цикле выполнять как анализ ИС, так и ее ремонт и повторный контроль наличия дефектов, а также проверку выполнения проектных норм изготовления ИС при проектировании и перепроектировании.

**Обзор решений.** При проектировании программного обеспечения (ПО) для сложного технологического оборудования, применяемого в микроэлектронной промышленности, как правило, рассматривают примеры уже существующего аппаратного обеспечения конкретных установок и их систем обработки и анализа изображений, а затем формулируют требования к архитектуре разрабатываемого ПО для контрольно-измерительного оборудования изделий микроэлектроники. Важным вопросом является также применение в подобном оборудовании систем технического зрения с различными способами фокусировки, т. к. без них невозможно осуществлять эффективный контроль дефектов топологии (необходимо всегда получать резкое изображение). Существуют как зарубежные установки контроля критических размеров и автоматического обнаружения дефектов полупроводниковых пластин (например, от компании MueTec или KLA-Tencor Corporation [1]), так и их отечественные аналоги от предприятия-производителя ОАО «Планар» [2]. Предварительно исследуем аппаратное обеспечение следующих установок, поддерживающих оборудование видеозахвата, от сторонних производителей: ЭМ-6015М, предназначенной для обнаружения дефектов и загрязнений на поверхности полупроводниковых пластин и фотошаблонов; ЭМ-6239, предназначенной для автоматизированного контроля микроразмеров на автоматически загружаемых полупроводниковых пластинах; ЭМ-6339, предназначенной для контроля критических размеров объектов топологии фотошаблонов и используемой в технологическом процессе производства микроэлектромеханических систем. Кроме того, рассмотрим разработанный новый объектив в установке ЭМ-6339, ахроматизированный для различных длин волн, что позволило улучшить воспроизводимость измерения координат на порядки, исключить подвижные детали в оптической части установки и образование пыли на подложке. Внешний вид установок приведен на рисунке 1.



Рисунок 1. – Контрольно-измерительное оборудование ОАО «Планар» с автофокусировкой

Использованное в установке ЭМ-6339 техническое решение позволило наблюдать объекты в видимом и ультрафиолетовом спектрах. Параметры объектива для установки ЭМ-6339:  $Na = 0,82$ ; рабочий отрезок – 7,5 мм;  $\lambda = 365$  нм, 590 нм, 670 нм. Диапазон контролируемых элементов для данной установки по  $X$  и  $Y$  – 0,5–35 мкм, случайная составляющая погрешности при контроле размеров ( $3\sigma$ ) – не более 6 нм; максимальное увеличение – 6000× крат. На рисунке 2 приведена иллюстрация спектра.

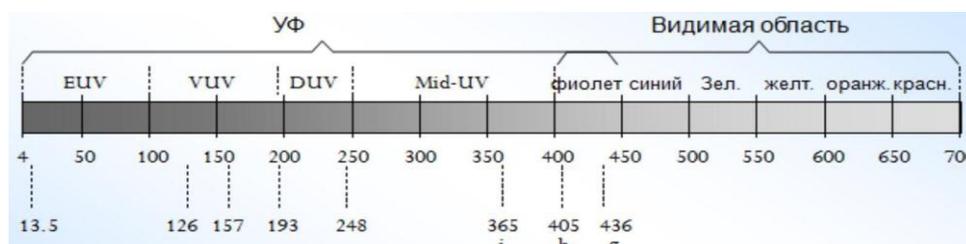


Рисунок 2. – Спектр, в котором работает контрольно-измерительное оборудование ОАО «Планар»

Функциональная структура процессов обработки информации в ПО перечисленных установок представляется ориентированным графом, в вершинах которого располагаются процессоры обработки данных, а дуги обеспечивают транспорт данных между ними. Архитектура программного обеспечения СТЗ представлена на рисунке 3. Для ее визуализации применена нотация, разработанная Г. Бучем [3]. Здесь показаны два типа связей между объектами (экземплярами классов системы). Классы по своему назначению соответствуют блокам функциональной схемы (для построения диаграмм было использовано средство Rational Rose / C++ Demo Version 4.0.3 фирмы Rational Software Corp.) [4].

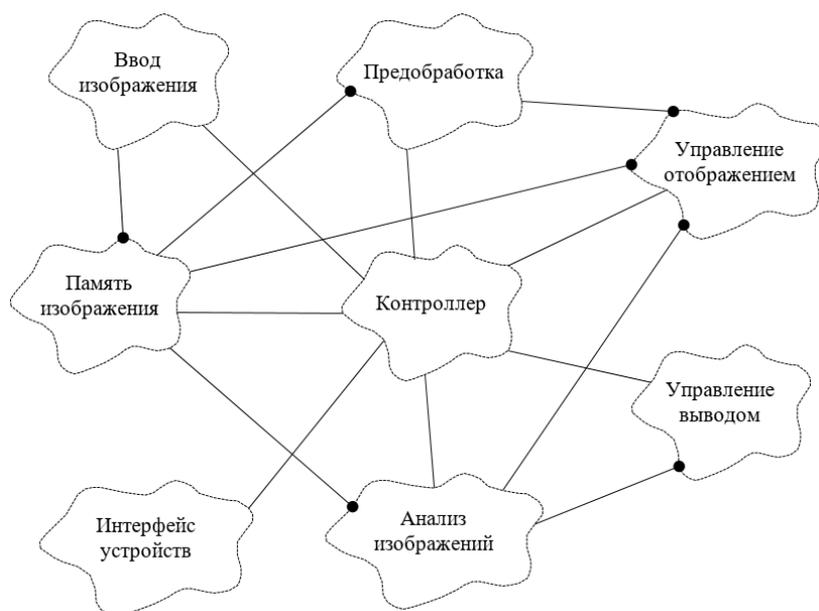


Рисунок 3. – Архитектура программного обеспечения СТЗ

Пример клиент-серверных отношений в модели: объект класса «Управление отображением» (ответственный за отображение информации в процессе обработки) использует данные, предоставляемые объектами классов «Предобработка» (соответствующих блоку предварительной обработки), «Память изображения» и «Анализ изображения».

СТЗ входит в состав рассматриваемого оборудования контроля качества топологических структур в микроэлектронике. Основные используемые в СТЗ базовые методы обработки изображений можно классифицировать [5–9] следующим образом: для формирования изображений (алгоритмы автоматической настройки системы ввода, алгоритмы ввода изображений, алгоритмы первичной обработки); для сегментации изображений (нахождение контуров, операции над однородными областями, декомпозиция изображений); для описания изображений (алгоритмы вычисления геометрических признаков и формальных инвариантов, алгоритмы вычисления ориентации объектов и их локализации); для анализа изображений и распознавания объектов (алгоритмы вычисления дискриминантных функций, структурно-синтаксические алгоритмы и построение модели-описания). Описан лишь один из вариантов разработки СТЗ на основе объектно-ориентированного проектирования [10]. На этапе подготовки к проектированию ПО для контрольно-измерительного оборудования изделий микроэлектроники необходимо пройти основные этапы создания СТЗ:

1. Получение и подготовка изображений для дальнейшей оценки и измерений (установка и настройка системы отображения, калибровка системы отображения, определение формата изображения, извлечение и чтение изображения, отображение изображения, сопоставление калибровочной информации, анализ изображения, улучшение изображения).

2. Идентификация объектов топологии на изображении различными способами (с помощью полутоновых или цветных мер, анализа отдельных частей изображения, методов машинного обучения).

3. Измерение необходимых параметров и характеристик на изображениях объектов топологии.

На основе анализа вариантов использования контрольно-измерительного оборудования в технологическом процессе производства микросхем и управлении качеством (рисунок 4) были сформулированы следующие требования к архитектуре ПО данного оборудования:

- модульность – программное обеспечение должно представлять собой совокупность динамически загружаемых модулей (библиотек), реализующих определенный набор функций и имеющих заданный программный интерфейс, система обработки и анализа изображений должна быть программно реализована в виде модуля;
- открытость – программный модуль должен иметь возможность расширения набора доступной функциональности и список поддерживаемых устройств (установок);
- конфигурируемость – должна быть реализована возможность настройки работы с конкретными устройствами и выбора некоторой последовательности использования функциональности без исходных кодов (без перекомпиляции);
- разделение графического интерфейса пользователя и функциональности – требуется для возможности работы программного модуля без участия пользователя согласно заданной конфигурации.



Рисунок 4. – Контрольно-измерительное оборудование ОАО «Планар» для системы управления качеством полупроводниковых пластин

Наиболее удобным шаблоном для проектирования и создания программного обеспечения контрольно-измерительного оборудования, который бы соответствовал перечисленным требованиям, является MVC [11; 12]:

1. MVC подразумевает разбиение на логические модули, каждый из которых может иметь свою модульную структуру, скрываемую за единым интерфейсом.

2. Открытость следует из возможности реализации иерархической модульности: каждый элемент может иметь требуемое количество внутренних модулей, единственным требованием к которым является соответствие определенному программному интерфейсу.

3. Разделение отображения и управления позволяет формировать наборы команд, передаваемых системе управления, и использовать их вместо реакции на действия пользователя в графическом интерфейсе, что обеспечивает возможность реализации конфигурируемости. Шаблон MVC подразумевает подобное разделение.

Недостатком MVC для обобщенной архитектуры контрольно-измерительного оборудования изделий микроэлектроники является отсутствие четко выделенной модели данных. Однако если объектом исследования является *процесс контроля и измерения объектов топологии изделий микроэлектроники*, то данными можно считать информацию о состоянии этого процесса.

Приведем пример проектирования при помощи архитектурного шаблона MVC одного из модулей программного комплекса системы управления установки контроля критических размеров (ПК УОККР) для производства изделий микроэлектроники, имеющего следующие возможности:

1. ПО системы управления имеет возможность получать команды из нескольких источников – добавляется *интерфейс приема управляющих команд*, в общем случае преобразующий конкретные команды в команды общего вида (метакоманды).

2. ПО системы управления имеет возможность получать видеопоток из нескольких источников – добавляется *интерфейс получения видеоданных*, преобразующий различные форматы данных к единому формату.

3. ПО системы управления имеет возможность передавать управляющие команды различным аппаратным модулям установки (механизмам) – добавляется *интерфейс управления оборудованием*. Этот интерфейс принимает команды управления оборудованием общего формата (метакоманды оборудования) и преобразует их к конкретному виду;

4. Реализована *виртуальная модель данных*, хранящая описание текущего состояния процесса контроля параметров.

#### **Основные прецеденты использования ПК УОККР и его модуля управления механизмами:**

– загрузка объекта исследования – подготовка объекта исследования для дальнейшей работы (предварительная ориентация в пространстве, перемещение в рабочую зону);

– выгрузка объекта исследования – удаление объекта исследования из рабочей зоны в хранилище (контейнер, кассету);

– инициализация установки и базирование механизмов – загрузка в установку данных, описывающих исходное состояние оборудования для решения конкретной задачи, и установка механизмов в соответствующее состояние/положение;

– управление перемещениями координатного стола – формирование обобщенных команд для управления движением координатного стола;

– совмещение и ориентация – привязка системы отсчета и координатной системы объекта к координатной системе установки;

– управление механизмами – подготовка команд управления оборудованием (метакоманд) и соответствующих параметров;

– формирование управляющих команд – преобразование метакоманд в формат, требуемый тем либо иным микроконтроллером управления оборудованием;

– создание карты-структуры объекта (т. к. объектом может служить как пластина, так и шаблон) – подготовка описания структуры объекта для дальнейшего его исследования (кадры, модули).

**Архитектура модуля управления механизмами** представлена в виде диаграммы компонентов, приведенной на рисунке 5.

Модуль управления механизмами состоит из следующих компонентов:

– фасад для SDK-микроконтроллеров – набор фасадных классов, обеспечивающих прозрачное и, насколько это возможно, единообразное выполнение различных команд с использованием реального оборудования видеозахвата. Каждый отдельный фасад разрабатывается для и с учетом особенностей конкретного оборудования;

– фасады блоков формирования последовательностей метакоманд – набор реализаций программного интерфейса `HardwareController`, предоставляющего единообразный набор методов для осуществления управления оборудованием и одновременно позволяющего скрывать особенности конкретного оборудования (методы для управления загрузкой/выгрузкой объекта исследования, инициализацией установки, перемещениями координатного стола; методы для получения статуса выполнения команды, данных или изображения от установки);

- блок формирования последовательностей метакоманд – осуществляет формирование пакетов метакоманд, выполняющих некоторые стереотипные последовательности действий, которые обеспечивают выполнение заданных пользователем действий;
- блок преобразования метакоманд в команды оборудования – преобразует метакоманды с указанными пользователем параметрами в команды конкретного оборудования (или последовательности вызовов SDK) с параметрами, характерными для этого оборудования;
- блок обратной связи – обеспечивает получение данных от оборудования (изображения, параметры состояния оборудования и т. п.), осуществляя соответствующие вызовы функциональности SDK-оборудования. При разработке данного модуля используется понятие «метакоманда». Это понятие предполагает некоторую задачу управления механизмами высокого уровня, более понятную пользователю комплекса. При этом обеспечение выполнения этой команды может потребовать выполнения нескольких команд управления на конкретном оборудовании. Примеры метакоманд: «получить следующее изображение», «сместиться в заданную точку» и т. п.

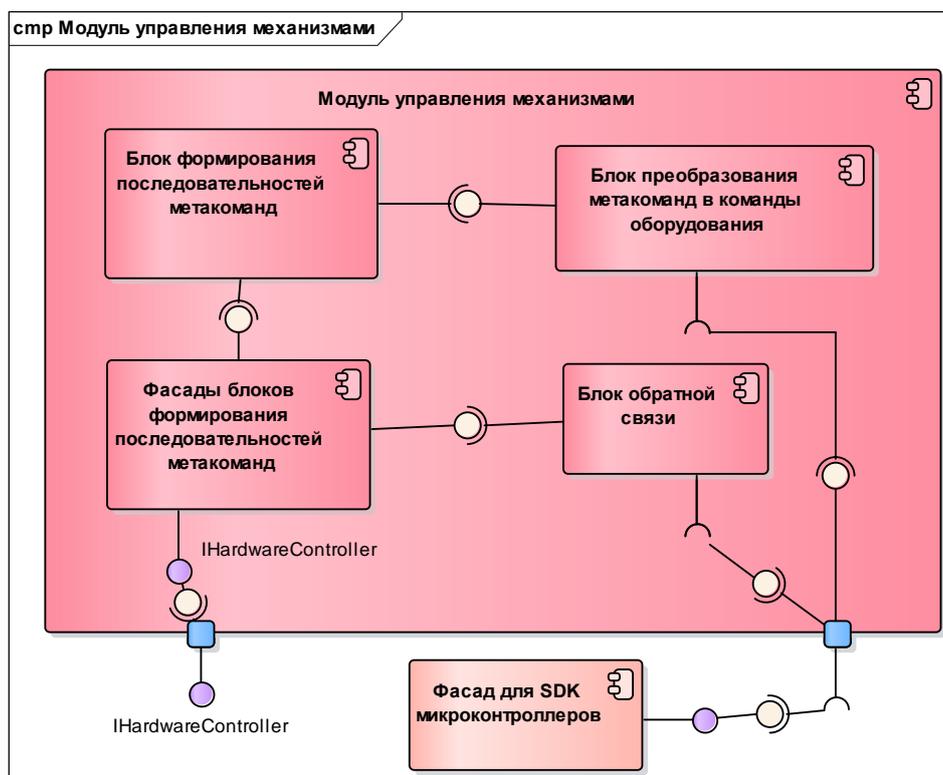


Рисунок 5. – Архитектура ПО модуля управления механизмами для установки контроля критических размеров полупроводниковых пластин

Для успешного поиска дефектов и контроля качества топологии необходимо получение качественного изображения оборудованием видеозахвата. Для этого используется автофокусировка с использованием функции резкости.

**Автоматическая фокусировка и функция резкости.** Актуальность автоматической фокусировки связана с тем, что без нее невозможно в принципе осуществить автоматический контроль изделий, поскольку определяющую роль в процессе контроля играет непрерывное поддержание резкого изображения. При этом могут достигаться различные цели контроля: поиск привнесенных дефектов, регистрация искажений топологического рисунка или контроль размеров элементов структур и т. п. Система автоматической фокусировки позволяет определить такое положение контролируемого образца относительно рабочего объектива, при котором изображение образца формируется объективом с распределением световой энергии, максимально схожим с распределением в самом образце, что особенно важно для достоверной регистрации топологического рисунка на полупроводниковой пластине.

Автоматическая фокусировка по изображению основывается на итеративных методах с использованием функции резкости (ФР), реальной оценки резкости дискретного изображения. Для серии изображений, используемой для фокусировки, ФР должна достигать максимумов для корректно сфокусированного изображения.

ФР дают значение, указывающее степень фокусировки для каждого изображения того же объекта. Эти функции основаны на знаниях о различиях в информационном наполнении в сфокусированных и несфокусированных изображениях [13; 14]. Основное предположение, лежащее в основе большинства этих функций, состоит в том, что расфокусированное изображение является результатом свертки изображения с определенной функцией распространения точек (point-spread function, PSF), которая обычно приводит к уменьшению высоких частот изображения. Хорошо сфокусированные изображения содержат больше информации и деталей (краев), чем несфокусированные. ФР можно разделить на несколько групп: функции, основанные на дифференцировании изображения (градиенте изображения); функции, основанные на глубине пиков и долин; функции, основанные на контрасте изображения; функции, основанные на гистограмме; функции, основанные на статистической информации; функции на основе Фурье-преобразования.

**Алгоритм оценки функции резкости для фокусировки.** Важными алгоритмами, определяющими в значительной степени качество исходных изображений для анализа и обработки, являются алгоритмы автофокусировки. Алгоритмы, выбирающие оптимальный фокус для изображения, основаны на нахождении максимума или же минимума определенной функции резкости, которая представляет собой «показатель метрики качества». Такие алгоритмы должны быть быстрыми, поскольку обычно общее время сканирования изображения ограничено.

Существуют десятки различных функций резкости. Некоторые из них применимы только в решении узконаправленных задач, некоторые достаточно универсальны. Далее будут рассмотрены функции резкости, используемые для обработки видео и изображений. Напомним, что функции резкости – это функции, которые достигают экстремума, когда исследуемый объект в фокусе.

При выборе функции резкости для решения конкретной задачи выполняется оценка ряда функций резкости путем количественной оценки нескольких критериев на конкретных сериях изображений. Могут быть применены следующие критерии:

- точность – это разница в шагах между максимумом функции резкости и правильным фокальным положением (полученным опытным оператором);
- диапазон – это интервал (измеренный от фокального положения), где функция монотонно уменьшается;
- число ложных максимумов;
- полная ширина при 50 % от максимума пика в функции резкости: чем она меньше, тем острее пик фокуса и тем точнее его можно определить;
- время вычисления значения ФР.

Алгоритм предназначен для определения оптимальной функции резкости из заданного набора функций для автоматической фокусировки по изображениям топологии ИС.

На вход поступает последовательность полутоновых изображений и список функций резкости: *th\_grad* – функция резкости на основе дифференцирования изображения; *sq\_grad* – аналогична *th\_grad*, только рассматривает разность квадратов между пикселями; *abs\_ten* – функция резкости, использующая оператор Собеля для свертки изображения; *gauss* – функция резкости, основанная на градиентном фильтре; *absolut* – функция на основе глубины пиков и долин; *variance* – функция резкости на основе контраста изображения; *norm\_variance* – нормализованная функция резкости на основе дисперсии; *dif\_range* – функция на основе гистограммы изображения; *entropy* – функция энтропии.

На выходе алгоритм выдает ранжированный список функций резкости.

Алгоритм состоит из следующих шагов:

1. Задать шаг пересчета функции резкости.
2. Задать вес критериев оценки.
3. Сформировать таблицу сравнения: столбцы соответствуют функциям резкости, строки соответствуют изображениям в последовательности.
4. Задать размер области фокусировки.
5. Выбрать изображения из последовательности.
6. Вычислить значение функции резкости по области фокусировки на выбранном изображении.
7. Сохранить значение в таблицу сравнения.
8. Повторять шаг 5, пока не конец последовательности изображений.
9. Вычислить критерии оценки для всех функций резкости.
10. Ранжировать функции резкости по вычисленным критериям с учетом заданных весов.

Результаты вычисления значений лучшей из функций резкости *sq\_grad* для набора из 23 изображений топологии ИС приведены на рисунке 6. По горизонтали обозначены номера кадров от 0 до 23, а по вертикали – значения метрики *sq\_grad*, которая представляет собой возведенный во вторую степень порогированный абсолютный градиент, являющийся лучшей функцией резкости для рассмотренного набора изображений и основанный на дифференцировании изображения. Первая разность вычисляется, когда она

больше определенного порога, возводится во вторую степень и накапливается так, что большие градиенты оказывают большее влияние на конечный результат:

$$F_{sq\_grad} = \sum_M \sum_N |g(i, j+1) - g(i, j)|^2,$$

$$\text{для } |g(i, j+1) - g(i, j)| \geq v,$$

где  $M$  и  $N$  – ширина и высота изображения соответственно;  
 $g(i, j)$  – яркость пикселя  $(i, j)$ ;  
 $v$  – заданный порог.

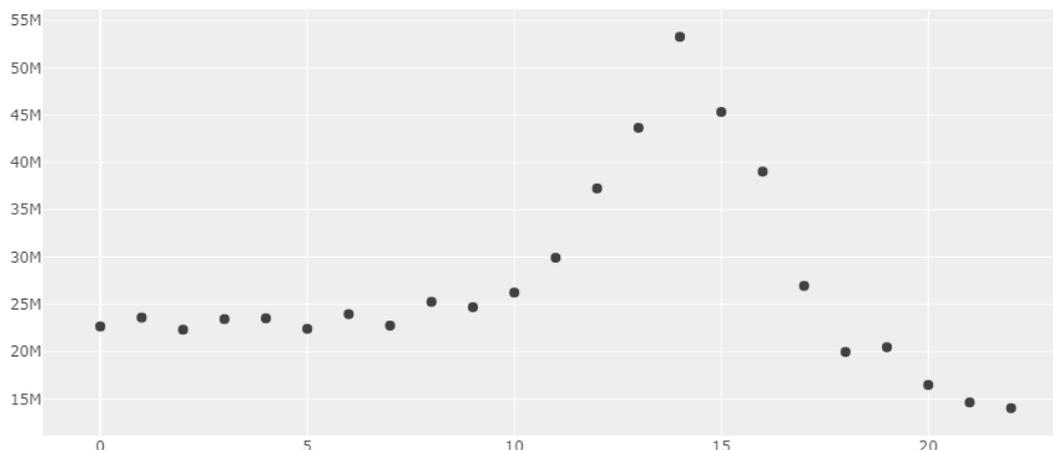


Рисунок 6. – График функции резкости для 23 исследуемых кадров с изображениями топологии полупроводниковой пластины

**Заключение.** Разработаны базовые принципы проектирования программного обеспечения оборудования контроля полупроводниковых структур с использованием системы технического зрения на основе вариантов использования такого оборудования в технологическом процессе производства микросхем и необходимое алгоритмическое обеспечение для автоматической фокусировки оборудования видеозахвата с использованием функции резкости. Приведен пример использования разработанной архитектуры при реализации программного модуля управления механизмами для программного комплекса установки контроля критических размеров полупроводниковых пластин. Разработка успешно используется при проектировании программного обеспечения для оборудования контроля полупроводниковых пластин и в комплексном проектировании оборудования, производимого ОАО «Планар». Это обеспечивает следующие преимущества: возможность получения более полной совместимости координатных систем установок за счет применения однотипных алгоритмов управления перемещениями, возможность гибкого изменения форматов данных для установок автоматического контроля дефектов, возможность реализации единой интерфейсной системы ПО и повышения эффективности эксплуатации оборудования за счет уменьшения стоимости разработки и изготовления вследствие высокой степени унификации аппаратного и программного обеспечения, используемого для системы управления качеством полупроводниковых пластин.

#### ЛИТЕРАТУРА

1. Оптико-механические комплексы для бездефектного изготовления фотошаблонов 0,35 мкм и 90 нм / С. Аваков, В. Овчинников, С. Карпович и др. // Фотоника. – 2007. – № 6. – С. 35–39.
2. Фалк Г. Гибкие решения для оптического контроля качества // Наноиндустрия. – 2018. – № 1(80). – С. 24–26. – DOI: [10.22184/1993-8578.2018.80.1.24.26](https://doi.org/10.22184/1993-8578.2018.80.1.24.26).
3. Буч Г. Объектно-ориентированный анализ и проектирование с примерами приложений на C++. – СПб.: Невский диалект, 1999. – 560 с.
4. Боггс У., Боггс М. UML и Rational Rose. – М.: Лори, 2000. – 580 с.
5. Сойфер В. А. Методы компьютерной обработки изображений. – М.: Физматлит, 2003. – 192 с.
6. Гонсалес Р., Вудс Р. Цифровая обработка изображений. – М.: Техносфера, 2005. – 1072 с.
7. Дудкин А. А., Садыхов Р. Х. Обработка изображений в проектировании и производстве интегральных схем. – Минск: ОИПИ НАН Беларуси, 2008. – 270 с.
8. Распознавание и анализ стохастических данных и цифровых изображений / С. В. Абламейко, Ю. С. Харин, Р. Х. Садыхов и др. // Вестн. Фонда фундам. исслед. – 2003. – № 4. – С. 91–106.

9. Krasnoproshin, V. V., Obraztsov V. A. Problems of Solvability and Choice of Algorithms for Decision Making by Precedence // *Pattern Recognit. Image Anal.* – 2006. – Vol. 16, iss. 2. – P. 155–169. – DOI: [10.1134/S1054661806020027](https://doi.org/10.1134/S1054661806020027).
10. Приемы объектно-ориентированного проектирования. Паттерны проектирования / Э. Гамма, Р. Хелм, Р. Джонсон и др. – СПб.: Питер, 2001. – 368 с.
11. Ларман К. Применение UML и шаблонов проектирования. – М.: Вильямс, 2002. – 624 с.
12. Кнут Д. Э. Искусство программирования: в 4 т. – СПб.: Вильямс, 2000. – Т. 3: Сортировка и поиск. – 832 с.
13. Tian Q., Fainman I., Lee S. H. Comparison of statistical pattern-recognition algorithms for hybrid processing. II. Eigenvector-based algorithm // *J. Optical Society America A.* – 1988. – Vol. 5, iss. 10. – P. 1670–1682. – DOI: [10.1364/JOSAA.5.001670](https://doi.org/10.1364/JOSAA.5.001670).
14. Automatic PCB Inspection Algorithms: A Survey / M. Moganti, F. Erçal, C. H. Dagli et al. // *Comput. Vis. Image Underst.* – 1996. – Vol. 63, iss. 2. – P. 287–313. – DOI: [10.1006/cviu.1996.0020](https://doi.org/10.1006/cviu.1996.0020).

## REFERENCES

1. Avakov, S., Ovchinnikov, V., Karpovich, S., Titko, E., & Trapashko, G. (2007). Optiko-mehanicheskie komplekсы dlja bezdefektnogo izgotovlenija fotoshablonov 0,35 mkm i 90 nm. *Fotonika*, (6), 35–39. (In Russ.).
2. Falk, G. (2018). Gibkie reshenija dlja opticheskogo kontrolja kachestva. *Nanoindustrija*, (1), 24–26. DOI: [10.22184/1993-8578.2018.80.1.24.26](https://doi.org/10.22184/1993-8578.2018.80.1.24.26). (In Russ.).
3. Buch, G. (1999). Ob"ektno-orientirovannyj analiz i proektirovanie s primerami prilozhenij na C++. St. Petersburg: Nevskij dialekt. (In Russ.).
4. Boggs, U., & Boggs, M. (2000). *UML i Rational Rose*. Moscow: Lori. (In Russ.).
5. Soifer, V. A. (2003). *Metody komp'yuternoj obrabotki izobrazhenii*. Moscow: Fizmatlit. (In Russ.).
6. Gonsales, R., & Vuds, R. (2005). *Tsifrovaya obrabotka izobrazhenii*. Moscow: Tekhnosfera. (In Russ.).
7. Dudkin, A. A., & Sadykhov, R. Kh. (2008). *Obrabotka izobrazhenii v proektirovanii i proizvodstve integral'nykh skhem*. Minsk: UIIP NAS Belarus. (In Russ.).
8. Ablameiko, S. V., Kharin, Yu. S., Sadykhov, R. Kh., Starovoitov, V. V., & Tuzikov, A. V. (2003). Raspoznavanie i analiz stokhasticheskikh dannyx i tsifrovyykh izobrazhenii. *Vestnik Fonda fundamental'nykh issledovanii*, (4), 101–106. (In Russ.).
9. Krasnoproshin, V. V., & Obraztsov, V. A. (2006). Problems of Solvability and Choice of Algorithms for Decision Making by Precedence. *Pattern Recognit. Image Anal.*, 16(2), 155–169. DOI: [10.1134/S1054661806020027](https://doi.org/10.1134/S1054661806020027).
10. Gamma, E., Khelm, R., Dzhonson, R., & Vliissides, Dzh. (2001). *Priemy ob"ektno-orientirovannogo proektirovaniya. Patterny proektirovaniya*. St. Petersburg: Piter. (In Russ.).
11. Larman, K. (2002). *Primenenie UML i shablonov proektirovaniya*. Moscow: Vil'yams. (In Russ.).
12. Knut, D. E. (2000). *Iskusstvo programmirovaniya: v 4 t. T. 3: Sortirovka i poisk [The Art of Computer Programming (in 4 vol., Vol. 3: Sorting and Searching)]*. – St. Petersburg: Vil'yams. (In Russ.).
13. Tian, Q., Fainman, I., & Lee, S. H. (1988). Comparison of statistical pattern-recognition algorithms for hybrid processing. II. Eigenvector-based algorithm. *J. Optical Society America A*, 5(10), 1669–1681. DOI: [10.1364/JOSAA.5.001670](https://doi.org/10.1364/JOSAA.5.001670).
14. Moganti, M., Erçal, F., Dagli, C. H., & Tsunekawa, S. (1996). Automatic PCB Inspection Algorithms: A Survey. *Comput. Vis. Image Underst.*, 63(2), 287–313. DOI: [10.1006/cviu.1996.0020](https://doi.org/10.1006/cviu.1996.0020).

Поступила 05.11.2024

## SOFTWARE DESIGN PATTERN FOR EQUIPMENT OF AUTOMATIC MASK INSPECTION SYSTEMS IN MICROELECTRONICS

S. AVAKOV

(Planar JSC, Minsk);

A. VORONOV, V. GANCHENKO

(United Institute of Informatics Problems of NAS of Belarus, Minsk)

*The paper investigates a relevant applied problem associated with software development for building equipment of automatic mask inspection systems and quality control topological structures in the microelectronics industry. This problem is one of key issues of design flow for equipment of automatic mask inspection systems and automatic defects detection. An original approach is proposed for selecting the sharpness function. Experiments have been conducted that confirm the effectiveness of original approach for obtaining high-quality initial images in equipment of automatic mask inspection systems using a technical vision system and, as a result, increases the percentage of yield of suitable products in microelectronics.*

**Keywords:** VLSI, software architecture of machine vision systems, systems for inspection of work masks in microelectronics.