

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РФ  
Федеральное государственное бюджетное образовательное  
учреждение высшего образования  
«ПЕНЗЕНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

Факультет  
информационных технологий  
и электроники  
наименование факультета

Кафедра  
Нано – и микроэлектроника  
наименование кафедры

Направление подготовки

11.04.04. «Электроника и наноэлектроника»  
код и наименование

Магистерская программа

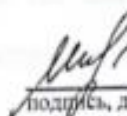
Нано – и микроэлектроника  
наименование

МАГИСТЕРСКАЯ ДИССЕРТАЦИЯ

на тему

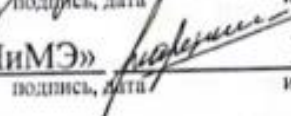
Элементы интегральных видеопроцессоров средств технического зрения для  
автоматизации механических угломеров  
тема магистерской диссертации

Студент группы 24ПДм1  
номер группы

  
подпись, дата

М.Р. Бакаева  
инициалы, фамилия

Руководитель к.т.н., доцент кафедры «НимЭ»  
должность

  
подпись, дата

Ю.А. Вареник  
инициалы, фамилия

Рецензент

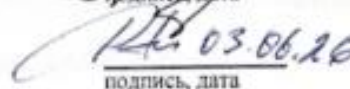
начальник отдела

АО «ПО «Электроприбор»  
должность, наименование организации

  
подпись, дата

А.В. Комиссаров  
инициалы, фамилия

Нормоконтролер ст. преподаватель  
должность

  
подпись, дата

Т.Н. Рыжова  
инициалы, фамилия

Работа допущена к защите в ГЭК (протокол заседания кафедры от 10.06.26 № 10)

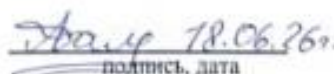
Заведующий кафедрой

  
подпись, дата

И.А. Пронин  
инициалы, фамилия

Работа защищена с оценкой отлично (протокол ГЭК от 18.06.26 № 1)

Секретарь ГЭК

  
подпись, дата

А.А. Вольсков  
инициалы, фамилия

## РЕФЕРАТ

Магистерская диссертация на тему "Элементы интегральных видеопроцессоров средств технического зрения для автоматизации механических угломеров " состоит из пяти глав пояснительной записки и восьми приложений. Пояснительная записка содержит 80 листов, 30 рисунок, 48 формулы, 11 таблицы, 12 использованных источников.

Ключевые слова: механический угломер, видеосигнал, специализированный видеопроцессор, ПЛИС, обнаружение делений лимба, система технического зрения, автоматизация считывания, пиксель, двоичный код, регистр, элементы методики цифровой обработки видеосигнала.

Цель проведенной работы: совершенствование программно-технических средств автоматизации измерения углов.

В современном мире, несмотря на активное внедрение цифровых технологий, в различных отраслях человеческой деятельности (строительство, машиностроение, геодезия и др.) по-прежнему широко распространены механические угломеры. Данные приборы остаются востребованными благодаря своей автономности, надежности и низкой стоимости. Однако существует объективная задача автоматизации технологических процессов, требующая оперативного ввода данных измерений в программно-технические комплексы.

В настоящее время человек является связующим звеном между механическим лимбом и цифровой системой: визуально считывает показания и вручную вводит их в базу данных или систему управления. Такое участие оператора вносит риск субъективных ошибок, как на этапе считывания, так и на этапе ввода информации, что снижает достоверность и скорость измерений. Поэтому совершенствование технических средств автоматизации процесса передачи в программно-технические комплексы угловых значений с лимба механического угломера является актуальным. Решение данной задачи позволит не только повысить точность и достоверность угловых

измерений, но и ускорить исполнение технологических процессов, а также обеспечить возможность проведения измерений в условиях, потенциально опасных для человека.

Предмет исследования: методики, алгоритмы и технические средства телевизионного считывания показаний механических угломеров с последующей цифровой обработкой и передачей данных.

Методы исследования. В работе используются методы цифровой обработки изображений, компьютерного зрения, методики схемотехнического проектирования, математического моделирования, а также язык описания аппаратуры для моделирования поведенческих моделей.

Во введении обоснована актуальность темы, сформулированы основная цель и задачи работы.

Первая глава посвящена механическим и оптико-электронным средствам измерения углов, а также рассматриваются некоторые современные методики распознавания объектов на видеоизображении, применимые для задачи распознавания перемещения делений лимбов.

Вот второй главе предлагаются элементы методики обработки видеоизображения для получения значения угла поворота в цифровом виде, включая этапы фильтрации, бинаризации и идентификации делений лимба.

В третьей главе описан, разработанный автором, специализированный видеопроцессор и его основные цифровые узлы, реализующие предложенную методику обработки видеоизображения для получения значения угла поворота в цифровом виде в реальном масштабе времени.

В четвертой главе описывается процесс создания поведенческих моделей ключевых функциональных узлов видеопроцессора на языке *Verilog*.

Научная и практическая значимость. Решением поставленной проблемы является разработка методики обработки видеосигнала, а также проектирование специализированного устройства на базе ПЛИС – видеопроцессора, реализующего телевизионный способ считывания

показаний шкалы угломера. Разработанный видеопроцессор позволяет реализовать в телевизионных технических средствах функцию считывания показаний механических угломеров с последующей передачей данных в программно-технические комплексы.

Полученные результаты внедрены в технологические процессы и в опытно-конструкторские работы предприятия «АО «НПП Рубин».

## СОДЕРЖАНИЕ

|                                                                                                                                      |    |
|--------------------------------------------------------------------------------------------------------------------------------------|----|
| Введение                                                                                                                             | 7  |
| 1.Методы обработки изображений для автоматизации считывания показаний шкал угломеров                                                 | 9  |
| 1.1 Современные механические и оптико-электронные средства измерения углов                                                           | 9  |
| 1.2 Методы обработки изображений для распознавания объектов                                                                          | 13 |
| 2 Разработка методики цифровой обработки видеосигнала для распознавания перемещения и считывания показаний шкал                      | 26 |
| 2.1 Элементы методики цифровой обработки видеосигнала для распознавания перемещения и считывания показаний шкал                      | 26 |
| 2.2 Нормализация и фильтрация видеоизображения шкал угломера                                                                         | 27 |
| 2.3 Детектирование границ и определение положения на видеоизображении делений шкал угломера                                          | 36 |
| 2.4 Вычисление значений углов по положению на видеоизображении делений шкал угломера                                                 | 37 |
| 3 Разработка структурной схемы видеопроцессора средств технического зрения для распознавания перемещения и считывания показаний шкал | 41 |
| 3.1 Структура телевизионной приставки для механического угломера                                                                     | 41 |
| 3.2 Структура видеопроцессора средств технического зрения для распознавания перемещения и считывания показаний шкал                  | 43 |
| 4 Разработка поведенческих моделей и синтез электрических схем функциональных узлов видеопроцессора                                  | 46 |
| 4.1 Усредняющий фильтр видеоизображения                                                                                              | 46 |
| 4.2 Медианный фильтр видеоизображения                                                                                                | 50 |
| 4.3 Бинаризатор видеоизображения                                                                                                     | 54 |
| 4.4 Детектор границ делений шкалы                                                                                                    | 59 |
| 4.5 Вычислитель значений углов                                                                                                       | 64 |

|                                                                                                                                                 |     |
|-------------------------------------------------------------------------------------------------------------------------------------------------|-----|
| 5 Анализ влияния освещенности научно-исследовательской лаборатории на работу инженера при изготовлении и эксплуатации проектируемого устройства | 69  |
| Заключение                                                                                                                                      | 76  |
| Список использованных источников                                                                                                                | 77  |
| Приложение А. Задание на магистерскую диссертацию                                                                                               | 80  |
| Приложение Б. Форма отчета о поиске патентов                                                                                                    | 84  |
| Приложение В. Опубликованные статьи за 2024/2026 учебные года                                                                                   | 86  |
| Приложение Г. Заявление и протокол о проверке на антиплагиат                                                                                    | 88  |
| Приложение Е. Структурная схема телевизионной приставки для механического угломера                                                              | 92  |
| Приложение Ж. Поведенческие модели функциональных узлов видеопроцессора на языке <i>Verilog</i>                                                 | 94  |
| Приложение З. Электрические схемы функциональных узлов видеопроцессора, синтезированные по поведенческим моделям                                | 109 |
| Приложение И. Материалы, иллюстрирующие результаты магистерской диссертации                                                                     | 116 |

## ВВЕДЕНИЕ

Современный этап развития измерительной техники и автоматизации технологических процессов характеризуется активным внедрением цифровых технологий во все сферы производства. Первостепенной задачей при построении эффективных автоматизированных систем управления является обеспечение достоверного и оперативного сбора данных для учета, анализа и выработки управляющих воздействий. В связи с этим актуальность приобретает вопрос интеграции традиционных механических средств измерений, в частности механических угломеров, в единое цифровое пространство автоматизированных систем контроля и управления.

На современном промышленном рынке и в арсенале предприятий существует значительное количество механических угломеров с отсчетом по лимбу. Данные приборы, работающие на основе ручного управления, остаются важными и незаменимыми в различных профессиональных сферах – от строительства и слесарного производства до изготовления изделий специального назначения и астрономии, где точные измерения углов критически важны для определения положения объектов в пространстве. В то же время, современные механические угломеры комплектуются встроенными цифровыми средствами измерения (инкрементальными энкодерами, инерциальными датчиками), что позволяет легко интегрировать их в автоматизированные системы. Механические угломеры с лимбом требуют визуального считывания показаний оператором. Это не только подвержено влиянию субъективных ошибок, но и ограничивает скорость и объективность измерений, связанных с человеческим фактором.

Показания, снятые с механического угломера, вводятся оператором вручную даже при использовании автоматизированной системы.

Так же в ряде практических задач возникает необходимость проведения бесконтактных операций в условиях, представляющих опасность для человека (агрессивные среды, высокие температуры, радиация и т.д.), где присутствие оператора невозможно или нежелательно.

Таким образом, разработка систем автоматизированного считывания показаний лимбов и передачи угловых значений в цифровом виде в систему управления является актуальной научно-технической задачей. В связи с этим возможно усовершенствование подобных систем путем проектирования прибора с телевизионным способом считывания показаний шкалы угломера с последующей интеграцией в компьютерные системы для анализа и хранения данных, что позволит обеспечить высокую точность и оперативность выполняемых операций и минимизировать риски для человеческого здоровья.

Цель проведенной работы: совершенствование программно-технических средств автоматизации измерения углов.

Для достижения поставленной цели были поставлены и решены задачи:

1. Выполнить анализ:
  - современных механически и оптико-электронных средств измерения углов;
  - методов обработки изображений для распознавания объектов;
2. Разработать элементы методики цифровой обработки видеосигнала для распознавания перемещения и считывания показаний шкал;
3. Разработать структурную схему видеопроцессора средств технического зрения для распознавания перемещения и считывания показаний шкал;
4. Разработать поведенческие модели функциональных узлов видеопроцессора на языке *Verilog*, синтезировать с помощью современных программных средств их электрические схемы.

## 1 Методы обработки изображений для автоматизации считывания показаний шкал угломеров

### 1.1 Современные механические и оптико-электронные средства измерения углов

Угломеры, как традиционные инструменты, так и современные электронные устройства, играют ключевую роль в обеспечении точности и надежности измерений. По своей конструкции, назначению и способу измерения угломеры делятся на четыре основных вида:

1. Цифровые угломеры (рисунок 1 - а). Вычислительной основой данных угломеров служат полупроводниковые компоненты. Их ключевой особенностью является интегрированный датчик (как правило, акселерометр или емкостный сенсор), измеряющий угол наклона или поворота относительно выбранной базы (чаще всего горизонтальной или вертикальной оси). Результаты в цифровом виде выводятся на дисплей, что минимизирует субъективные ошибки считывания, характерные для аналоговых приборов. Наиболее точные устройства данного класса – электронные инклинометры находят применение в машиностроении, строительстве и геодезии. Ограничением в использовании является зависимость от источника электропитания, что может усложнить эксплуатацию в полевых условиях.

2. Лазерные угломеры (рисунок 1 - б). Принцип их работы основан на проецировании двух лазерных лучей на контролируемые точки, между которыми необходимо определить угол. Измерение может производиться визуально по шкале или электронным способом с помощью встроенного процессора. Существенным недостатком является снижение видимости луча при высокой окружающей освещенности, а также зависимость контрастности от отражающих свойств поверхности. Для компенсации этого фактора используются специализированные приемники-мишени, которые обеспечивают точное определение луча и могут подавать световой или звуковой сигнал о правильном наведении.

3. Оптико-механические угломеры (рисунок 1 - в,г). К данной категории относятся приборы, использующие оптическую систему для визирования и механическую шкалу для отсчета. Типичным и наиболее функциональным представителем является тахеометр, объединяющий угломеры и дальномерные функции. Отдельно можно выделить маятниковые (гироскопические) угломеры, принцип действия которых основан на свойстве маятника или гироскопа сохранять положение относительно вектора гравитации или гироскопа сохранять положение относительно вектора гравитации или инерциального пространства, что позволяет производить измерения независимо от магнитных полей.

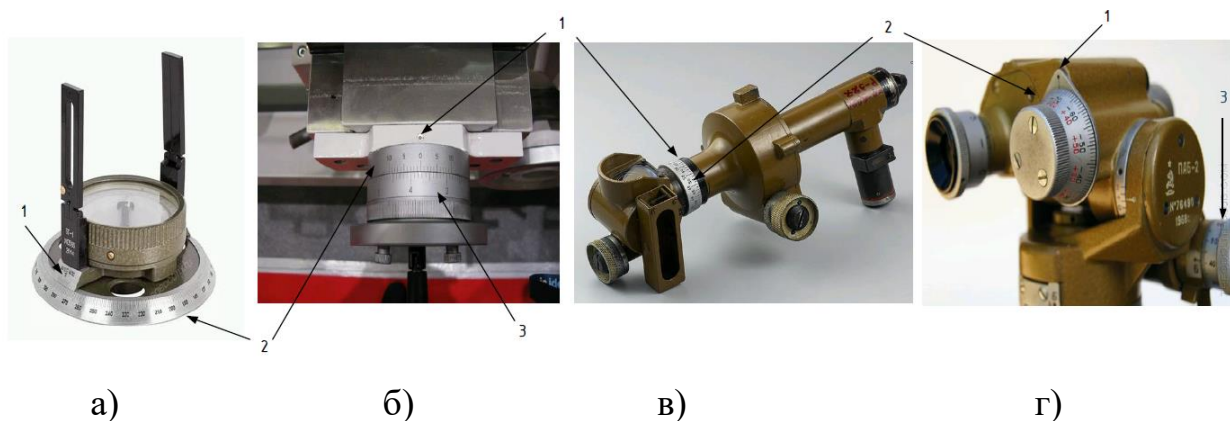
4. Механические угломеры (рисунок 1 - д). Конструктивно простые и надежные инструменты, не требующие источников питания. Подразделяются на два основных типа: простые угловые шаблоны и угломеры с отсчетной шкалой (нониусные или циферблатные). К классическим разновидностям относятся транспортиры различных конструкций и поворотные угломеры с фиксирующим механизмом. Их преимущества - высокая надежность, долговечность и независимость от условий окружающей среды [1].



а) электронный угломер; б) лазерный угломер; в) оптический угломер;  
г) гироскоп; д) механический угломер

Рисунок 1 – Виды угломеров

Особую важность механические угломеры (МУ) имеют в различных аспектах военной техники. Данный тип угломеров используется в навигационных системах для корректировки курса движения и в артиллерийской технике для установки углов наклона орудия и корректировки траектории. На рисунке 2 приведены примеры изделий, в составе которых присутствуют МУ.



а) строительная буссоль ИЗЮМ БГ-1;

б) нониусная шкала на стенке;

в) перископический панорамный прицел (Панорама Герца);

г) артиллерийская буссоль ПАБ-2М

1 - опорная риска;

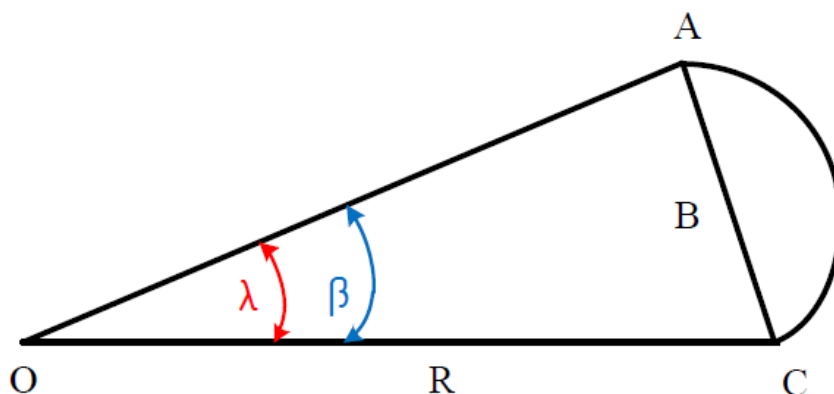
2 - лимб малых делений угломера;

3 - лимб больших делений угломера

Рисунок 2 – Механические угломеры в составе изделий

Большинство углоизмерительных приборов оснащены двумя шкалами - шкалой «грубого отсчета» или кольцо лимба, предназначенной для установки количества больших делений угломера (БДУ), и шкалой «точного отсчета» или барабан панорамы, используемой для определения количества малых делений угломера (МДУ) измеряемых углов. Деление угломера принятой в артиллерии основано на базовой единице измерения под названием тысячная, которая равна  $1/6000$  части оборота.

Геометрический принцип функционирования МУ, показанный на рисунке 3, основан на сравнении величины измеряемого угла с соответствующей частью длины окружности. Дуга  $AC = 6,28R/6000 \approx 1/955R$ . В расчетах принимается допущение, согласно которому длина дуги  $AC$  приравнивается к длине стягивающей ее хорды  $AC$ , если они составляют  $1/1000$  радиуса  $R$  или дальности  $OC$ . Корректность такого приближения обусловлено существенным превышением дальности  $OC$  над габаритами объекта контроля.



$\lambda$  - малое деление угломера;  $\beta$  - тысячная

Рисунок 3 – Деление угломера и тысячная

Таким образом, между МДУ и БДУ формируется смешанная шестидесятерично-десятичную система измерения углов: если одно малое деление угломера в градусной системе соответствует  $3,6'$ , то одно большое деление угломера равняется  $6^\circ$  [2,3].

## 1.2 Методы обработки изображений для распознавания объектов

Для решения задачи выделения границ объектов на изображениях существует множество алгоритмов, выбор которых определяется спецификой конкретной задачи. Теоретическая основа большинства методов основана на том, что граница объекта соответствует зоне значительного перепада яркости между относительно однотонными областями. Однако на практике подобные перепады могут быть обусловлены не только геометрическими контурами объектов, но и факторами, затрудняющими детектирование, такими как: текстурные особенности поверхности, неравномерность освещения, наличие теней или бликов.

Для повышения надежности выделения контуров исходное изображение подвергается предварительной обработке (линейной или нелинейной).

Классическим методом линейной обработки изображения с целью выделения полезной информации на фоне шума является преобразование Фурье. Дискретное двумерное преобразование Фурье (ДПФ) позволяет перейти от пространственного представления изображения, заданного функцией яркости  $G(j,k)$ , к его частотному спектру:

$$\Gamma(l, m) = \frac{1}{n_x n_y} \sum_{l=0}^{n_x-1} \sum_{m=0}^{n_y-1} G(j, k) \exp \left( -2\pi i \left( j \frac{l}{n_x} + k \frac{m}{n_y} \right) \right), \quad (1)$$

где  $l, m$ -преобразования или частотные параметры;  $n_x, n_y$  – число элементов;  $i$  – мнимая единица.

Обратное преобразование Фурье восстанавливает исходное изображение:

$$G(j, k) = \sum_{l=0}^{n_x-1} \sum_{m=0}^{n_y-1} \Gamma(l, m) \exp \left( 2\pi i \left( j \frac{l}{n_x} + k \frac{m}{n_y} \right) \right), \quad (2)$$

где  $j, k$  – изображения или пространственные переменные.

В случае периодического шума его компоненты проявляются в спектре  $\Gamma(l, m)$  в виде изолированных пиков. Фильтрация в этом случае заключается в

подавлении данных пиков путем замены их на значения средней локальной амплитуды, что приводит к сглаживанию изображения.

Однако прямое вычисление ДПФ связано со значительными вычислительными затратами, что стимулировало разработку упрощенного преобразования. Одним из таких преобразований является преобразование Уолша-Адамара при  $n_x = n_y = n$  имеет спектр:

$$\Gamma(l, m) = \frac{1}{n} \sum_{l=0}^{n-1} \sum_{m=0}^{n-1} G(j, k) (-1)^{\alpha_{jh}(l, m)}, \quad (3)$$

где  $\alpha_{jh}(l, m) = \sum_{v=0}^{n-1} (\beta_v(j)\beta_v(l) + \beta_v(k)\beta_v(m))$ , а коэффициенты  $\beta_v(q)$  принимают значение 0 или 1 в соответствии со значением  $v$ -го разряда двоичного представления числа  $q$  [4,5].

Более простой метод линейной обработки шума основан на пространственном сглаживании путем усреднения значений яркости. Одним из распространенных подходов – скользящее среднее в заданном «окне» размером  $(2L+1)(2M+1)$ , новое значение яркости пикселя:

$$\overline{G}(i, j) = \frac{1}{n} \sum_{m=-M}^M \sum_{l=-L}^L G(i + l, j + m), \quad (4)$$

где  $\overline{G}(i, j)$ - результирующее значение яркости;  $(i, j)$ -координаты текущего обработанного пикселя;  $G(i + l, j + m)$ -исходное значение яркости пикселя с координатами  $(i + l, j + m)$  до обработки;  $M$  и  $L$ - параметры окна по горизонтали и вертикали (количество столбцов и строк);  $i$  и  $m$  – индексы смещения внутри окна по вертикали и горизонтали.

На рисунке 4 представлен принцип работы усредняющего фильтра.

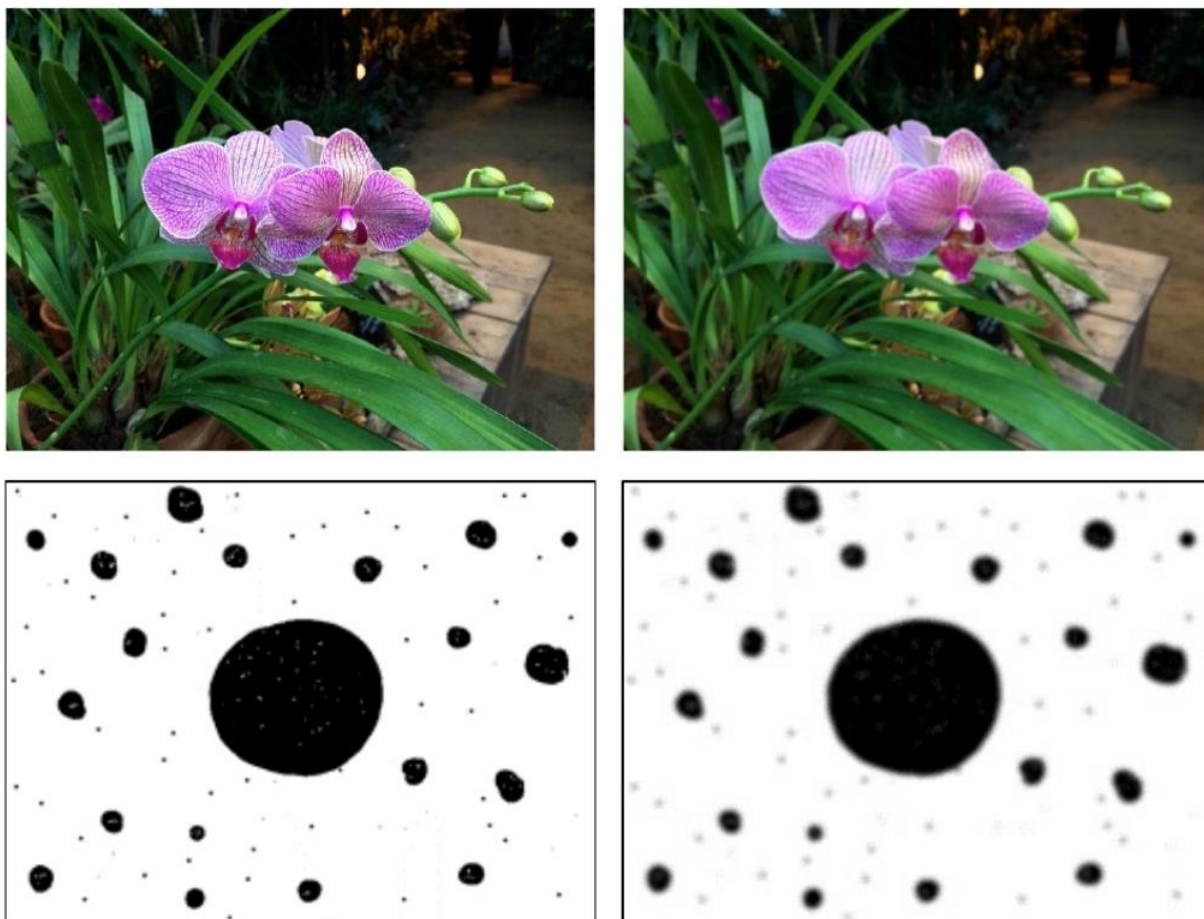


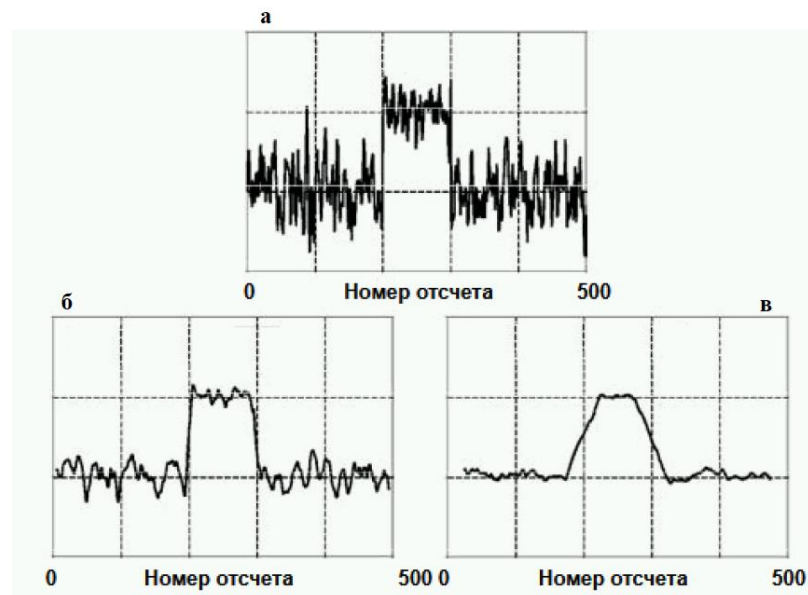
Рисунок 4 – Результат применения усредняющего фильтра

Несмотря на эффективность в подавлении высокочастотного шума, методы на основе простого усреднения обладают недостатком: они устраняют не только шум, но и мелкие детали изображения, приводя к размытию резких границ и эффекту, аналогичному расфокусировке [6,7].

Один из методов нелинейной обработки сигналов, направленный на подавление адаптивного шума является медианная фильтрация. Этот метод особенно эффективен при наличии импульсного шума, который представляет собой ограниченный набор пиковых значений на фоне нулей. Медианный фильтр реализуется как процедура локальной обработки с использованием скользящего окна различной геометрической формы, включающего нечетное число отсчетов изображения. Процесс обработки заключается в том, что для каждого положения окна отсчеты, попадающие в его область, упорядочиваются по возрастанию или убыванию значений. Центральный

отсчет в этом упорядоченном списке называется медианой рассматриваемой группы из  $N$  отсчетов. Для данной медианы существует  $(N-1)/2$  отсчетов, которые меньше или равны ей по величине, и столько же отсчетов, превышающих ее. Эта медиана занимает центральное положение в окне для обработанного сигнала. В результате применения медианного фильтра наклонные участки и резкие перепады значений яркости на изображениях остаются неизменными, тогда как импульсные помехи подавляются при условии, что их длительность составляет менее половины окна. Увеличение размера окна приводит к стиранию более крупных деталей изображения. Существуют модификации базового алгоритма (одномерные и двумерные), направленные на улучшение его избирательности. К ним относится, например взвешенный медианный фильтр, в котором при формировании упорядоченной выборки каждое значение из окна учитывается с заданным целочисленным весом, что позволяет управлять степенью влияния пикселей из разных областей окна в результате [8,9].

На рисунке 5 и 6 представлен принцип работы медианного фильтра на примере сигнала и изображения.



а - исходное изображение; б - медианная фильтрация по 11 элементам;

в - медианная фильтрация по 51 элементу

Рисунок 5 – Принцип работы медианного фильтра

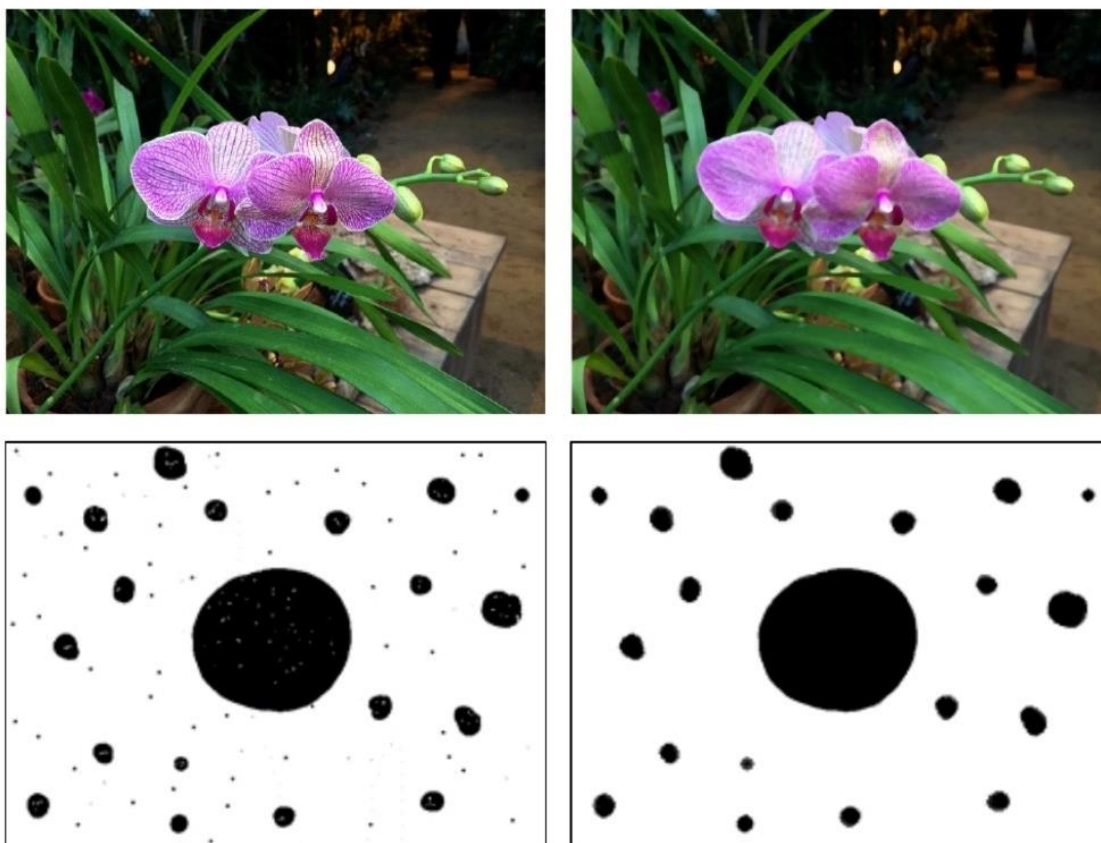


Рисунок 6 – Результат применения медианного фильтра

Линейная обработка изображения – процесс бинаризации. Существуют различные методы бинаризации, которые классифицируются на две основные группы: глобальные (пороговые) и локальные (адаптивные).

Глобальные методы бинаризации обрабатывают изображение целиком. В ходе работы находится порог бинаризации  $t$ , с помощью которого происходит деление на черное и белое, причем величина порога остается неизменной в течение всего процесса. К пороговым методам относят:

- бинаризация с нижним порогом;
- бинаризация с верхним порогом;
- бинаризация с двойным ограничением;
- неполная пороговая обработка;
- многоуровневое пороговое преобразование.

Одним из наиболее известных методов глобальной бинаризации является метод Оцу, который вычисляется порог  $t$ , минимизирующий среднюю ошибку сегментации, т.е. среднюю ошибку в определении принадлежности пикселей к объекту или фону. Яркости пикселей изображения можно рассматривать как случайные величины, а их гистограмму – как оценку плотности распределения вероятностей. При известной плотности распределения вероятностей можно определить оптимальный порог для сегментации изображения на два класса: 0 и 1 (объект и фон соответственно). Наиболее простой подход к определению порога в рамках глобального метода бинаризации заключается в нахождении минимального  $g_{min}$  и максимального  $g_{max}$  значения яркости изображения и нахождении среднего значения между ними:

$$t = (g_{max} - g_{min}) / 2 \quad (5)$$

Локальные (адаптивные) методы бинаризации предлагают разбиение изображения на области, для каждой из которых вычисляется порог, основываясь на информации об интенсивности пикселей. Алгоритмы данного класса делят изображение на блоки определенного размера, причем размер блока должен быть минимальным, но достаточным для сохранения исходных особенностей и деталей изображения. В то же время блоки должны быть настолько большими, чтобы влияние шумов на результат было минимально. Использование функции сглаживания результирующего растра при адаптивной бинаризации позволяет достичь удовлетворительных результатов без применения дополнительных фильтров. Наиболее распространенными адаптивными алгоритмами являются: метод Ниблэка, метод Бернсена, метод Эйквеля, метод Саувола и метод Кристиана.

В методе Ниблэка для каждого пикселя необходимо получить свое значение порога. Его величина определяется на основе вычисления локально среднего и локального среднеквадратичного отклонения. Значение порога для точки с координатами  $(m, n)$  считается по формуле 6:

$$t(m,n)=\mu(m,n)+k*\sigma(m,n), \quad (6)$$

где  $\mu(m,n)$ -локально среднее;  $\sigma(m,n)$ - среднеквадратичного отклонение в локальной окрестности рассматриваемой точки изображения;  $k$  - часть границы изображения, которая берется в качестве объекта[10,11].

Метод Бернсена предполагает деление всего изображения на квадраты размером  $gxg$ , имеющие центр в точке  $(m, n)$ . Каждый пиксель имеет порог в пределах квадрата, вычисляющийся по формуле 7:

$$t(m,n)=(J_{high} + J_{low})/2, \quad (7)$$

где  $J_{high}$  и  $J_{low}$  – наибольший и наименьший уровень яркости квадрата. Если текущий пиксель  $t(m,n)$  меньше  $\xi$ -константа, изначально заданная пользователем, то пиксель будет относиться только к черному или белому [12].

В методе Эйквеля изображение обрабатывается при помощи двух концентрических окон ( $S$  и  $L$  –маленькое и большое), которые имеют квадратную форму. Данные окна последовательно накладываются на изображение слева направо, сверху вниз, с шагом в длину стороны маленького окна. Для большого окна рассчитывается порог  $B$ , чтобы можно было поделить на два кластера. Если математическое ожидание уровня яркости в двух кластерах имеют разницу, превышающую некоторый заданный уровень, то все пиксели внутри окна  $S$  бинаризируются в соответствии с порогом  $t$ .

Определение локального порога бинаризации в методе Саувола осуществляется с помощью прохождения всего изображения окном  $w \times w$ . В данном методе порог  $t(x,y)$  определяется формулой 8:

$$t(x,y)=m(x,y)(1 + k \left( \frac{s(x,y)}{R} - 1 \right)), \quad (8)$$

где  $m(x,y)$ -среднее значение,  $s(x,y)$ - среднеквадратичное отклонение интенсивности пикселя,  $R$ -максимально отклонение,  $k$ -параметр, который принимает значение в диапазоне (0.2, 0.5) [12].

На рисунке 7 приведены примеры влияния различных методов бинаризации на изображение.

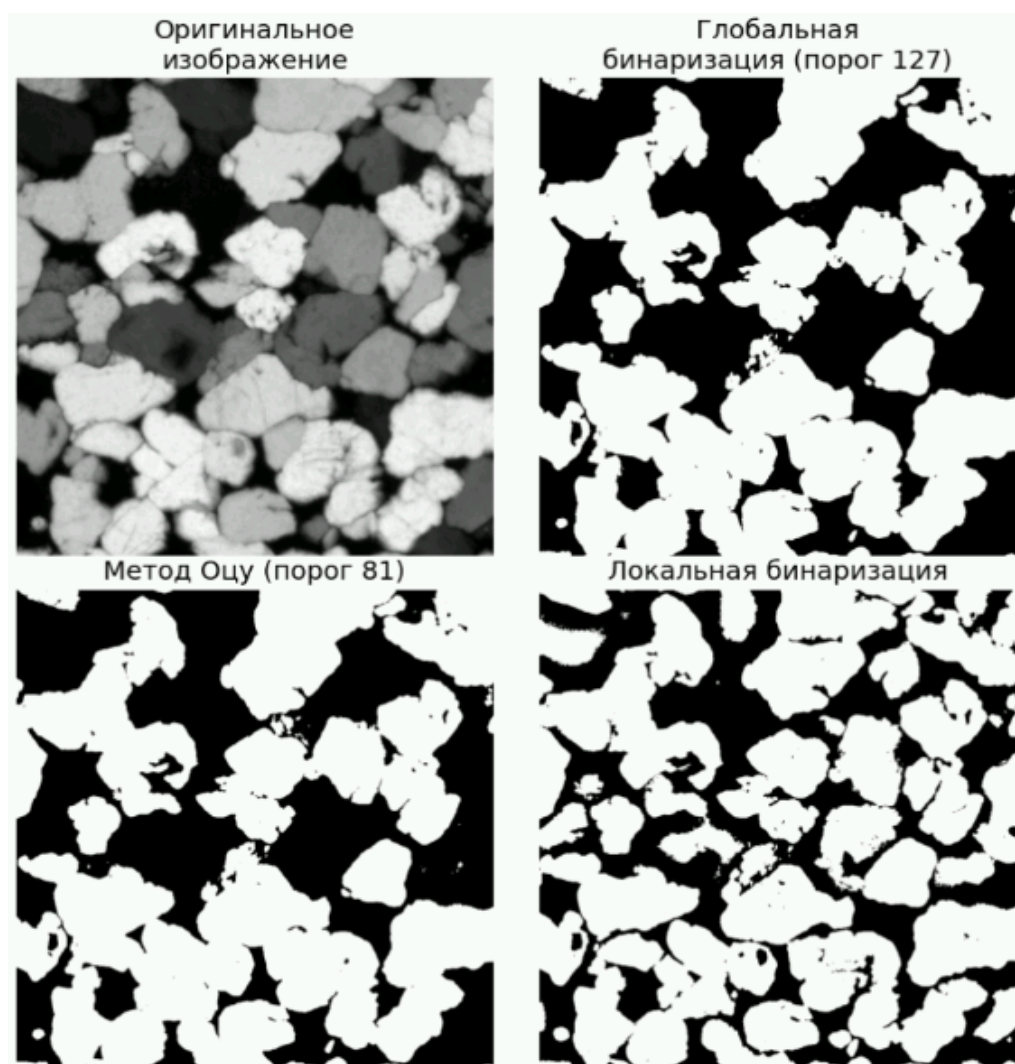


Рисунок 7 – Примеры применения методов бинаризации

После выполнения нормализации изображения осуществляется переход к следующему этапу – определению границ изображения. Среди классических методов детектирования границ на видеоизображения широко применяются операторы Собеля и Лапласа, детектор границ Кэнни и преобразование Хафа.

Оператор Собеля реализует аппроксимацию градиента яркости изображения посредством свертки с небольшими сепарабельными целочисленными фильтрами в ортогональных направлениях. В результате

для каждой точки вычисляется двумерный вектор градиента, характеризующий направление и скорость наибольшего изменения яркости. Данная информация позволяет оценить степень резкости перепада, вероятность принадлежности пикселя градиента объекта, а также ее пространственную ориентацию. Физически, результирующий вектор в точке с постоянной яркостью будет нулевым, тогда как в зоне перепада яркости вектор укажет направление ее возрастания. Математически оператор Собеля реализуется путем наложения на изображение двух ортогональных масок свертки размерностью 3x3:

$$G_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} * A, \quad (9)$$

$$G_y = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} * A, \quad (10)$$

где  $A$  – исходное изображение; «\*» - обозначение двумерной операции свертки;  $G_x$  и  $G_y$  – изображения, на которых каждая точка содержит приближенные производные по  $x$  и по  $y$ .

Маски позволяют выявить границы, ориентированные как вертикально, так и горизонтально на изображении. Конечное значение градиента определяется по формуле 11:

$$G(x, y) = \sqrt{G_x^2(x, y) + G_y^2(x, y)} \quad (11)$$

В отличие от оператора Собеля, аппроксимирующего первую производную, оператор Лапласа вычисляет лапласиан изображения – скалярную величину, равную сумме производных второго порядка. На практике это реализуется с помощью линейной фильтрации, а именно это достигается путем умножения каждого элемента двумерной апертуры 3x3 на соответствующий элемент матрицы Лапласа [12].

Более комплексным подходом является детектор границ Кэнни, который представляет собой фильтр, состоящий из суммы четырех экспонент, и реализуется многоступенчатым алгоритмом для обнаружения широкого спектра границ в изображении. Алгоритм включает в себя несколько этапов, первый из которых включает в себя процесс размытия изображения с целью устранения шума и лишних деталей, который называется сглаживанием. Оператор Кэнни применяет фильтр, приближающийся к первой производной гауссианы ( $\sigma=1,4$ ).

На втором этапе алгоритма Кэнни осуществляется вычисление градиента яркости изображения. Границы объектов соответствуют локальным максимумам градиента. Поскольку границы могут иметь разные направления, то алгоритм Кэнни использует четыре фильтра для обнаружения горизонтальных, вертикальных и диагональных ребер в размытом изображении. На этом этапе используется вышеописанный оператор Собеля для получения первой производной по горизонтали ( $G_x$ ) и вертикали ( $G_y$ ). Используя формулу 12 для вычисления градиента, можно определить угол направления границы:

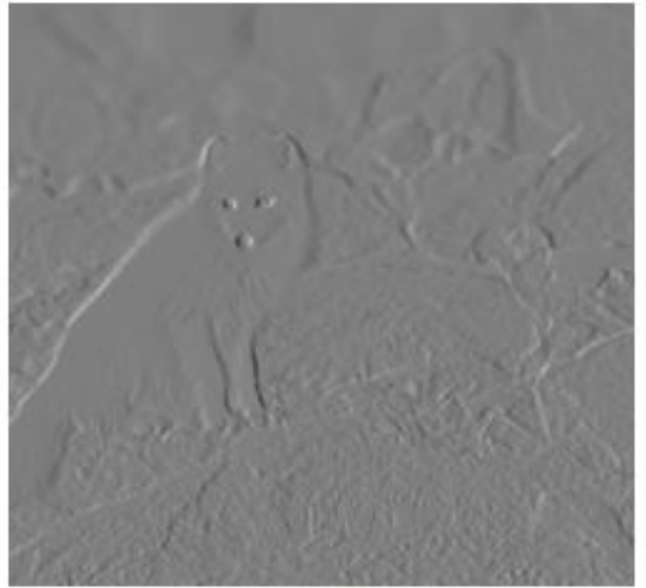
$$Q = \arctan(G_y/G_x) \quad (12)$$

Полученный угол округляется до одного из четырех возможных значений:  $0^\circ$ ,  $45^\circ$ ,  $90^\circ$  и  $135^\circ$ . Это необходимо для последующего подавления немаксимумов, то есть только локальные максимумы отмечаются как границы. Также выполняется двойная фильтрация, позволяющая идентифицировать потенциальные границы, и подавление всех краев, не относящихся к установленному контуру [13].

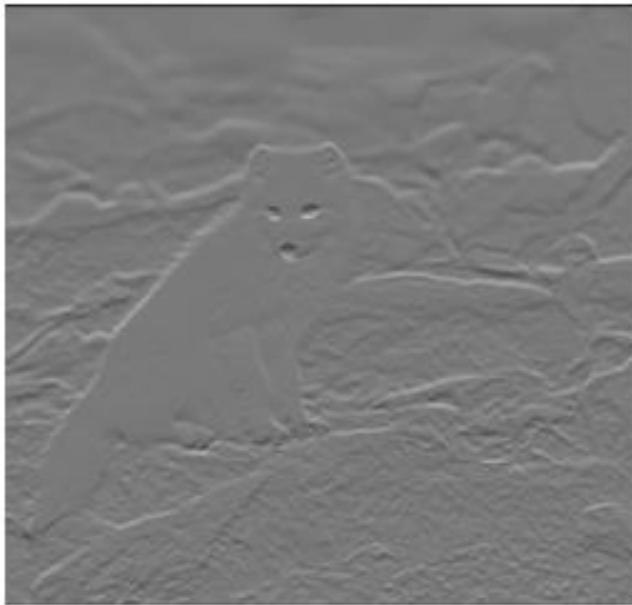
На рисунках 8 и 9 представлено применение оператора Собеля с бинаризацией и без, а также на рисунках 10 и 11 - детектор границ Кэнни.



а)



б)



в)



г)

а) исходное изображение; б)  $G_x$ ; в)  $G_y$ ; г) градиентная величина

Рисунок 8 – Результат применения оператор Собеля

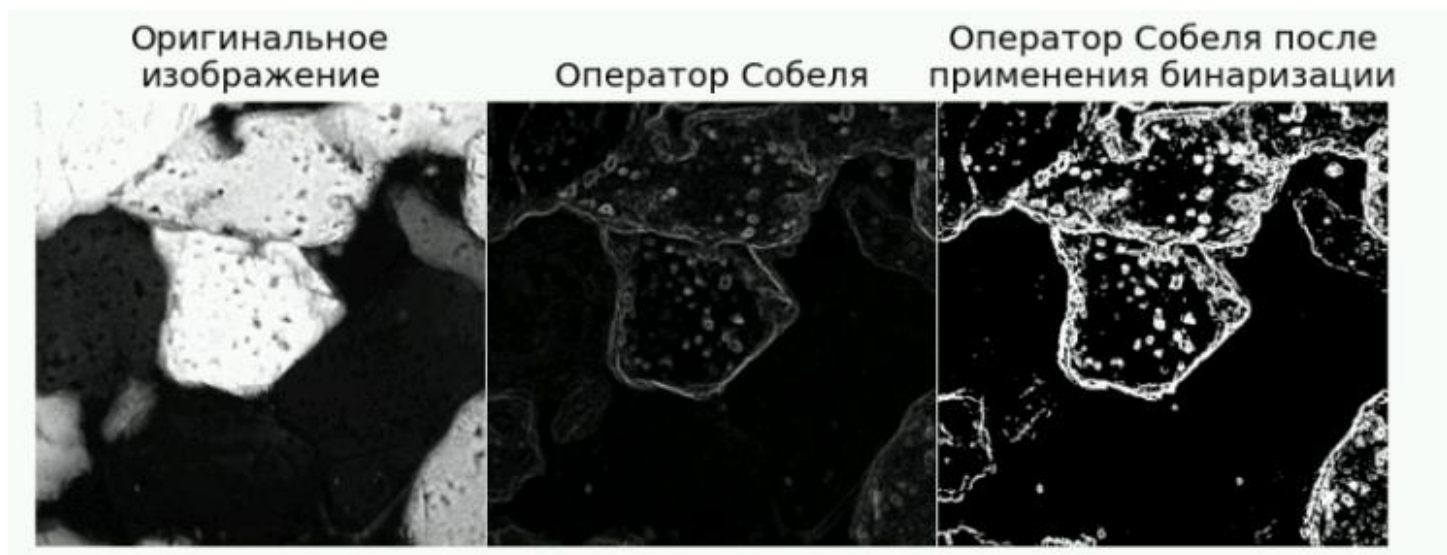


Рисунок 9 – Результат применения оператора Собеля



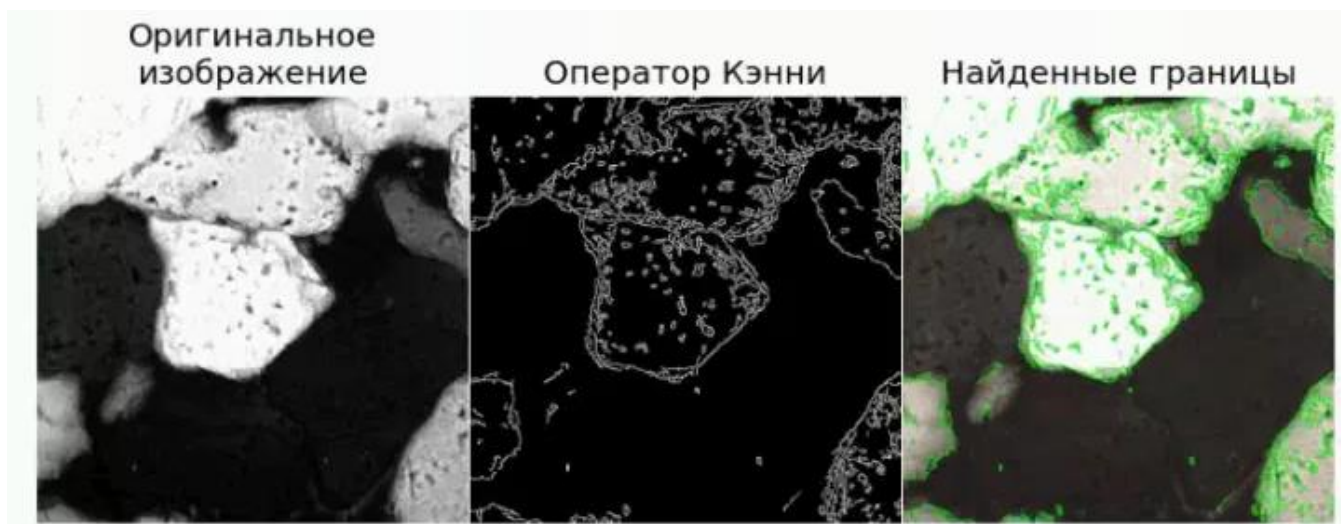
а)



б)

а) исходное изображение; б) границы Кэнни

Рисунок 10 – Результат применения детектора границ Кэнни



Обнаруженные контуры изображения подсвечены зеленым

Рисунок 11 – Результат применения детектора границ Кэнни

Для поиска линий и других простых форм на изображении используется преобразование Хафа. Этот метод служит для выявления объектов, принадлежащих определенному классу фигур (например, прямым или окружностям), с помощью процедур голосования. В процессе поиска линий основополагающим является утверждение о том, что абсолютно любая точка изображения может являться частью некоторого набора линий.

Алгоритм использует представление линий в полярной системе координат:

$$R = x * \cos(f) + y * \sin(f), \quad (13)$$

где  $x$  и  $y$  – координаты;  $f$  – угол между перпендикуляром к прямой и осью  $OX$ ;  $R$  – длина нормали к прямой из начала координат.

Через каждую точку с координатами на изображении можно провести несколько прямых с разными значениями  $R$  и  $f$ , таким образом, каждой точке  $(x, y)$  изображения соответствует набор точек  $(R, f)$  в пространстве, образующий синусоиду [14,15].

## 2 Разработка методики цифровой обработки видеосигнала для распознавания перемещения и считывания показаний шкал

### 2.1 Элементы методики цифровой обработки видеосигнала для распознавания перемещения и считывания показаний шкал

Целью обработки видеопотока является автоматическое детектирование факта перемещения объекта контроля и точное считывание текущих показаний со шкалы в условиях, близких к реальному времени. Основные этапы методики должны быть устойчивы к изменяющимся условиям освещенности, вибрациям и обеспечивать работу в режиме, близком к реальному времени.

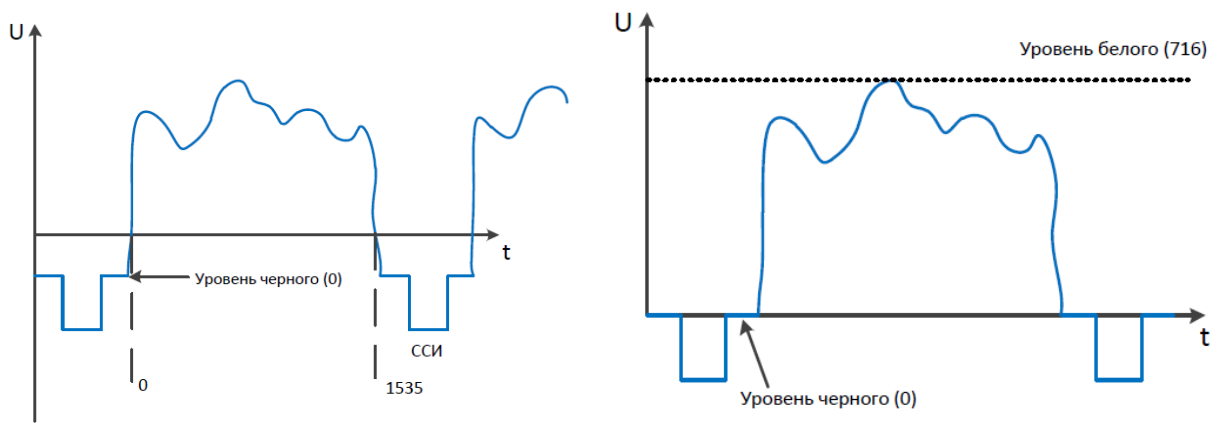
Алгоритм предлагаемой методики представляет обобщенную структуру для цифровой обработки видеосигнала, которая допускает адаптацию под конкретные требования и включает следующие этапы:

- 1) нормализация видеосигнала для получения значений яркости элементов изображения в диапазоне, необходимом для дальнейших вычислений;
- 2) фильтрация видеосигнала для ослабления шумов на изображении и его бинаризации;
- 3) детектирование границ изображения делений шкал механического угломера и фиксация координат их положения на изображении;
- 4) вычисление угла поворота лимба по линейным координатам положения делений;
- 5) дополнение изображения значениями вычисленных углов, передача их через цифровой интерфейс в персональный компьютер.

## 2.2 Нормализация и фильтрация видеоизображения шкал угломера

Аналоговый видеосигнал представляет собой двухмерный массив данных, кодирующий параметры яркости и цветности, которые обновляются с заданной частотой кадров. Структура видеосигнала включает активную часть, содержащую полезную информацию, и интервалы гашения. Интервал строчного гашения соответствует периоду возврата луча из правой части экрана в левую для начала новой строки. Интервал кадрового гашения – это возврат луча для начала нового кадра. Внутри интервалов гашения передаются синхронизирующие сигналы. Наличие этих периодов является необходимым условием корректного воспроизведения изображения, предотвращения визуальных артефактов (полосы или мерцание) и стабильной синхронизации устройств отображения.

Для последующей цифровой обработки видеоданных требуется нормировка. Ключевым этапом является фиксация уровня черного, то есть приведение его опорного напряжения к нулевому значению. Эта операция, иллюстрируется на рисунке 12, устраняет проблему «плавающего» среднего уровня видеосигнала, вызванного изменчивой амплитудой при различной яркости элементов изображения (рисунок 12 – а). Результат коррекции представлен на рисунке 12 – б. Нормированные данные формируют стабильную и предсказуемую основу для последующих этапов обработки.



а) видеосигнал после АЦП: уровень чёрного меняется в зависимости от средней яркости изображения; б) нормированный видеосигнал с уровнем чёрного, зафиксированным в нуле

Рисунок 12 – Форма полного сигнала строки

Процесс фильтрации изображения применяется к активной части строки видеоизображения и реализуется в три стадии:

- усредняющий фильтр;
- медианный фильтр;
- бинаризатор.

Для реализации усредняющего фильтра в программном коде предусмотрено использование четырех регистров, в которые видеоданные передаются последовательно. Выходные сигналы каждого регистра объединяются в комбинационную схему, представляющей собой сумматор яркости четырех соседних пикселей. После выполнения операции суммирования получается усредненная сумма, которая делится на четыре. Деление на четыре достигается путем отбрасывания двух младших разрядов (коэффициент деления числа должен соответствовать степени двойки, то есть  $2^2 = 4$ ), в результате чего формируется новый пиксель.

При последовательной загрузке данных предыдущие значения яркостей пикселей сдвигаются вперед, освобождая место для записи значения яркости нового пикселя из исходной строки изображения, тем самым реализуется движение «окна» вычисления вдоль строки изображения.

Для синхронизации передачи яркостного сигнала и временных синхросигналов используются три дополнительных оконечных регистра *SHV*, которые компенсируют задержку времени прохождения яркостного сигнала через регистры вычислительной схемы.

В результате полученная сумма записывается в выходной регистр, при этом, если сигнал равен «1», передается усредненное значение, а если «0», то сигнал с выхода третьего регистра поступает в мультиплексор, который в свою очередь выбирает один из нескольких входных сигналов, то есть управляет потоками данных.

На рисунке 13 приведена структурная схема усредняющего фильтра.

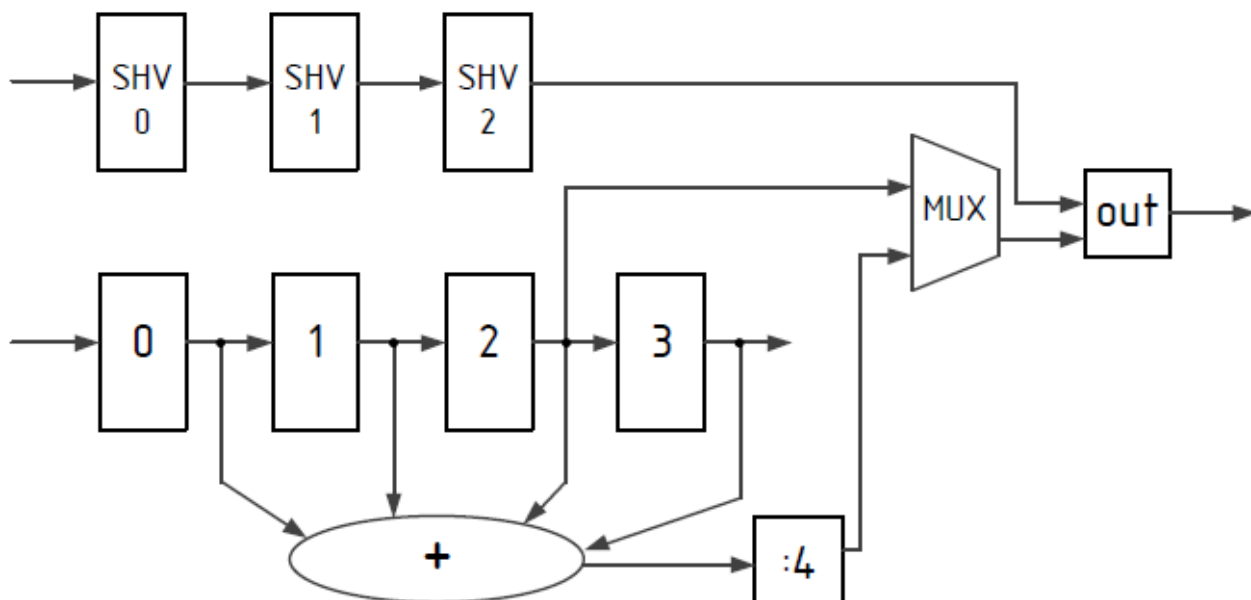


Рисунок 13 - Структурная схема усредняющего фильтра

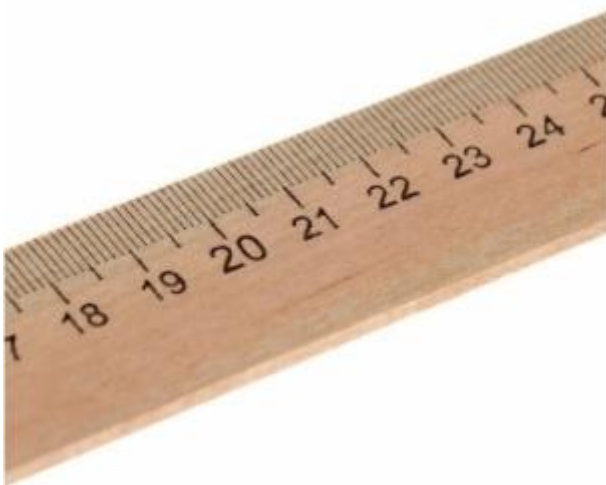
Таким образом, на выходе формируется новое изображение, в котором яркость каждого пикселя представляет собой среднее значение яркости четырех соседних пикселей старого изображения, что показано на рисунке 14.



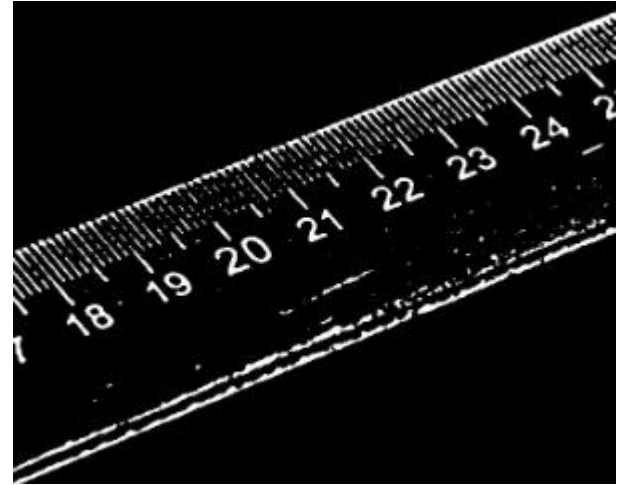
а)



б)



в)



г)

а, в – исходное изображение;

б, г – изображение после применения усредняющего фильтра

Рисунок 14 - Результат применения усредняющего фильтра

Медианный фильтр используется в задачах подавления аддитивного шума, особенно когда шум имеет импульсный характер и состоит из ограниченного набора пиковых значений на фоне нулевых уровней. Данный фильтр выполняет ранжирование значений яркости в заданной области по возрастанию, после чего медиана сформированного ряда значений представляет собой наиболее вероятное значение яркости, соответствующее рассматриваемому пикселю.

Каждый пиксель получает свой вес, и при создании упорядоченного ряда значений учитывается количество раз, которое каждый пиксель «входит» в этот ряд в соответствии с его весом.

В первую очередь медианный фильтр выполняется процедура сравнения яркости пикселей друг с другом с помощью условия: если значение яркости одного пикселя больше другого, то записываем «1», если меньше – «0». Таким образом, результаты сравнения фиксируются в двоичном коде. После происходит суммирование и сортировка по возрастанию значений разрядов двоичного кода. В таблицах 1 и 2 представлены наглядные примеры результатов выполнения данных процедур.

Таблица 1 – Исходная выборка пикселей изображения

| № элемента | 0         | 1         | 2     | 3         | 4         |
|------------|-----------|-----------|-------|-----------|-----------|
| Пиксель    | $X_{n-2}$ | $X_{n-1}$ | $X_n$ | $X_{n+1}$ | $X_{n+2}$ |
| Яркость    | 62        | 37        | 20    | 54        | 83        |

Таблица 2 – Сравнение яркости пикселей

| № элемента | 0 | 1 | 2 | 3 | 4 | $\Sigma$ |
|------------|---|---|---|---|---|----------|
| 0          |   | 1 | 1 | 1 | 0 | 3        |
| 1          | 0 |   | 1 | 0 | 0 | 1        |
| 2          | 0 | 0 |   | 0 | 0 | 0        |
| 3          | 0 | 1 | 1 |   | 0 | 2        |
| 4          | 1 | 1 | 1 | 1 |   | 4        |

Полученная сумма двоичных разрядов результата сравнения указывает на позицию соответствующего элемента в выборке, отсортированной по возрастанию, что позволяет ранжировать пиксели по яркости. Далее выбирается значение яркости, для которого сумма разрядов равна 2. Однако, не всегда сумма разрядов равна 2, поэтому для корректной работы алгоритма фильтрации необходимо ввести дополнительные правила:

- 1) если существует значение суммы равное числу 2, то медианой полученного ряда является ячейка с номером 2;
- 2) двух значений суммы равных 4 не бывает, при этом сумма равная 0 есть всегда;
- 3) если, при сравнении суммы, число 0 встречается более двух раз, то медиана равна пикселей в сумме равной 0;
- 4) если значение суммы равные 0 встретились два раза, то сумма равная 2 есть всегда;
- 5) если сумма равная 1 встретила два раза, то медиана равна яркости первого пикселя из ранжированного ряда с суммой равной 1.

Таким образом, элемент  $X_n$  при фильтрации заменяется элементом  $X_{n+1}$ , яркость которого равна медианному значению среди выборки пикселей (таблица 3).

Таблица 3 – Выборка пикселей, упорядоченных по ранжированию

| № элемента | 0     | 1         | 2         | 3         | 4         |
|------------|-------|-----------|-----------|-----------|-----------|
| Пиксель    | $X_n$ | $X_{n-1}$ | $X_{n+1}$ | $X_{n-2}$ | $X_{n+2}$ |
| Яркость    | 20    | 37        | 54        | 62        | 83        |

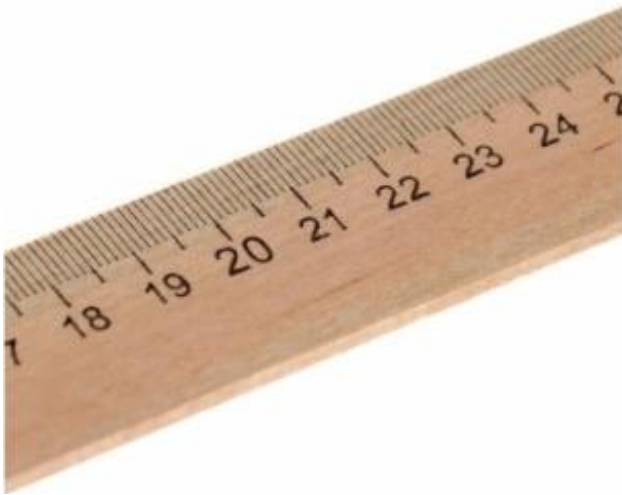
Значение яркости для текущего пикселя должно заменяться отфильтрованным значением, если его яркость превышает порог, установленный на основе яркости соседних пикселей. Данный порог может быть адаптирован в зависимости от особенностей изображения. Алгоритм эффективно устраняет импульсные помехи, сохраняя резкость изображения, однако может приводить к потере мелких объектов размеров от 1 до 3 пикселей, которые не являются помехами.



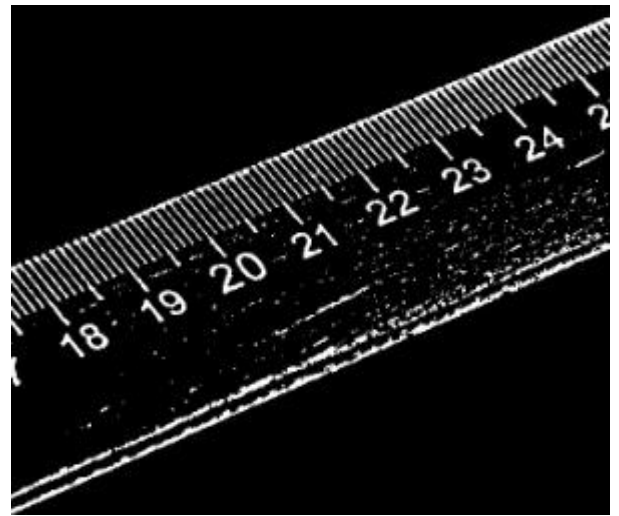
а)



б)



в)



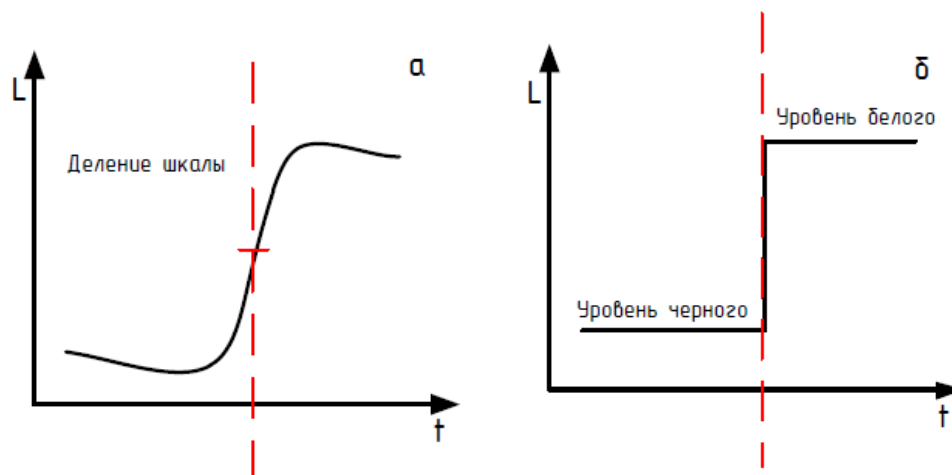
г)

а, в – исходное изображение;

б, г – изображение после применения медианного фильтра

Рисунок 15 - Результат применения медианного фильтра

Процесс бинаризации подразумевает преобразование полутонового или цветного изображения в бинарное (черно-белое). Бинаризация изображения по яркостному перепаду или пороговая проиллюстрирована на рисунке 16.

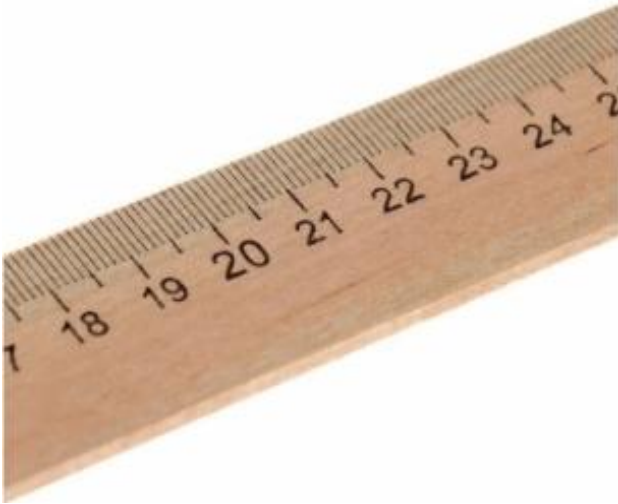


а - исходный видеосигнал; б - бинаризованный видеосигнал;

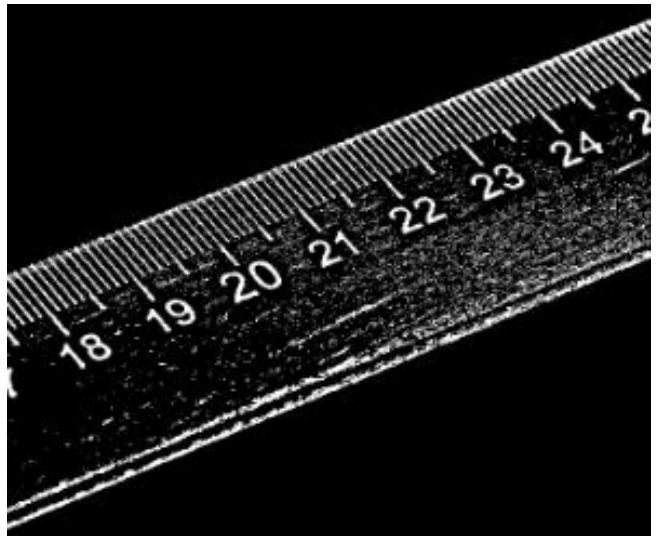
$L$ -яркость изображения;  $t$ -время

Рисунок 16 – Графическое представление фрагмента строки до и после процесса бинаризации

Наиболее простой метод пороговой обработки состоит в классификации с помощью единого порога каждого пикселя исходного изображения на две части – объект и фон. Ключевым моментом для успешного применения метода является корректный выбор порога  $t$ , оптимальное значение которого расположено в области минимума (впадине) между двумя пиками на гистограмме яркости изображения. На рисунке 17 представлено как такой подход позволяет эффективно отделить полезный сигнал (объект) от фона.



а)



б)



в)



г)

а, в – исходное изображение;

б, г – бинаризованное изображение

Рисунок 17 - Результат применения пороговой бинаризации

Основная цель бинаризации заключается в радикальном уменьшении объема информации, что существенно упрощает дальнейшую работу с изображением [16,17,18].

## 2.3 Детектирование границ и определение положения на видеоизображении делений шкал угломера

После завершения процедуры фильтрации данные поступают в детектор границ, данный процесс направлен на выявление резких изменений интенсивности яркости, которые соответствуют границам объектов (между штрихами шкалы и фоном). Детектирование используется для определения перепада среза и фронта видеосигнала, то есть действительного положения деления угломера в строке. Граница на изображении соответствует точке, где происходит резкое изменение значения яркости сигнала с черного на белый, графически это показано на рисунке 18.

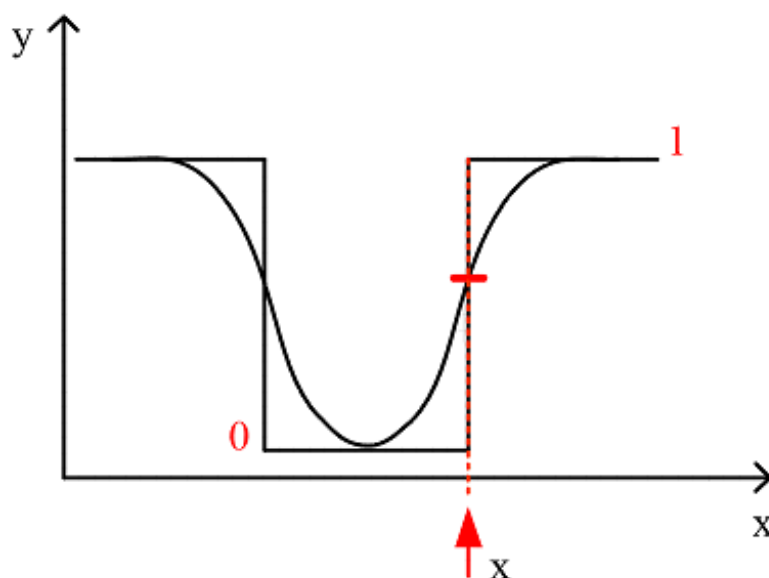


Рисунок 18 - Перепад яркостного сигнала строки изображения деления лимба

При определении положения делений шкалы угломера детектирование границ позволяет опередить точные координаты краев (фронта и среза видеосигнала).

В этом случае архитектура видеопроцессора предполагает наличие регистров для фиксации координат пикселей, соответствующих яркостному перепаду в верхней ( $x_T$ ) и нижней ( $x_B$ ) строках изображения [19].

## 2.4 Вычисление значений углов по положению на видеоизображении делений шкал угломера

Полученные координаты пикселей от детектора границ яркостного перепада в верхней ( $x_T$ ) и нижней ( $x_B$ ) строках используются для вычисления расстояния между ними в пикселях:

$$x = x_T - x_B \quad (14)$$

Для пересчета данного расстояния в угловую меру необходим коэффициент преобразования  $N$  числа пикселя в угол поворота. Его можно определить с помощью коэффициента пересчета видеоизображения, который характеризует разрешающую способность системы и выражается в количестве пикселей, приходящихся на одно малое деление угломера:

$$R = 1 \text{ м. д. у.} / N_p, \quad (15)$$

где  $N_p$  - количество пикселей укладываемые в одном малом делении угломера.

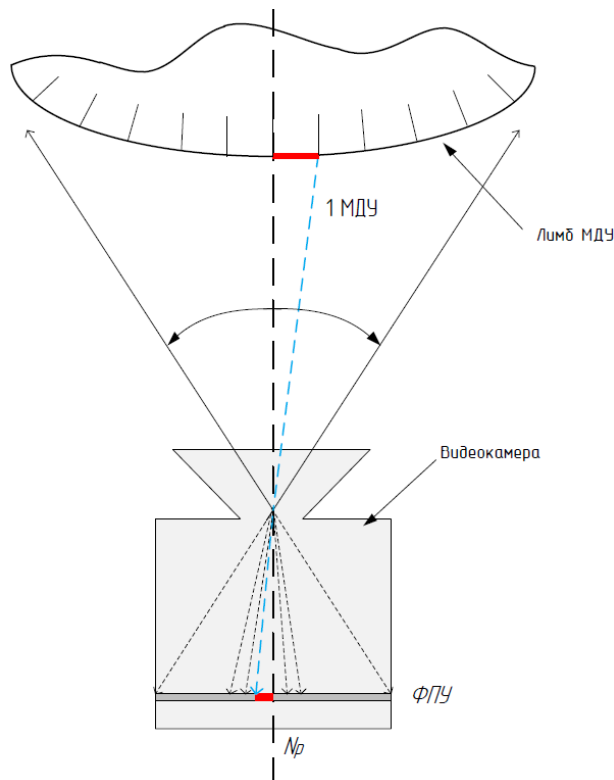


Рисунок 19 – Проецирование участка лимба с одним малым делением угломера на фотоприемное устройство

Следует отметить, что величина коэффициента преобразования зависит от расстояния между камерой и лимбом. Сужение поля зрения камеры повышает ее разрешающую способность, позволяя разместить больше пикселей в пределах одного малого деления угломера. Однако это требует более точной юстировки камеры относительно барабана, так как отсчет малых делений угломера основан на фиксации прохождения делений лимба мимо опорного деления – событийный отсчет. При этом погрешность измерения расстояния между малыми делениями лимба в пикселях влияет только на определение сотых долей МДУ.

Однако погрешность в измерениях МДУ и БДУ может возникнуть в случае, если скорость движения рисков превышает частоту кадров видеокамеры (смазывание кадра), что приводит к пропуску события. Следовательно, накладываются ограничения на скорость вращения лимба. Для расчета максимальной скорости маховика данной установки будут использованы исходные данные, представленные в таблице 4.

Таблица 4 – Исходные данные

| Наименование                                     | Значение |
|--------------------------------------------------|----------|
| Частота кадров видеокамеры $f_k$ , Гц            | 25       |
| Расстояние между делениями $L_{\text{дел}}$ , мм | 1        |
| Диаметр барабана маховика $d_{\text{мах}}$ , мм  | 40       |

Для определения периода кадровой развертки, используя формула (3), что соответствует времени, меньше которого риски не могут проходить мимо опорного деления, если деления будут проходить с частотой, превышающей период кадровой развертки, это приведет к пропуску подсчетов.

$$T_k = \frac{1}{f_k} = 40 \text{ мс} \quad (16)$$

Далее рассчитывается длина полного оборота:

$$L_{max} = \pi d_{max} = 125,6 \text{ мм} \quad (17)$$

На основании имеющихся данных определяется максимальная скорость движения делений:

$$V_{max} = \frac{L_{дел}}{T_k} = 25 \text{ мм/с} \quad (18)$$

В результате максимальная угловая скорость вращения маховика в данном случае будет равна:

$$\omega_{max} = \frac{V_{max}}{L_{max}} \cong 0,2 \text{ об/с} \quad (19)$$

Таким образом, поворот может осуществляться не более чем на 20% от полного оборота за одну секунду иначе может, наблюдается смазывание кадра, что приводит к пропуску события. На рисунке 20 приведен пример смазывания кадра при съемке движущегося объекта.



Рисунок 20 – Смазывание кадра движущегося объекта

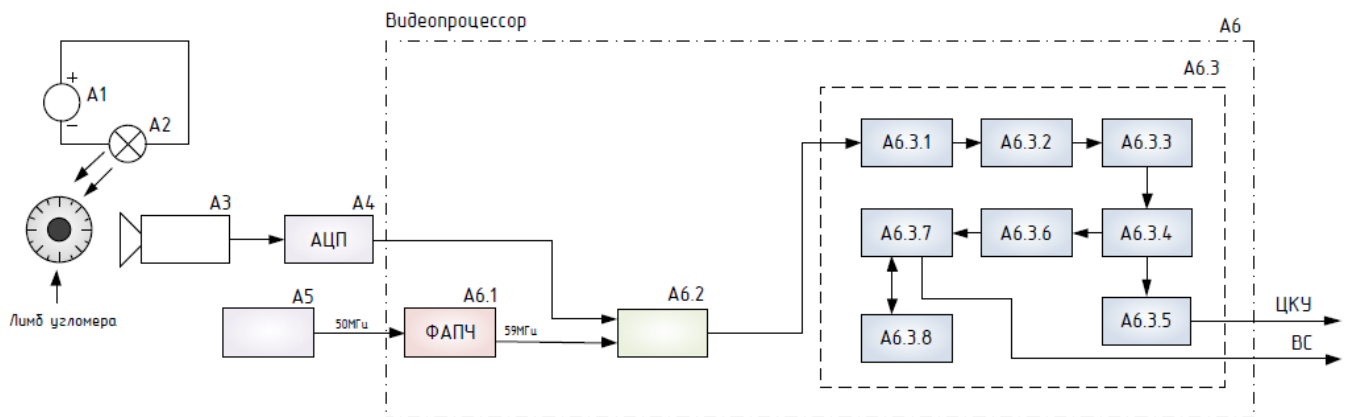
Изображение разбивается на два поля – четные и нечетные строки. Каждое поле содержит половину информации о кадре, что способствует уменьшению объема данных, передаваемых за один раз, и обеспечивает плавное отображение движущихся объектов. Первоначально передается поле с четными строчками, за которым следует передача нечетных строк. Данная

последовательность исключает необходимость одновременной обработки обоих полей, что оптимизирует время обработки. Развертка видеосигнала осуществляется по строчному принципу, подразумевающему последовательную передачу четных и нечетных строк. Это соседствующее представление строк способствует увеличению частоты кадров [20,21]. Параллельный контроль в четном и нечетном полях позволяет удвоить максимальную угловую скорость и сделать ее равной  $\omega_{max} = 0,4$  об/с.

### 3 Разработка структурной схемы видеопроцессора средств технического зрения для распознавания перемещения и считывания показаний шкал

#### 3.1 Структура телевизионной приставки для механического угломера

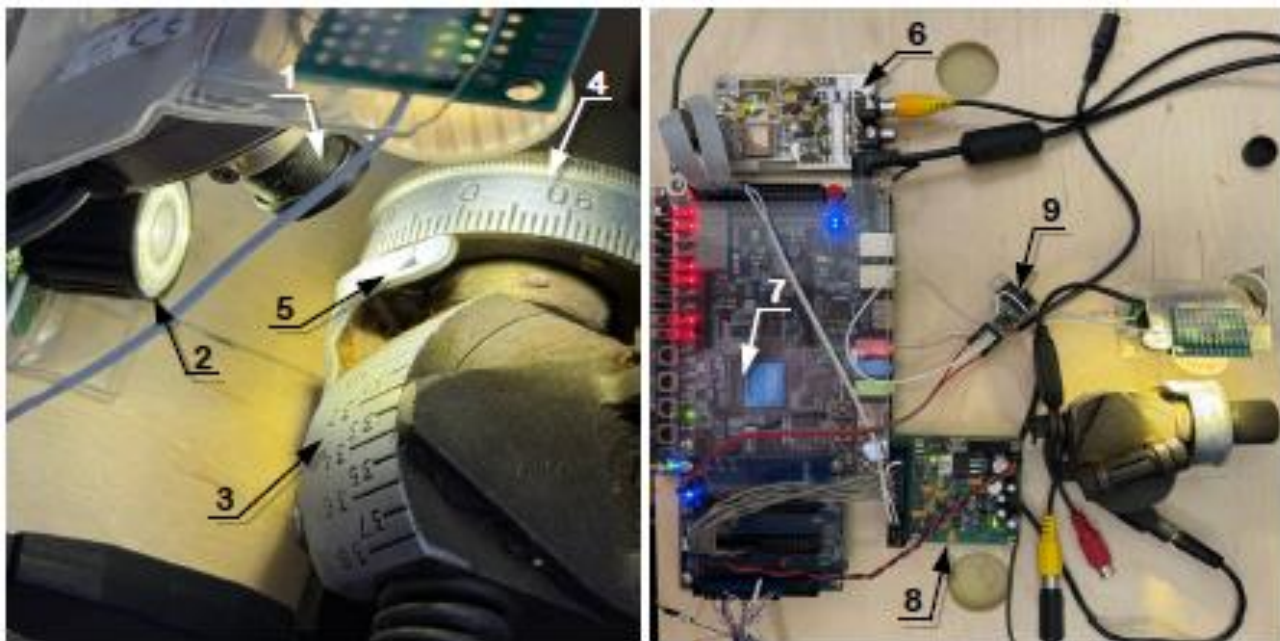
Основные этапы методики обработки изображения реализованы на основе специализированного видеопроцессора, структурная схема которого представлена на рисунке 21.



A1-источник питания; A2-светодиодный источник света;  
A3-видеокамера; A4-аналогово – цифровой преобразователь (АЦП);  
A5 - задающий генератор; A6-видеопроцессор; A6.1 - фазовая автоподстройка частоты (ФАПЧ); A6.2 – блок видеозахвата; A6.3.6-блок вывода графической информации; ЦКУ – вывод цифрового кода угла;  
ВС - вывод видеосигнал

Рисунок 21 – Структурная схема видеопроцессора

Функциональность и корректность работы схемы подтверждена на действующем макете, общий вид которого показан на рисунке 22.



а)

б)

а) 1 – видеокамера; 2 - источники света; 3 - лимб БДУ; 4 - лимб МДУ;  
5 - опорная риска; б) 6 - цифроаналоговый преобразователь; 7- ПЛИС;  
8 - видео АЦП; 9 - источник тока для питания светодиодов

Рисунок 22 – Действующий макет

Светодиодный источник света (А2) питается от источника питания (А1). Светодиод создает контролируемое, стабильно и равномерное освещение шкалы угломера, позволяя четко выделить объект на фоне для видеокамеры (А3), которая преобразует оптическое изображение в электрический сигнал – последовательность видеокадров. Далее происходит преобразование аналогового сигнала с выхода видеосенсора камеры в дискретную цифровую форму (двоичный код) с помощью цифрового преобразователя (А4). Задающий генератор синхронизирует работу видеопроцессора, подавая на фазовую автоподстройку частоты (ФАПЧ) частоту равную 50МГц, которая на выходе соответствует 59МГц. Данная частота обеспечивает дискретизацию видеосигнала в соответствии с длительностью строк видеосигнала.

### 3.2 Структура видеопроцессора средств технического зрения для распознавания перемещения и считывания показаний шкал

В состав видеопроцессора представленном на рисунке 1-А6, входит блок А6.2, отвечающий за формирование 10-разрядного кода аналогово-цифрового преобразователя. В процессе работы данный блок производит преобразование отсчетов видеосигнала с уровнем чёрного, зависящим от постоянной составляющей сигнала, в нормированную систему координат, где код со значением «0» соответствует черному цвету, а «716» – белому. Блок А6.2 формирует мгновенное значение яркости, а также соответствующие ему координату пикселя в строке и кадре, с последующей селекцией строчных и кадровых синхроимпульсов. Полученные сигналы служат ориентиром для остальных функциональных узлов видеопроцессора, позволяя более точно определить начало и конец каждого кадра. В случае исчезновения синхроимпульса в отсутствии входного видеосигнала, блок А6.2 продолжает генерировать синхроимпульсы, тем самым обеспечивая непрерывность передачи. При передаче видеосигнала в аналоговом формате, длительность синхроимпульса может искажаться, что негативно сказывается на качестве видеоизображения, поэтому данное искажение корректируется в блоке.

В блоке А6.3 осуществляется фильтрация изображения, которая состоит из трех последовательных стадий, с последующим вычислением смещения шкалы лимба угломера по изображению. За первую стадию фильтрации отвечает усредняющий фильтр видеоизображения (блок А6.3.1). При усреднении яркости пикселя учитываются значения яркости пикселей, расположенных в окрестности целевого пикселя для последующего расчета среднего значения. Расчет среднего значения осуществляется путем суммирования значений яркости всех пикселей в области и деления полученной суммы на общее количество пикселей. В результате яркость целевого пикселя заменяется усредненным значением, что способствует сглаживанию изображения и уменьшению шума.

На следующем этапе обработки изображения, осуществляемом в блоке А6.3.2, применяется медианный фильтр, который служит для оценки наиболее вероятного значения яркости сигнала для каждого пикселя. Данный фильтр выполняет ранжирование значений яркости в заданной области по возрастанию, после чего медиана сформированного ряда значений представляет собой наиболее вероятное значение яркости, соответствующее рассматриваемому пикселю. В результате, происходит дальнейшее ослабление шумов.

Третья стадия фильтрации включает процесс бинаризации, осуществляемый в блоке А6.3.3. Данная процедура преобразует пиксели с серыми значениями в черные, тогда как пиксели с более светлыми значениями в белые. Это приводит к значительному уменьшению объема информации за счет трансформации изображения в четкое черно-белое представление.

После завершения процедуры фильтрации данные поступают в детектор границ (блок А6.3.4) для определения перепада среза и фронта на выходе, то есть действительное положение деления угломера в строке. В данном блоке производится вычисление угла на основе разности между значениями пикселей, что позволяет определить, на сколько делений нижняя риска отстает от фиксированной верхней риски. Вычисления производятся в пределах диапазона делений угломера, что обеспечивает более дискретное значение. Таким образом, результатом работы данного блока является определение положения как больших делений угломера (БДУ), так и малых (МДУ).

Затем полученное изображение в блоке А6.3.6 дополняется двумя линиями, одна из которых показывает положение деления лимба, а вторая определяет положение опорного деления. Данные линии служат визуальным ориентиром для пользователя.

Для отображения числовых значений, полученных в блоке А6.3.5, используется генератор символов (блок А6.3.8), который осуществляет вывод

данных на экран монитора. Таблица символов, используемая для отображения чисел, представляет собой постоянное запоминающее устройство (ПЗУ), в котором хранится пиксельный образ каждого символа.

Завершающий блок А6.3.7 форматирует видеоданные для формирования входного видеосигнала, соответствующего стандартам ГОСТ [22,23,24]. Процесс включает преобразование видеоданных в формат, совместимый с требованиями видеомониторов, что гарантирует корректное отображение информации и стабильность работы устройства в различных условиях эксплуатации [25].

## 4 Разработка поведенческих моделей и синтез электрических схем функциональных узлов видеопроцессора

### 4.1 Усредняющий фильтр видеоизображения

Фрагмент описания структуры усредняющего фильтра видеоизображения приведен на языке описания электрических схем *Verilog* в таблице 5.

Таблица 5 - Фрагмент усредняющего фильтра видеоизображения на языке *Verilog*

| № | Содержимое строки                                                                                                                                                                                                                                                                                                                                                                                     | Комментарий                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
|---|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1 | <pre> module AvrFLT (inHVblank, inSSC, inVData, inNpix, inNstr, inCLK, inFltOn, outHVblank, outSSC, outVData, outNpix, outNstr);  input inHVblank;  input inSSC;  input [9:0] inVData;  input [10:0] inNpix;  input [9:0] inNstr;  input inCLK;  input inFltOn;  output reg outHVblank;  output reg outSSC;  output reg [9:0] outVData;  output reg [10:0] outNpix;  output reg [9:0] outNstr; </pre> | <p><i>inHVblank</i> – вход сигнала строчного и кадрового гашения;</p> <p><i>inSSC</i> – вход сигнала строчно-кадровых синхроимпульсов;</p> <p><i>inVData</i> – вход цифрового яркостного сигнала;</p> <p><i>inNpix</i> – входной двоичный код номера пикселя;</p> <p><i>inNstr</i> – входной двоичный код номера строки изображения;</p> <p><i>inCLK</i> – входной тактовый сигнал дискретизации видеосигнала;</p> <p><i>inFltOn</i> – вход управления включением/выключением фильтра;</p> <p><i>outHVblank</i> – выход сигнала строчного и кадрового гашения;</p> <p><i>outSSC</i> – выход строчно-кадровых синхроимпульсов;</p> <p><i>outVData</i> – выход цифрового яркостного сигнала;</p> <p><i>outNpix</i> – выход двоичного</p> |

|   |                                                                                                                                                                                                                                                                    |                                                                                                                                                                                                                                             |
|---|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|   |                                                                                                                                                                                                                                                                    | <p>кода номера пикселя в строке;</p> <p><i>outNstr</i> – выход двоичного кода номера строки.</p>                                                                                                                                            |
| 2 | <pre>reg [9:0] Stream [0:4]; reg [22:0] SHV [0:2];</pre>                                                                                                                                                                                                           | <i>Stream</i> и <i>SHV</i> - вспомогательные регистры.                                                                                                                                                                                      |
| 3 | <pre>wire [11:0] AVR = Stream [1] [9:0] + Stream [3] [9:0] + Stream [0] [9:0] + Stream [2] [9:0];</pre>                                                                                                                                                            | Сложение сохраненных значений яркостных сигналов четырех пикселей.                                                                                                                                                                          |
| 4 | <pre>always @(posedge inCLK) begin</pre>                                                                                                                                                                                                                           | Процедурный блок, описывающий работу схемы, синхронизируемой тактовым сигналом <i>inCLK</i> .                                                                                                                                               |
| 5 | <pre>Stream [0] [9:0] &lt;= inVData; Stream [1] [9:0] &lt;= Stream [0] [9:0]; Stream [2] [9:0] &lt;= Stream [1] [9:0]; Stream [3] [9:0] &lt;= Stream [2] [9:0];</pre>                                                                                              | Последовательная запись значений яркостного сигнала четырех пикселей благодаря выставленной последовательности из <i>Stream</i> регистров.                                                                                                  |
| 6 | <pre>outVData &lt;= ( inFltOn ) ? AVR [11:2] : Stream [2] [9:0];</pre>                                                                                                                                                                                             | Деление на 4 полученной суммы яркостного сигнала четырех пикселей путем отбрасывания двух младших разрядов.                                                                                                                                 |
| 7 | <pre>SHV [0] [22:0] &lt;= {inNstr, inNpix, inSSC, inHVblank}; SHV [1] [22:0] &lt;= SHV [0] [22:0]; SHV [2] [22:0] &lt;= SHV [1] [22:0]; outHVblank &lt;= SHV [2] [0]; outSSC &lt;= SHV [2] [1]; outNpix &lt;= SHV [2] [12:2]; outNstr &lt;= SHV [2] [22:13];</pre> | Три дополнительных регистра <i>SHV</i> используются для синхронизации передачи яркостного сигнала и временных синхросигналов, то есть триггеры выравнивают по времени вывод синхросигналов с яркостным сигналом. Таким образом, компенсируя |

|   |                      |                                                                                                                                                                    |
|---|----------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|   |                      | задержку времени прохождения яркостных сигналов, чтобы они шли единым фронтом и соответствовали номеру пикселю и номеру строки, приходящим на вход мультиплексора. |
| 8 | end<br><br>endmodule | Конец модуля.                                                                                                                                                      |

Для получения структурной электрической схемы на основе *Verilog*-описания модуля необходимо выполнить следующие действия: *Tools* → *Netlist Viewers* → *RTL Viewer*. Синтезированная структурная электрическая схема модуля усредняющего фильтра в среде *Quartus II* [26,27], показанная на рисунке 23.

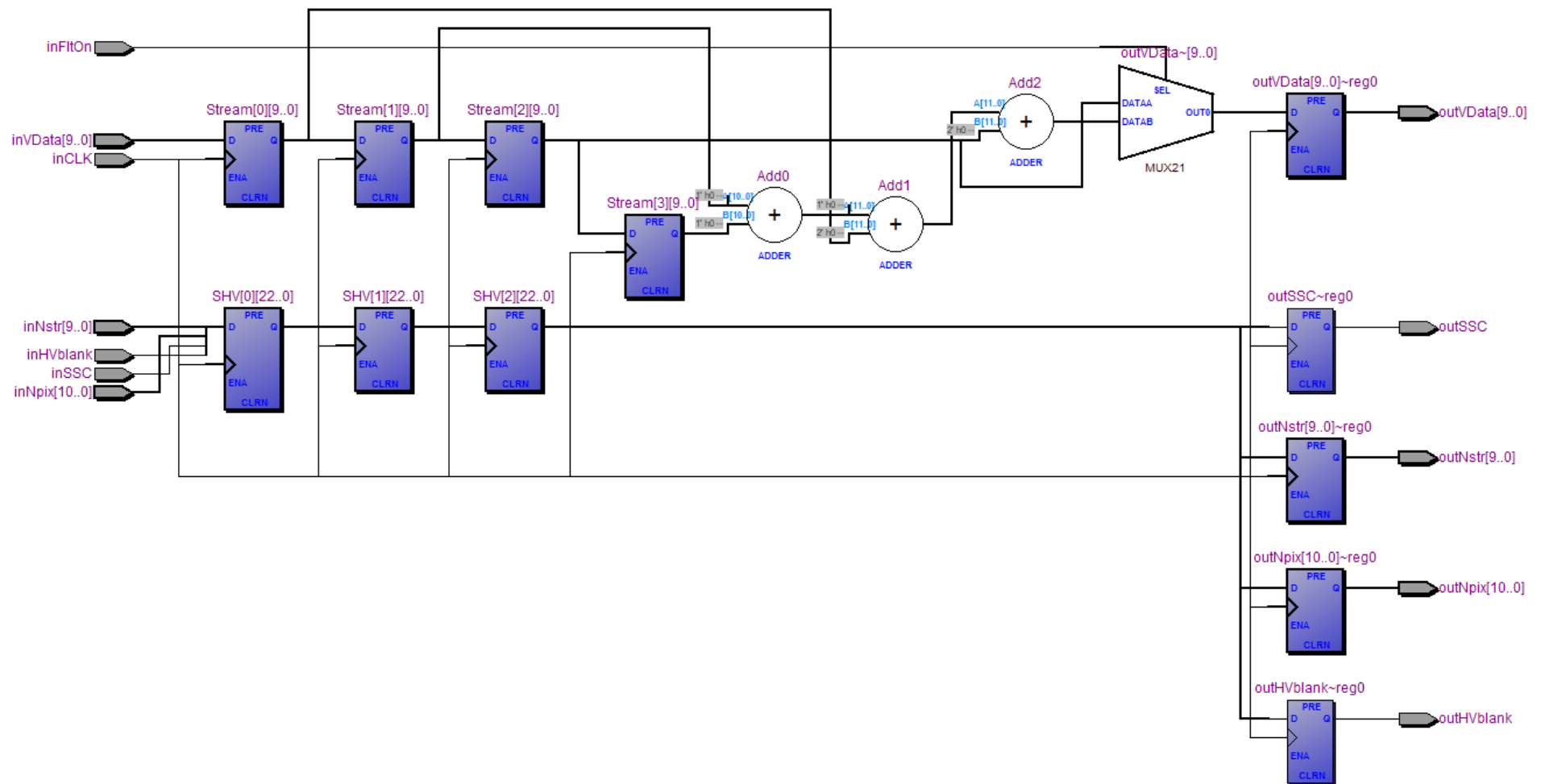


Рисунок 23 – Синтезированная по *Verilog* структурная электрическая схема усредняющего фильтр видеозображения

## 4.2 Медианный фильтр видеоизображения

Фрагмент структуры медианного фильтра видеоизображения описывается на языке описания электрических схем *Verilog* в таблице 6.

Таблица 6 - Фрагмент медианного фильтра видеоизображения на языке *Verilog*

| № | Содержимое строки                                                                                                                                                                                                                                                                                                                                                                                                                                         | Комментарий                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
|---|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1 | <pre> module MedianFLT (inHVblank, inSSC, inVData, inNpix, inNstr, inCLK, inFltOn, outHVblank, outSSC, outVData, outNpix, outNstr);  input inHVblank;  input inSSC;  input [9:0] inVData;  input [10:0] inNpix;  input [9:0] inNstr;  input inCLK;  input inFltOn;  output reg outHVblank;  output reg outSSC;  output reg [9:0] outVData;  output reg [10:0] outNpix;  output reg [9:0] outNstr;  reg [9:0] Stream [0:4];  wire [2:0] Summ [0:4]; </pre> | <p><i>inHVblank</i> – вход сигнала строчного и кадрового гашения;</p> <p><i>inSSC</i> – вход сигнала строчно-кадровых синхроимпульсов;</p> <p><i>inVData</i> – вход цифрового яркостного сигнала;</p> <p><i>inNpix</i> – входной двоичный код номера пикселя;</p> <p><i>inNstr</i> – входной двоичный код номера строки изображения;</p> <p><i>inCLK</i> – входной тактовый сигнал дискретизации видеосигнала;</p> <p><i>inFltOn</i> – вход управления включением/выключением фильтра;</p> <p><i>outHVblank</i> – выход сигнала строчного и кадрового гашения;</p> <p><i>outSSC</i> – выход строчно-кадровых синхроимпульсов;</p> <p><i>outVData</i> – выход цифрового яркостного сигнала;</p> <p><i>outNpix</i> – выход двоичного кода номера пикселя в строке;</p> <p><i>outNstr</i> – выход двоичного кода номера строки;</p> <p><i>Stream</i> и <i>Summ</i> – вспомогательные триггеры.</p> |

|   |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |                                                                                                                                                                                                 |
|---|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 2 | <pre> assign Summ [0] [2:0] = (Stream [1] [9:0] &lt; Stream [0] [9:0]) +         (Stream [2] [9:0] &lt; Stream [0] [9:0]) +         (Stream [3] [9:0] &lt; Stream [0] [9:0]) +         (Stream [4] [9:0] &lt; Stream [0] [9:0]); assign Summ [1] [2:0] = (Stream [0] [9:0] &lt; Stream [1] [9:0]) +         (Stream [2] [9:0] &lt; Stream [1] [9:0]) +         (Stream [3] [9:0] &lt; Stream [1] [9:0]) +         (Stream [4] [9:0] &lt; Stream [1] [9:0]); assign Summ [2] [2:0] = (Stream [0] [9:0] &lt; Stream [2] [9:0]) +         (Stream [1] [9:0] &lt; Stream [2] [9:0]) +         (Stream [3] [9:0] &lt; Stream [2] [9:0]) +         (Stream [4] [9:0] &lt; Stream [2] [9:0]); assign Summ [3] [2:0] = (Stream [0] [9:0] &lt; Stream [3] [9:0]) +         (Stream [1] [9:0] &lt; Stream [3] [9:0]) +         (Stream [2] [9:0] &lt; Stream [3] [9:0]) +         (Stream [4] [9:0] &lt; Stream [3] [9:0]); assign Summ [4] [2:0] = (Stream [0] [9:0] &lt; Stream [4] [9:0]) +         (Stream [1] [9:0] &lt; Stream [4] [9:0]) +         (Stream [2] [9:0] &lt; Stream [4] [9:0]) +         (Stream [3] [9:0] &lt; Stream [4] [9:0]); wire [2:0] Num0x = (Summ [0] == 3'd0) +         (Summ [1] == 3'd0) +         (Summ [2] == 3'd0) +         (Summ [3] == 3'd0) +         (Summ [4] == 3'd0); wire [2:0] Num1x = (Summ [0] == 3'd1) +         (Summ [1] == 3'd1) + </pre> | <p>Данный блок реализует алгоритм ранжирования и выборки медианного значения (см. пункт 2.2) с учетом дополнительного условия – значение двойки не обнаружено или обнаружена несколько раз.</p> |
|---|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|

```

(Summ [2] == 3'd1) +
(Summ [3] == 3'd1) +
(Summ [4] == 3'd1);
wire [9:0] Mediana = (Summ [0] == 3'd2) ? Stream [0]
[9:0] :
((Summ [1] == 3'd2) ? Stream [1] [9:0] :
((Summ [2] == 3'd2) ? Stream [2] [9:0] :
((Summ [3] == 3'd2) ? Stream [3] [9:0] :
((Summ [4] == 3'd2) ? Stream [4] [9:0] :
((Num0x > 3'd2) ? (
(Summ [0] == 3'd0) ? Stream [0] [9:0] :
((Summ [1] == 3'd0) ? Stream [1] [9:0] :
Stream [0] [9:0])) :
((Num1x > 3'd1) ? (
(Summ [0] == 3'd1) ? Stream [0] [9:0] :
((Summ [1] == 3'd1) ? Stream [1] [9:0] :
((Summ [2] == 3'd1) ? Stream [2] [9:0] :
Stream [3] [9:0])) ) : Stream [2] [9:0]))))));
wire [9:0] Difference12x = (Stream [2] [9:0] > Stream [1]
[9:0]) ? (Stream [2] [9:0] - Stream [1] [9:0]) :
(Stream [1] [9:0] - Stream [2] [9:0]);
wire [9:0] Difference23x = (Stream [2] [9:0] > Stream [3]
[9:0]) ? (Stream [2] [9:0] - Stream [3] [9:0]) :
(Stream [3] [9:0] - Stream [2] [9:0]);
reg [22:0] SHV [0:2];

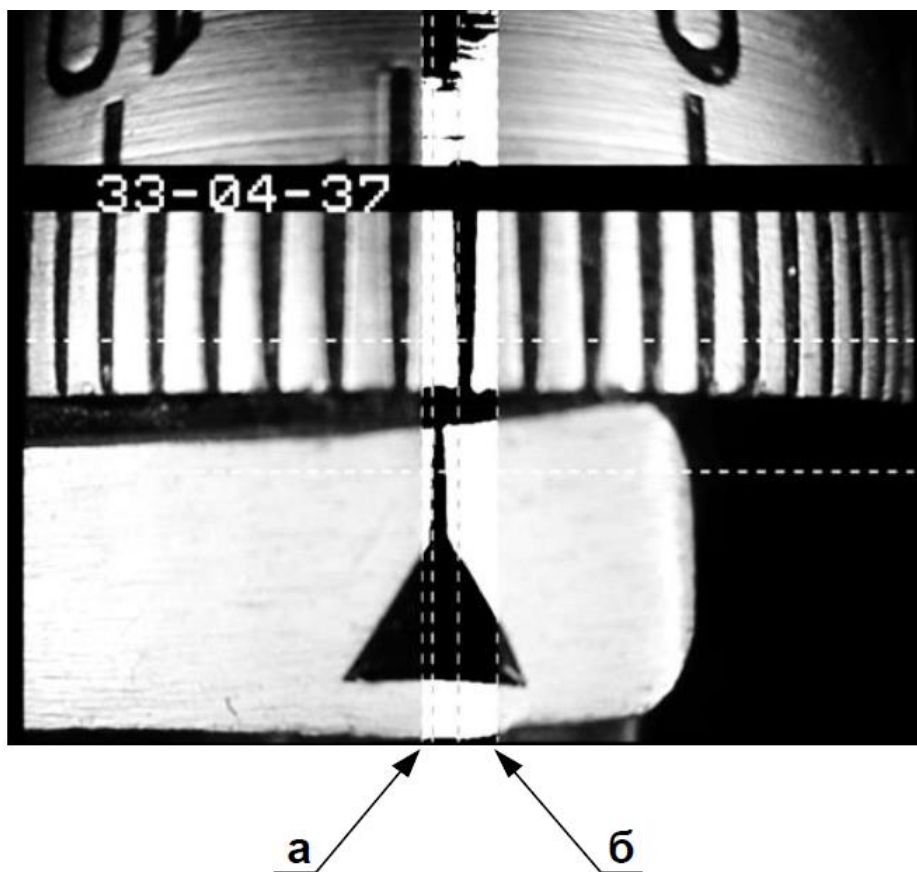
```

|   |                                                                                                                                                                                                                                                                              |                                                                                                        |
|---|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------|
| 3 | always @(negedge inCLK)<br><br>begin                                                                                                                                                                                                                                         | Процедурный блок,<br>описывающий работу<br>схемы, синхронизируемой<br>тактовым сигналом <i>inCLK</i> . |
| 4 | Stream [0] [9:0] <= inVData;<br><br>Stream [1] [9:0] <= Stream [0] [9:0];<br><br>Stream [2] [9:0] <= Stream [1] [9:0];<br><br>Stream [3] [9:0] <= Stream [2] [9:0];<br><br>Stream [4] [9:0] <= Stream [3] [9:0];<br><br>outVData <= (inFltOn) ? Mediana : Stream [2] [9:0];  | Сохранение<br>последовательности из 5<br>пикселей.                                                     |
| 5 | SHV [0] [22:0] <= {inNstr, inNpix, inSSC, inHVblank};<br><br>SHV [1] [22:0] <= SHV [0] [22:0];<br><br>SHV [2] [22:0] <= SHV [1] [22:0];<br><br>outHVblank <= SHV [2] [0];<br><br>outSSC <= SHV [2] [1];<br><br>outNpix <= SHV [2] [12:2];<br><br>outNstr <= SHV [2] [22:13]; | Выравнивающие <i>SHV</i><br>регистры по времени.                                                       |
| 6 | end<br><br>endmodule                                                                                                                                                                                                                                                         | Конец модуля.                                                                                          |

Полный листинг кода блока медианной фильтрации приведен в приложении Ж, а также синтезированная структурная электрическая схема по *Verilog* описанию модуля представлена в приложениях З.

### 4.3 Бинаризатор видеоизображения

Процесс бинаризации на экране монитора отображается в реальном времени в виде «полосы», в зоне которой осуществляется поиск делений (рисунок 24 – а, б). Фрагмент блока бинаризации, приведен на языке описания электрических схем *Verilog* в таблице 7.



а и б - границы зоны, в пределах которой осуществляется поиск делений на изображении в контрольных строках

Рисунок 24 - Выходное видеоизображение

Таблица 7 - Фрагмент блока бинаризации видеоизображения на языке *Verilog*

| № | Содержимое строки                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             | Комментарий                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
|---|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1 | <pre> module Binarizator(inHVblank, inHS, inY, inNpix, inNstr, inCLK, outHVB, outHS, outY, outNpix, outNstr, LeftVLine, RightVLine, TopHLine, BotHLine );  input inHVblank;  input inHS;  input [9:0] inY;  input [10:0] inNpix;  input [9:0] inNstr;  input inCLK;  output reg outHVB;  output reg outHS;  output reg [9:0] outY;  output reg [10:0] outNpix;  output reg [9:0] outNstr;  output [10:0] LeftVLine = 11'd692;  output [10:0] RightVLine = 11'd820;  output [9:0] TopHLine = 10'd260;  output [9:0] BotHLine = 10'd362; </pre> | <p><i>inHVblank</i> – вход сигнала строчного и кадрового гашения;</p> <p><i>inY</i> – входной двоичный код сигнала яркости;</p> <p><i>inNpix</i> – входной двоичный код номера пикселя;</p> <p><i>inNstr</i> – входной двоичный код номера строки изображения;</p> <p><i>inCLK</i> – входной тактовый сигнал дискретизации видеосигнала;</p> <p><i>inHS</i> – вход сигнала строчных синхроимпульсов;</p> <p><i>outY</i> – выходной двоичный код сигнала яркости;</p> <p><i>outNpix</i> – выход двоичного кода номера пикселя в строке;</p> <p><i>outNstr</i> – выход двоичного кода номера строки;</p> <p><i>outHS</i> – выход сигнала строчных синхроимпульсов;</p> <p><i>outHVB</i> – выходной сигнала строчного и кадрового гашения;</p> <p><i>LeftVLine</i> и <i>RightVLine</i> – входные двоичные коды позиции левой и правой границ области поиска яркостного перепада в строке изображения;</p> <p><i>TopHLine</i> и <i>BotHLine</i> – входные двоичные коды позиции по вертикали верхней и нижней строк</p> |

|   |                                                                                                                                                                                                                                                                                                                                                                                                                                                          |                                                                                                                                                                                                                                                                                        |
|---|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|   |                                                                                                                                                                                                                                                                                                                                                                                                                                                          | изображения, в которых производится поиск яркостных перепадов;                                                                                                                                                                                                                         |
| 2 | <pre>wire [9:0] CenterLine = (TopHLine + BotHLine) &gt;&gt; 1; parameter YavrTop = 10'd150; parameter YavrBot = 10'd170; wire TopBar = (inNpix &gt; LeftVLine) &amp; (inNpix &lt; RightVLine) &amp; (inNstr &lt; CenterLine); wire BotBar = (inNpix &gt; LeftVLine) &amp; (inNpix &lt; RightVLine) &amp; (inNstr &gt; CenterLine); wire [9:0] TopY = (inY &gt; YavrTop) ? 10'd716 : 10'd0; wire [9:0] BotY = (inY &gt; YavrBot) ? 10'd716 : 10'd0;</pre> | Выделение верхней и нижней строки на основе сравнения координат пикселя с заданными границами и центральной линией                                                                                                                                                                     |
| 3 | <pre>always @(posedge inCLK) begin</pre>                                                                                                                                                                                                                                                                                                                                                                                                                 | Процедурный блок, описывающий работу схемы, синхронизируемой тактовым сигналом <i>inCLK</i> .                                                                                                                                                                                          |
| 4 | <pre>outY &lt;= TopBar ? TopY : (BotBar ? BotY : inY); outHVB &lt;= inHVblank; outHS &lt;= inHS; outNpix &lt;= inNpix; outNstr &lt;= inNstr; Stream [1] [9:0] &lt;= Stream [0] [9:0]; Stream [2] [9:0] &lt;= Stream [1] [9:0]; Stream [3] [9:0] &lt;= Stream [2] [9:0];</pre>                                                                                                                                                                            | <p>Замена пикселя на служебные уровни <i>TopY</i> и <i>BotY</i> в зонах, обозначенными сигналами <i>TopBar</i> и <i>BotBar</i>.</p> <p>А также реализуется сдвиговые регистры для задержки пикселей, что позволяет одновременно обрабатывать несколько соседних точек изображения.</p> |
| 5 | <pre>end endmodule</pre>                                                                                                                                                                                                                                                                                                                                                                                                                                 | Конец модуля.                                                                                                                                                                                                                                                                          |

Пороговые уровни яркостного сигнала для верхней (*parameter YavrTop*) и нижней (*parameter YavrBot*) строк видеокadra выбраны эмпирически из-за неравномерности яркости по кадру (150 и 170 соответственно). В данной

модели реализации методики используются фиксированные пороги бинаризации, однако схему можно развить дополнительным узлом, который позволит автоматически подстраивать пороги для конкретного участка по измерению статики видеоизображения [28,29].

На рисунке 25 приведена структурная электрическая схема, сформированная в результате синтеза модуля бинаризации, описанного на языке *Verilog*.

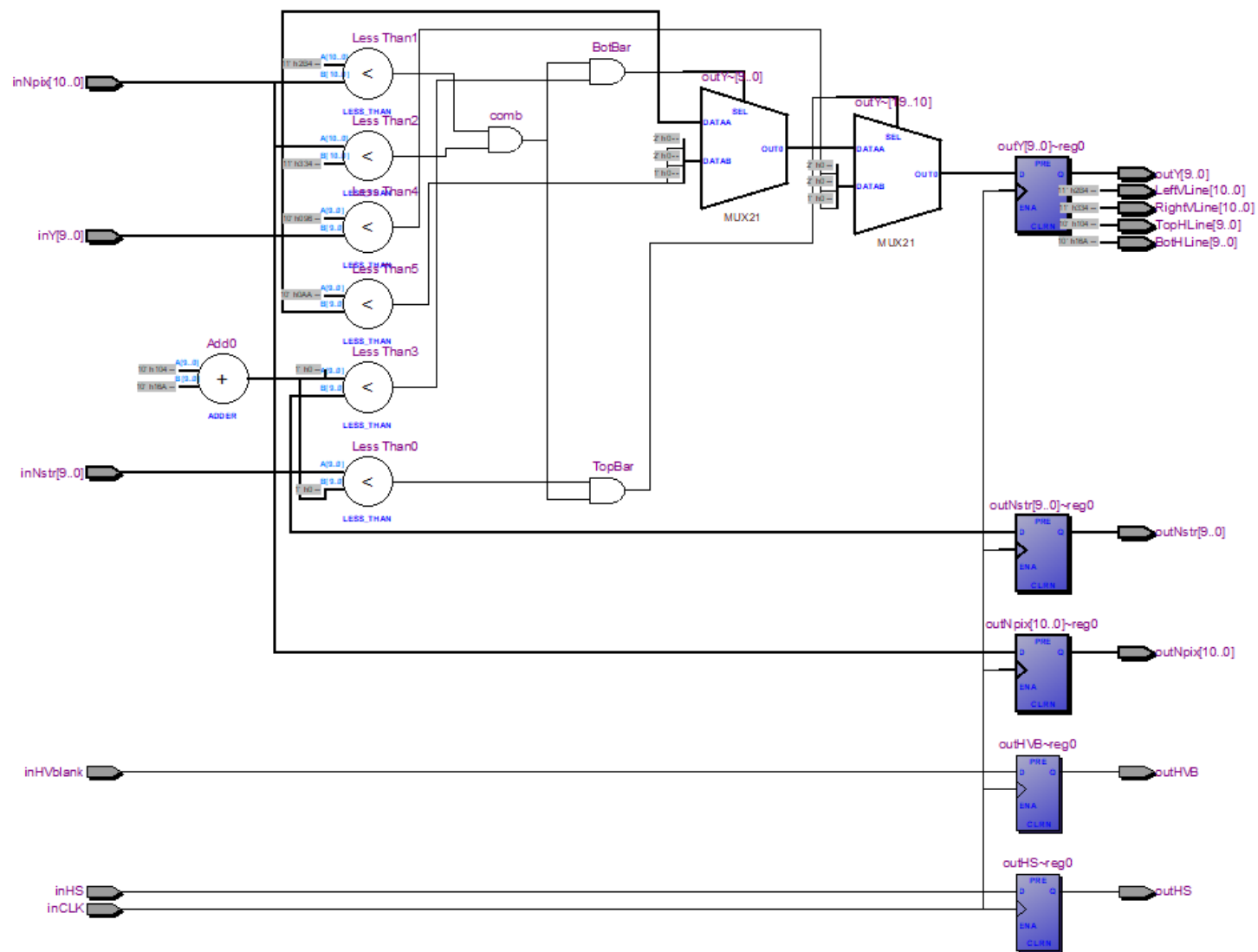


Рисунок 25 – Синтезированная по Verilog структурная электрическая схема блока бинаризации изображения

#### 4.4 Детектор границ делений шкалы

Фрагмент описания блока детектирования положения деления лимба на изображении описывается на языке описания электрических схем *Verilog* в таблице 8.

Таблица 8 - Фрагмент блока детектирования положения деления на языке *Verilog*

| № | Содержимое строки                                                                                                                                                                                                                                                                                                                                                                                                                                                               | Комментарий                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |
|---|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1 | <pre> module EdgeDetector (inY, inNpix, inNstr, inCLK, LeftVLine,    RightVLine,    TopHLine,    BotHLine, BotEdgeNpix, TopEdgeNpix, outY);  input [9:0] inY;  input [10:0] inNpix;  input [9:0] inNstr;  input inCLK;  input [10:0] LeftVLine;  input [10:0] RightVLine;  input [9:0] TopHLine;  input [9:0] BotHLine;  output reg [9:0] outC;  output reg [10:0] TopEdgeNpix;  output reg [10:0] BotEdgeNpix;  reg DetectEn;  reg oldMsbY;  reg TopEdge;  reg BotEdge; </pre> | <p><i>inY</i> – входной двоичный код сигнала яркости;</p> <p><i>inNpix</i> – входной двоичный код номера пикселя;</p> <p><i>inNstr</i> – входной двоичный код номера строки изображения;</p> <p><i>inCLK</i> – входной тактовый сигнал дискретизации видеосигнала;</p> <p><i>LeftVLine</i> и <i>RightVLine</i> – входные двоичные коды позиции левой и правой границ области поиска яркостного перепада в строке изображения;</p> <p><i>TopHLine</i> и <i>BotHLine</i> – входные двоичные коды позиции по вертикали верхней и нижней строк изображения, в которых производится поиск яркостных перепадов;</p> <p><i>BotEdgeNpix</i> и <i>TopEdgeNpix</i> – регистры для вывода двоичных кодов найденных позиций яркостных перепадов в верхней и нижней строках соответственно;</p> <p><i>DetectEn</i>, <i>oldMsbY</i>, <i>TopEdge</i>, <i>BotEdge</i> – вспомогательные</p> |

|   |                                                                                                                          |                                                                                                                                                                                                                 |
|---|--------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|   |                                                                                                                          | триггеры.                                                                                                                                                                                                       |
| 2 | always @(posedge inCLK)<br><br>begin                                                                                     | Процедурный блок, описывающий работу схемы, синхронизируемой тактовым сигналом <i>inCLK</i> .                                                                                                                   |
| 3 | oldMsbY <= inY[9];                                                                                                       | В регистр <i>oldMsbY</i> записывается яркость текущего пикселя.                                                                                                                                                 |
| 4 | DetectEn <= ((inNpix > LeftVLine) & (inNpix < RightVLine)) ? (oldMsbY & ~inY[9]) : 1'b0;                                 | В регистр <i>DetectEn</i> записывается значение «1» в случае, если яркость предыдущего пикселя равна «1», а последующего – «0», то есть фиксируется событие обнаружения яркостного перепада в окне изображения. |
| 5 | if (DetectEn & ((inNstr == TopHLine)   (inNstr == TopHLine + 1'b1)) & (inNpix >= BotEdgeNpix))<br>TopEdgeNpix <= inNpix; | В регистр <i>TopEdgeNpix</i> записывается номер текущего пикселя, в котором зафиксирован яростный перепад в верхней строке.                                                                                     |
| 6 | if (DetectEn & ((inNstr == BotHLine)   (inNstr == BotHLine + 1'b1))) BotEdgeNpix <= inNpix;                              | В регистр <i>BotEdgeNpix</i> записывается номер текущего пикселя, в котором зафиксирован яростный перепад в нижней строке.                                                                                      |
| 7 | TopEdge <= (TopEdgeNpix [10:1] == inNpix[10:1]);                                                                         | В триггеры <i>TopEdge</i> и <i>BotEdge</i> фиксируются позиции движущейся и опорной риски в результате сравнения двух диапазонов битов для их дальнейшего                                                       |
| 8 | BotEdge <= (BotEdgeNpix [10:1] == inNpix[10:1]);                                                                         |                                                                                                                                                                                                                 |

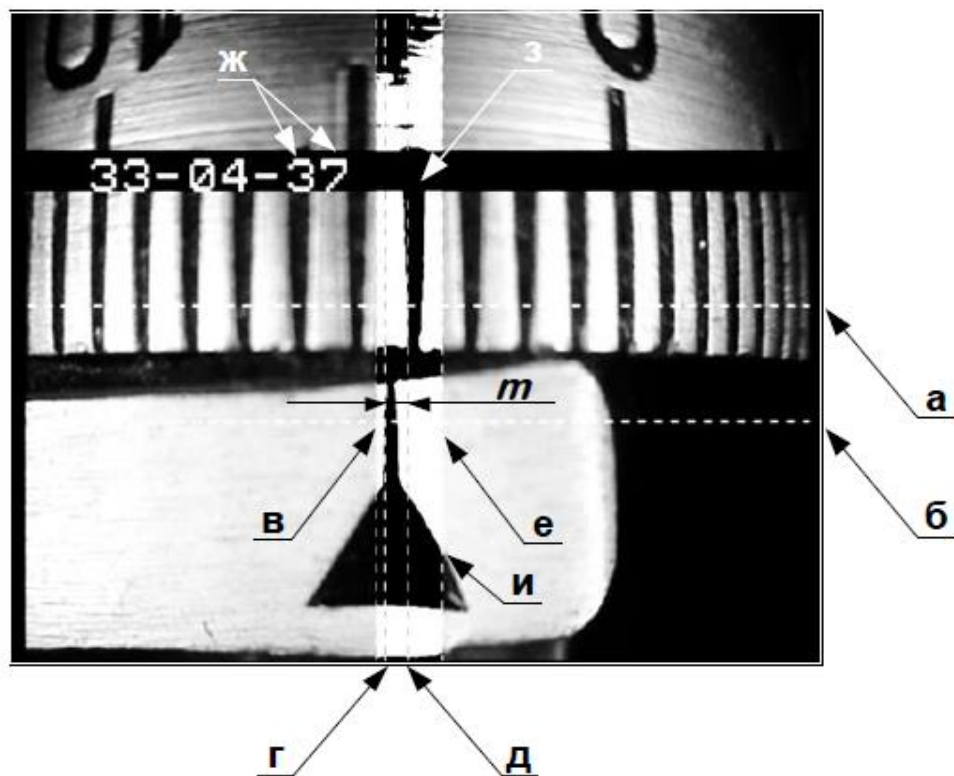
|    |                                                                                                              |                                                                                                                                                                        |
|----|--------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|    |                                                                                                              | выделения на изображении.                                                                                                                                              |
| 9  | $\text{OutY} \leq (\text{TopEdge} \mid \text{BotEdge}) ? (\text{inNstr}[3] ? 10'd716 : 10'd0) : \text{inY};$ | Формирование выходного яркостного сигнала <i>outY</i> . Изображение дополняется вертикальной пунктирной линией по сигналам триггеров <i>TopEdge</i> и <i>BotEdge</i> . |
| 10 | $\text{end}$ $\text{endmodule}$                                                                              | Конец модуля                                                                                                                                                           |

Данный блок дополняет видеоизображение лимба индикаторными линиями, которые представлены на рисунке 26:

- контрольные строки, *а* и *б* (рисунк1), используются для мониторинга изображения: в верхней строке располагается изображение подвижных делений, в нижней строке представлено изображение опорного деления;

- линия *г*, представленная на рисунке 1, обозначает место, где схема зафиксировала перепад яркости, соответствующий положению неподвижного деления, а линия *д* обозначает положение подвижного деления;

- границы зоны, в пределах которой осуществляется поиск делений на изображении в контрольных строках, обозначаются линиями *в*, *е*. За пределами данной зоны схема не выполняет детектирование перепадов яркости. Положение границ определяется шириной *т*, выраженной в количестве пикселей, между опорным (неподвижным) делением и ближайшим делением подвижного барабана в зоне поиска между границами *в* и *е*.



ж- десятые и сотые доли малого деления угломера угла поворота;  
 з- движущееся деление; и- опорная деление

Рисунок 26 - Выходное видеоизображение

Результатом синтеза блока детектирования положения делений лимба на языке *Verilog* является структурная схема (рисунок 27), отражающая его внутреннюю организацию.

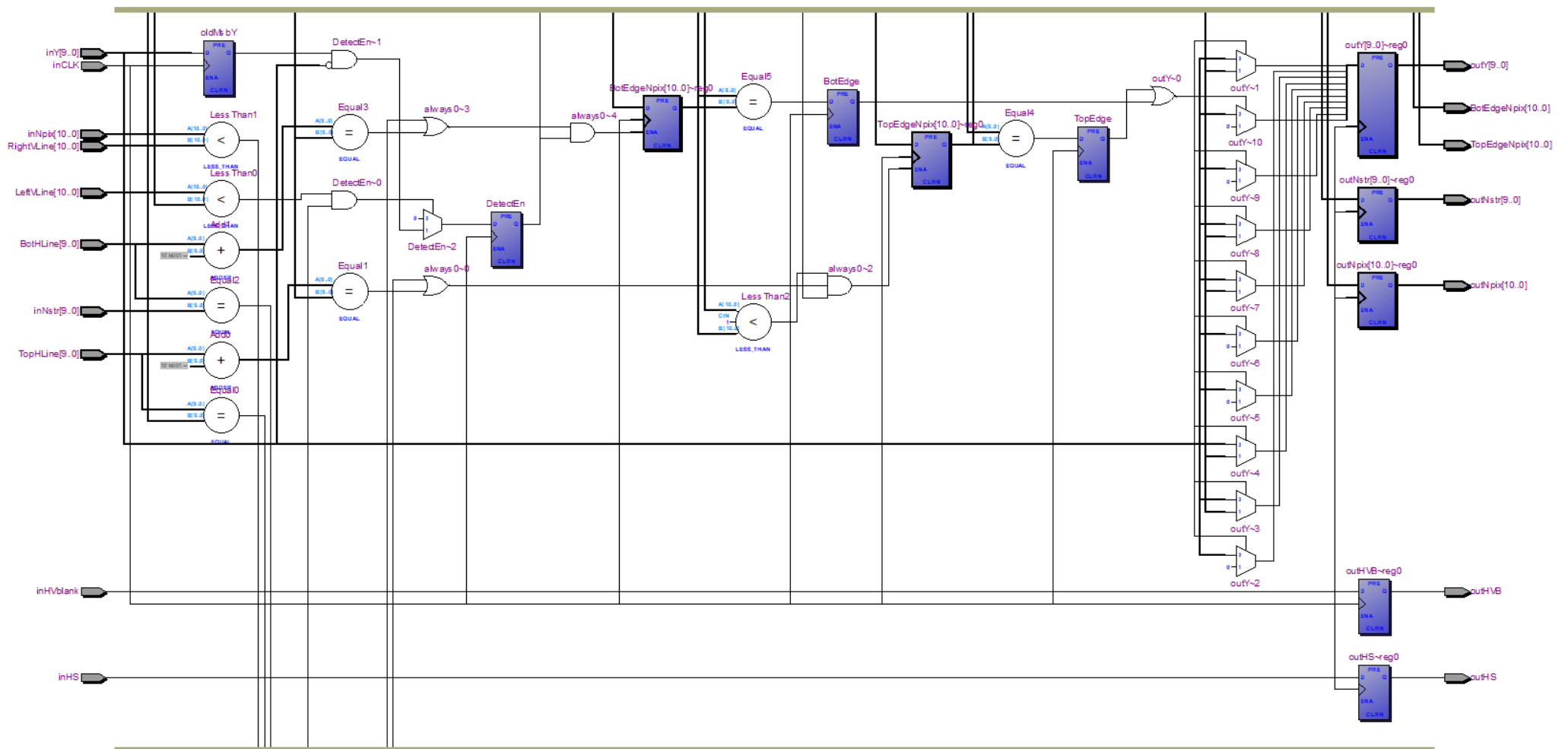


Рисунок 27 – Синтезированная по *Verilog* структурная электрическая схема блока детектирования положения делений лимба на изображении

## 4.5 Вычислитель значений углов

Выходные данные блока детектирования *BotEdgeNpix* и *TopEdgeNpix* поступают на блок, отвечающий за вычисление угла поворота. Схема взаимосвязи блоков представлена на рисунке 28.

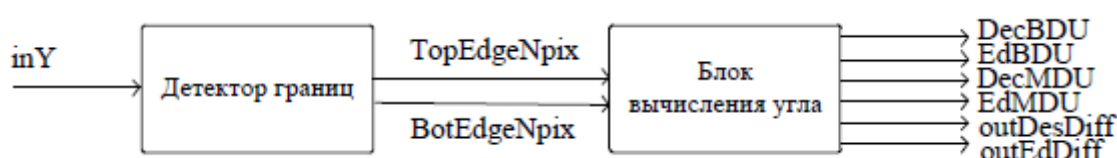


Рисунок 28 - Структурная схема

В рамках разработанной методики для преобразования смещения в пикселях в угловое смещение в единицах шкалы прибора (в данном случае – сотых долях деления угломера) был введен и реализован коэффициент пересчета. С учетом поля зрения видеокамеры использованной в разработанной методике, в одном МДУ размещается количество пикселей  $N_p$  равное 120, для преобразования которых в 100 единиц необходимо умножить значение  $N_p$  на коэффициент пересчета  $k$ , который равен 0,844.

При реализации алгоритма коэффициент пересчета представлен на основе операций сдвига и сложения. Значение коэффициента аппроксимировано следующим разложением в ряд по степеням двойки:

$$k = 2^{-1} + 2^{-2} + 2^{-4} + 2^{-5} = 0,5 + 0,25 + 0,0625 + 0,03125 = 0,84375 \approx 0,844 \quad (20)$$

Данное представление коэффициента позволяет повысить скорость получения результата по сравнению с блоком умножения с плавающей запятой.

Фрагмент описания блока, позволяющий вычислить двоично-десятичный код угла поворота на языке описания электрических схем *Verilog*, как указано в таблице 9.

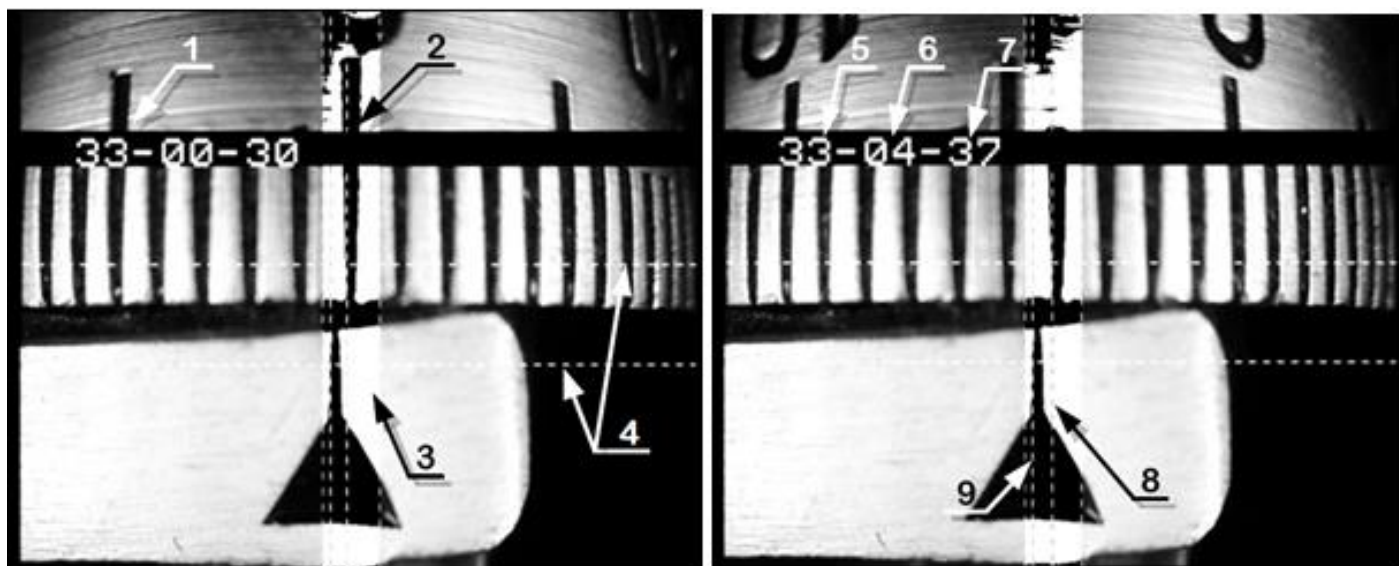
Таблица 9- Фрагмент описания на языке *Verilog* блока вычисления угла

| № | Содержимое строки                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            | Комментарий                                                                                                                                                                                                                                                                                                                                  |
|---|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|   | <pre> module DUcounter(inCLK, inTopEdgeNpix, inBotEdgeNpix, outDesDiff, outEdDiff);  input inCLK;  input [10:0] inTopEdgeNpix;  input [10:0] inBotEdgeNpix;  output reg [3:0] outEdDiff;  output reg [3:0] outDesDiff;  reg [10:0] difEdgeTime;  reg [10:0] oldDifEdgeTime;  reg incDifOldNewEdgesTime;  reg [10:0] DifOldNewEdgesTime; </pre>                                                                                                                                                                               | <p><i>inCLK</i> – входной тактовый сигнал дискретизации видеосигнала;</p> <p><i>inTopEdgeNpix</i>, <i>inBotEdgeNpix</i> – входные двоичные кода найденных позиций яркостных перепадов в верхней и нижней строках соответственно;</p> <p><i>difEdgeTime</i>, <i>oldDifEdgeTime</i>, <i>DifOldNewEdgesTime</i> – вспомогательные триггеры.</p> |
| 8 | <pre> wire [16:0] difEdgeTimeCorrDiv = { difEdgeTime,4'd0} + { difEdgeTime,3'd0} + { difEdgeTime,1'd0} + difEdgeTime;  reg [10:0] difEdgeTimeTimeCorr;  wire [10:0] difEdgeTime = difEdgeTimeTimeCorr - ((outDesDiff == 4'd9) ? 7'd90 : ((outDesDiff == 4'd8) ? 7'd80 : ((outDesDiff == 4'd7) ? 7'd70 : ((outDesDiff == 4'd6) ? 7'd60 : ((outDesDiff == 4'd5) ? 7'd50 : ((outDesDiff == 4'd4) ? 7'd40 : ((outDesDiff == 4'd3) ? 7'd30 : ((outDesDiff == 4'd2) ? 7'd20 : ((outDesDiff == 4'd1) ? 7'd10 : 7'd0)))))))); </pre> | <p>Нормализация расстояния <i>m</i> с использованием дробного коэффициент, преобразующего это значение в сотые доли малых делений угломера.</p>                                                                                                                                                                                              |
|   | <pre> always @ (posedge inCLK) begin </pre>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  | <p>Процедурный блок, описывающий работу схемы, синхронизируемой тактовым сигналом <i>inCLK</i>.</p>                                                                                                                                                                                                                                          |
| 9 | <pre> difEdgeTime &lt;= inTopEdgeNpix - inBotEdgeNpix; </pre>                                                                                                                                                                                                                                                                                                                                                                                                                                                                | <p>В регистр <i>difEdgeTime</i> записывается разность между номером пикселя</p>                                                                                                                                                                                                                                                              |

|    |                                                                                                                             |                                                                                                                                                                                                                                                                                                                                                     |
|----|-----------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|    |                                                                                                                             | <p>верхнего перепада <math>inTopEdgeNpix</math> и номером пикселя нижнего перепада <math>inBotEdgeNpix</math>, которая соответствует расстоянию <math>m</math>.</p>                                                                                                                                                                                 |
| 10 | $oldDifEdgeTime \leq difEdgeTime;$                                                                                          | <p>В регистр <math>oldDifEdgeTime</math> записывается предыдущее значения разности положений рисок в пикселях и фиксируется положение переменной <math>difEdgeTime</math>, которое будет учтено в следующем кадре изображения.</p>                                                                                                                  |
| 11 | $incDifOldNewEdgesTime \leq oldDifEdgeTime < difEdgeTime;$                                                                  | <p>В триггер сохраняются события увеличения расстояния между опорной и движущейся рисками.</p>                                                                                                                                                                                                                                                      |
| 12 | $DifOldNewEdgesTime \leq (oldDifEdgeTime < difEdgeTime) ? (difEdgeTime - oldDifEdgeTime) : (oldDifEdgeTime - difEdgeTime);$ | <p>Регистр <math>difEdgeTime</math> сохраняется изменение расстояния между опорной и перемещающейся риской. Если предыдущее значение меньше нового, то вычисляется разница между новым и предыдущем пикселем, а в противном случае-обратная разность, что позволяет избежать получения отрицательного значения и определить модуль перемещения.</p> |
| 13 | $difEdgeTimeTimeCorr \leq difEdgeTimeCorrDiv [15:5];$                                                                       | <p>В регистр <math>difEdgeTimeTimeCorr</math> записывается значение, поделенное на <math>2^5</math> или 32. В результате достигается минимизация при делении.</p>                                                                                                                                                                                   |

|    |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |                                                                                                                                                                                                                  |
|----|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 14 | <pre> outDesDiff &lt;= (difEdgeTimeTimeCorr &gt; 11'd89) ? 4'd9 :     (((difEdgeTimeTimeCorr &gt; 11'd79) &amp; (difEdgeTimeTimeCorr &lt; 11'd90)) ? 4'd8 :     (((difEdgeTimeTimeCorr &gt; 11'd69) &amp; (difEdgeTimeTimeCorr &lt; 11'd80)) ? 4'd7 :     (((difEdgeTimeTimeCorr &gt; 11'd59) &amp; (difEdgeTimeTimeCorr &lt; 11'd70)) ? 4'd6 :     (((difEdgeTimeTimeCorr &gt; 11'd49) &amp; (difEdgeTimeTimeCorr &lt; 11'd60)) ? 4'd5 :     (((difEdgeTimeTimeCorr &gt; 11'd39) &amp; (difEdgeTimeTimeCorr &lt; 11'd50)) ? 4'd4 :     (((difEdgeTimeTimeCorr &gt; 11'd29) &amp; (difEdgeTimeTimeCorr &lt; 11'd40)) ? 4'd3 :     (((difEdgeTimeTimeCorr &gt; 11'd19) &amp; (difEdgeTimeTimeCorr &lt; 11'd30)) ? 4'd2 :     (((difEdgeTimeTimeCorr &gt; 11'd9) &amp; (difEdgeTimeTimeCorr &lt; 11'd20)) ? 4'd1 : 4'd0)))))); outEdDiff &lt;= difEdgeTime [3:0]; </pre> | <p>Вычисление десятков и единиц перемещения верхних делений относительно нижних, что представляет собой младшую часть двоично-десятичного кода угла поворота (десятые и сотые доли малого деления угломера).</p> |
|    | end                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    | Конец модуля.                                                                                                                                                                                                    |

Отсчёт МДУ в данном блоке производится схемой, фиксирующей событие изменения дробной части МДУ: если дробная часть увеличивалась и происходит переполнение счёта с 99 на 0, то счётчик МДУ инкрементируется, если дробная часть уменьшалась и происходит переполнение счёта с 0 на 99, то счётчик МДУ декрементируется. По такому же принципу строится и счётчик БДУ с отслеживанием изменения значения МДУ. Схема подсчёта МДУ и БДУ в данной работе не представлена ввиду меньшей важности для решения задачи. Выходное видеоизображение приведено на рисунке 29 [30].



а)

б)

а) 1 – отметка лимба 5МДУ; 2 – отметка лимба 0МДУ; 3 – контрольный участок изображения; 4 – строки отслеживания делений лимба;

б) измеренное значение угла: 5 – БДУ; 6 – МДУ; 7 – сотые доли МДУ;  
8 – распознанное положение делений лимба; 9 – распознанное положение опорной риски

Рисунок 29 – Выходное видеоизображение

Синтез цифровых схем, а также полный листинг кода всех описанных модулей представлены в приложениях Ж и З.

## 5 Анализ влияния освещенности научно-исследовательской лаборатории на работу инженера при изготовлении и эксплуатации проектируемого устройства

При разработке, изготовлении и последующей эксплуатации любого технического объекта особое значение приобретает соблюдение требований безопасности жизнедеятельности. Объектом рассмотрения в данном разделе являются условия труда персонала на этапах изготовления (сборки, настройки) и эксплуатации проектируемого устройства.

1. Опасные и вредные производственные факторы. В соответствии с ГОСТ 12.0.003-2015 на инженера, технолога или оператора при изготовлении и эксплуатации проектируемого оборудования возможно воздействие следующих групп факторов:

### 1) Фактические факторы:

- опасность поражения электричеством при работе, наладке и ремонте с устройством, питающимся от сети переменного тока напряжением 220В и частотой 50Гц;
- возможное превышение допустимых уровней звука при работе механических узлов, вентиляционных систем охлаждения или технологического оборудования;
- недостаточная освещенность рабочей зоны;
- несоответствие температуры, влажности или подвижности воздуха в рабочем помещении оптимальным значением;
- электромагнитное излучение;
- риск накопления зарядов на корпусе оборудования или теле оператора.

### 2) Химические факторы

При наличии в технологическом процессе пайки, травления, нанесения покрытий или использование химреактивов.

### 3) Психофизиологические факторы:

- нервно-психические перегрузки;
- зрительное переутомление;
- статические физические нагрузки

2. Эргономика рабочего места. При проектировании рабочего места учитываются требования ГОСТ 12.2.032-78. Основные параметры рабочего места при выполнении работ сидя приведены в таблице 10.

Таблица № 10 – Основные параметры рабочего места ГОСТ 12.2.032-78

| Параметр                  | Требования и нормативные значения                                                  |
|---------------------------|------------------------------------------------------------------------------------|
| Рабочая поверхность стола | Регулируемая по высоте в пределах от 680 до 800 мм                                 |
| Пространство для ног      | Высота-не менее 600 мм, ширина-не менее 500мм                                      |
| Рабочее кресло            | Подъемно-поворотная конструкция с регулировкой высоты сиденья и угла наклона спины |
| Расположение монитора     | Расстояние от глаз оператора до экрана – от 600 до 700 мм                          |

3. Защита от шума и вибрации. Допустимые уровни звука на рабочих местах операторов установлены ГОСТ 12.1.003-2014 и СанПиН 1.2.3685-21 и не должны превышать 50-60 дБа для работ, требующих сосредоточенности и связанных с вычислительной техникой.

Для снижения шумового воздействия предусмотрены следующие меры:

- использование в конструкции устройства малошумных комплектующих;
- применение звукопоглощающих материалов и конструкций с ограждением помещения;
- рациональное размещение оборудования (шумные агрегаты выносятся за пределы рабочей зоны или закрываются кожухами).

4. Пожарная безопасность. Помещение, в котором проводятся работы по взрывопожарной и пожарной безопасности относится к категории В согласно СП 12.12130.2009, так как в нем находятся твердые горючие вещества и материалы (изоляция проводов, мебель, пластик корпусов, бумага).

Причины пожара могут быть связаны с коротким замыканием в электросети, перегрузкой оборудования, нарушением правил эксплуатации.

Для предотвращения пожара и обеспечения безопасности людей в соответствии с ГОСТ 12.1.004-91 и «Правилами противопожарного режима в РФ» предусмотрены:

- организационные мероприятия (проведение противопожарных инструктажей);
- технические мероприятия (системы автоматической пожарной сигнализации и углекислотные огнетушители типа ОУ-5 и ОУ-8 для тушения электроустановок под напряжением)
- конструктивное решение (устройство аварийного отключения питания и применение негорючих кабельных линий).

5. Мероприятия по обеспечению электробезопасности. На этапе изготовления и наладки:

- корпус устройства выполнять из материалов, исключающих возможность случайного прикосновения к токоведущим частям (класс защиты I или II согласно ГОСТу 12.2.007.0-75);
- в конструкции предусмотреть защитное заземление

6. Обеспечение комфортных условий труда (микроклимат и освещение). Параметры микроклимата в помещении должны поддерживаться в пределах, установленных ГОСТ 12.1.005-88 и СанПиН 1.2.3685-21:

- оптимальная температура воздуха 22-24 °С (для холодного периода) и 23-25 °С (для теплого периода)<sup>4</sup>
- относительная влажность соответствует 40-60%;
- скорость движения воздуха – не более 0,1 м/с.

Характер выполняемых работ при эксплуатации проектируемого устройства (работа с дисплеем, считывание показаний приборов, мелкая сборка) требуется создание качественной осветительной среды.

В соответствии со СНиП 23-05-95 для помещений, где ведутся работы средней точности (разряд зрительных работ Б-1 или В-2), нормируемая освещенностью при системе общего освещения должна составлять 300-500 лк.

Для подтверждения соответствия используемой системы освещения СНиП 23-05-95, произведем расчет освещенности рабочего места инженера в

лаборатории точечным методом. Данный метод применяется для расчета местного, общего равномерного и локализованного освещения при любом расположении освещаемых поверхностей.

Нормируемая минимальная освещенность рабочего места определяется уравнением:

$$E = \frac{Fm\sum e p \sum Y}{1000K_3h}, \quad (21)$$

где  $F$  – световой поток светильника с учетом количества входящих в него светодиодных ламп, лм;  $m$  – коэффициент, учитывающий отраженную составляющую света и действие удаленных светильников;  $\sum e$  – условная освещенность, которая создавалась бы  $i$ -м светильником при установке в нем лампы 1000 лк;  $p$  – расстояние от плоскости, в которой расположена контрольная точка, до проекции светильника, м;  $\sum Y$  – коэффициент, учитывающий переход от горизонтальной плоскости к вертикальной;  $K_3$  – коэффициент запаса, учитывающий загрязнение светильников и износ источников света;  $h$  – высота подвеса светильников над расчетной поверхностью, м.

Условная освещенность, которая создавалась бы  $i$ -м светильником при установке в него лампы 1000 лк может быть определена по формуле:

$$e_i = \frac{I_\alpha \cos^3 \alpha_i}{h^2}, \quad (22)$$

где  $I_\alpha$  – сила света по направлению к точке;  $\alpha_i$  – угол между нормалью к поверхности, которой принадлежит точка и лучом падающего света, определяемый выражением:

$$\alpha_i = \arctg\left(\frac{d_i}{h}\right), \quad (23)$$

где  $d_i$  – расстояние от контрольной точки  $A$  до проекции светильника.

Коэффициент, учитывающий переход от горизонтальной плоскости к вертикальной равен:

$$\sum Y = Y_1 + Y_2 + \dots + Y_i \quad (24)$$

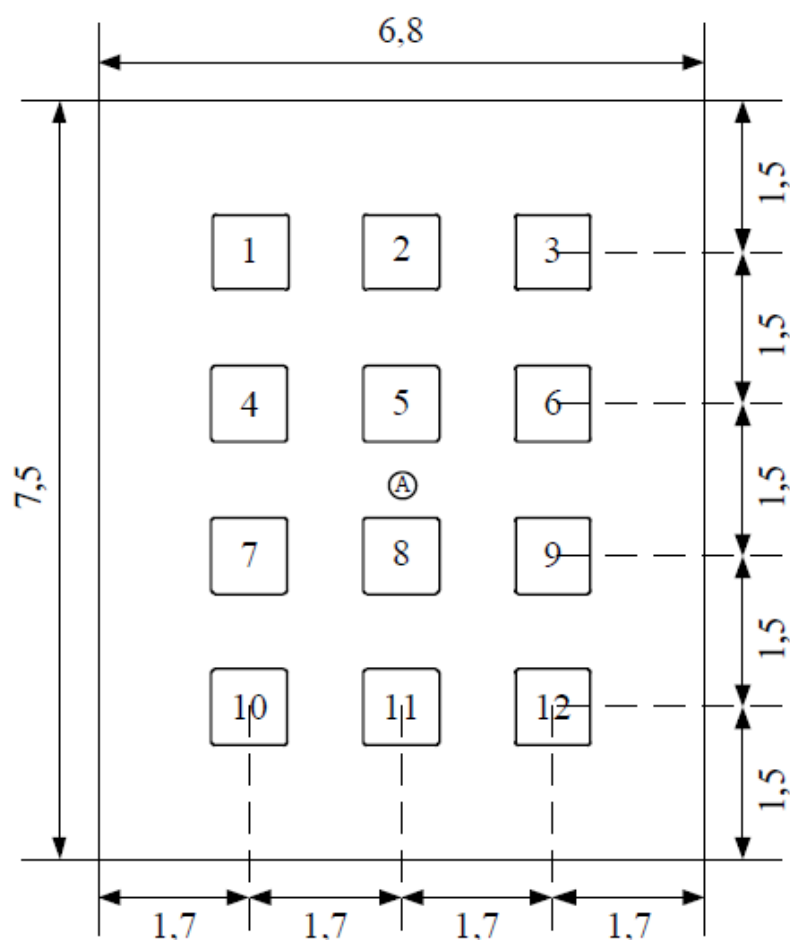
$$Y_i = \cos \beta + \frac{d_i}{h} \sin \beta, \quad (25)$$

где  $\beta$  -угол наклона вертикальной плоскости, равный  $90^\circ$ , таким образом  $\cos\beta = 0$  и  $\sin\beta = 1$ .

Данная лаборатория имеет характеристики, приведенные в таблице 11, расположение светильников в помещении показано на рисунке 30.

Таблица № 11 – Параметры и характеристики лаборатории

| Параметр                                                  | Характеристика                                                            |
|-----------------------------------------------------------|---------------------------------------------------------------------------|
| Тип освещения                                             | Комбинированное: естественно боковое и общее искусственное                |
| Тип светильников                                          | Светодиодные лампы <i>GLT8F</i> , с прямым косинусным светораспределением |
| Световой поток, лк                                        | 860                                                                       |
| Количество источников света, шт                           | 12 светильников, в каждом из которых установлено 4 светодиодные лампы     |
| Габариты лаборатории, м                                   | Длина – 6,8; ширина – 7,5; высота – 3,5                                   |
| Условия эксплуатации                                      | Помещение с малым выведением пыли                                         |
| Значение коэффициентов для светодиодных ламп              | Коэффициент неравномерности $m$ равен 1,1                                 |
|                                                           | Коэффициент запаса $K_z$ равен 1,5                                        |
| Высота подвеса светильников над расчетной поверхностью, м | $h$ равна 3,5                                                             |



1-12 – светильник; А - контрольная точка

Рисунок 30 – Расположение светильников в лаборатории

Световой поток  $F$  одного светильника составляет 3440 лк. Произведем расчет минимальной освещенности на рабочем месте, используя приведенные выше данные и формулы:

$$d_5 = d_8 = \sqrt{0,75^2 + 1^2} = 1,25 \text{ м}, \quad (26)$$

$$d_2 = d_{11} = \sqrt{2,25^2 + 1^2} = 2,46 \text{ м}, \quad (27)$$

$$d_4 = d_6 = d_7 = d_9 = \sqrt{0,75^2 + 1,7^2} = 1,86 \text{ м}, \quad (28)$$

$$d_1 = d_3 = d_{10} = d_{12} = \sqrt{2,25^2 + 1,7^2} = 2,82 \text{ м}, \quad (29)$$

$$\alpha_5 = \alpha_8 = \arctg\left(\frac{1,25}{3,5}\right) = \arctg(0,3571) = 20^\circ, \quad (30)$$

$$\alpha_2 = \alpha_{11} = \arctg\left(\frac{2,46}{3,5}\right) = \arctg(0,7028) = 35^\circ, \quad (31)$$

$$\alpha_4 = \alpha_6 = \alpha_7 = \alpha_9 = \arctg\left(\frac{1,86}{3,5}\right) = \arctg(0,5314) = 28^\circ, \quad (32)$$

$$\alpha_1 = \alpha_3 = \alpha_{10} = \alpha_{12} = \arctg\left(\frac{2,82}{3,5}\right) = \arctg(0,8057) = 39^\circ, \quad (33)$$

$$I_5 = I_8 = 132 \text{ кД}, \quad (34)$$

$$I_2 = I_{11} = 97 \text{ кД}, \quad (35)$$

$$I_4 = I_6 = I_7 = I_9 = 68 \text{ кД}, \quad (36)$$

$$I_1 = I_3 = I_{10} = I_{12} = 84 \text{ кД}, \quad (37)$$

$$e_5 = e_8 = \frac{132 \cdot \cos^3 20^\circ}{3,5^2} = 8,94 \text{ лк}, \quad (38)$$

$$e_2 = e_{11} = \frac{97 \cdot \cos^3 35^\circ}{3,5^2} = 4,35 \text{ лк}, \quad (39)$$

$$e_4 = e_6 = e_7 = e_9 = \frac{68 \cdot \cos^3 28^\circ}{3,5^2} = 3,83 \text{ лк}, \quad (40)$$

$$e_1 = e_3 = e_{10} = e_{12} = \frac{84 \cdot \cos^3 39^\circ}{3,5^2} = 3,21 \text{ лк}, \quad (41)$$

$$\sum e = e_1 + e_2 + \dots + e_i = 2 \cdot (8,94 + 4,35) + 4 \cdot (3,83 + 3,21) = 54,74 \text{ лк} \quad (42)$$

$$Y_5 = Y_8 = 0,37, \quad (43)$$

$$Y_2 = Y_{11} = 0,71, \quad (44)$$

$$Y_4 = Y_6 = Y_7 = Y_9 = 0,53, \quad (45)$$

$$Y_1 = Y_3 = Y_{10} = Y_{12} = 0,81, \quad (46)$$

$$\sum Y = 2 \cdot (0,37 + 0,71) + 4 \cdot (0,53 + 0,81) = 7,52, \quad (47)$$

$$E = \frac{3440 \cdot 1,1 \cdot 54,74 \cdot 1 \cdot 7,52}{1000 \cdot 1,5 \cdot 3,5} = 346,147 \approx 346 \text{ лк} \quad (48)$$

Полученные результаты 346 лк удовлетворяют СНиП 23-05-95ю, таким образом, реализованная в лаборатории система освещения удовлетворяет требованиям безопасности труда и удобства работы.

Помещение, в котором проводились работы, относится к группе помещений без повышенной опасности с точки зрения поражения электрическим током и все приборы питаются от сети переменного тока промышленной частоты 50 Гц, напряжением 220В с надёжно заземленной нейтралью.

Также в соответствии с ГОСТ 12.1.004-91 приняты меры противопожарной безопасности: рубильник прямого доступа для быстрого отключения приборов от сети, огнетушители ОХП-10 и ОХПВ-10.

## ЗАКЛЮЧЕНИЕ

Целью проведенной работы является совершенствование программно-технических средств автоматизации измерения углов.

Для достижения поставленной цели были решены задачи, которые включали в себя анализ современных средств измерения углов, а также методов обработки изображений для распознавания объектов. На основе полученной информации были разработаны:

- элементы методики цифровой обработки видеосигнала для распознавания перемещения и считывания показаний шкал механического угломера;
- структурная схема видеопроцессора средств технического зрения для распознавания перемещения и считывания показаний шкал;
- поведенческие модели функциональных узлов видеопроцессора на языке *Verilog*, синтезированы с помощью современных программных средств их электрические схемы.

Предложенный видеопроцессор, реализующий методики, был испытан на действующем макетном образце, построенном на базе ПЛИС, видеокамере и лимбе от механического угломера. Проведенные испытания подтвердили работоспособность предложенного технического решения и его эффективность при решении задачи автоматического съема показаний. Полученные материалы наглядно отображают структуру и конструкцию спроектированного специализированного видеопроцессора.

Полученные результаты использованы в научно-исследовательских и опытно-конструкторских работах «АО «НПП Рубин». Представленные решения обладают универсальностью и могут быть применены при создании автоматизированных систем контроля и управления с использованием механических средств измерения.

## СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОНИКОВ

1. Что такое угломер, виды и типы угломеров, где и для чего он применяется. – URL: <https://rgk-shop.ru/blog/chto-takoe-uglomer-i-dlya-chego-on-primenyaetsya/?ysclid=mm5wh6cyl5820972888>. (дата обращения: 28.02.2025).
2. Мера углов, принятая в артиллерии. – URL: <https://ppt-online.org/3865389?ysclid=mma8yqh8v2286128981>. (дата обращения: 28.02.2025).
3. Миловзоров О., Грибов Н. Современная технологическая оснастка машиностроительных производств. Учебник для вузов. – ЛитРес, 2025.
4. Кандидов В.П., Чесноков С.С., Шленов С.А. Дискретное преобразование Фурье.-2019.
5. Старовойтов В.В., Голуб Ю.И. Получение и обработка изображений на ЭВМ.-2018.
6. Новиков А. И., Пронькин А. В., Шамин Н. О. Частотный метод фильтрации периодических помех цифровых изображений //Вестник РГРТУ. – 2021. – №. 78. – С. 130.
7. ИГНАТОВ А. К. СПОСОБ И УСТРОЙСТВО ФИЛЬТРОВАНИЯ ШУМА ВИДЕОСИГНАЛОВ. – 2009.
8. Гашников М.В. и др. Методы компьютерной обработки изображений.- 2003.
9. Червяков Н. И., Ляхов П. А., Оразев А. Р. 3D-обобщение метода очистки от импульсного шума для обработки видеоданных //Компьютерная оптика. – 2020. – Т. 44. – №. 1. – С. 92-100.
10. Портнов А. В. и др. Особенности применения алгоритма определения границ объектов на изображениях при использовании градиентных методов //Вестник УрФО. Безопасность в информационной сфере. – 2024. – Т. 4. – №. 54. – С. 5-14.
11. Лях А. М., Рауэн Т. В., Муханов В. С. Методика бинаризации кадров видеосъемки движения микроорганизмов //Системы контроля окружающей среды. – 2023. – №. 2 (52). – С. 116.

12. Ибрафиллов Х. С. Исследование методов бинаризации изображений //Вестник науки и образования. – 2017. – Т. 2. – №. 6 (30). – С. 43-50.
13. Пелевин Е. Е., Балясный С. В. Оптимальные алгоритмы выделения контуров изображения в системе технического зрения //Juvenis scientia. – 2016. – №. 6. – С. 6-8.
14. Талапина Д. М., Новосельская О. А. Модификация алгоритма Хафа для распознавания и классификации штриховых изображений //Труды БГТУ. Серия 3: Физико-математические науки и информатика. – 2024. – №. 2 (284). – С. 89-95.
15. Ханова А. А., Озерова М. И. Обзор методов выделения контуров на изображениях //Информационные технологии в науке и производстве. – 2020. – С. 89-92.
16. Тумаков Д. Н. и др. Основные алгоритмы сегментации изображений, 2024 – 76 с.
17. Горитов А. Н. Предварительная обработка изображений в системах технического зрения //Доклады Томского государственного университета систем управления и радиоэлектроники. – 2018. – Т. 21. – №. 4-1. – С. 53-58.
18. Залесский Б. А., Прадун Д. В. Алгоритм адаптивной фильтрации мультиспектральных изображений //Информатика. – 2018. – Т. 1. – №. 2 (22). – С. 31-38.
19. Немова Н. А., Кудрина М. А., Соловьева Я. В. Автоматизированная система выделения границ на изображении. – 2023.
20. КАРПИК А. П. и др. ЦИФРОВОЙ ВИДЕОНАКЛОНОМЕР ДЛЯ ВЫСОКОТОЧНОГО ОПРЕДЕЛЕНИЯ ЗНАЧЕНИЙ УГЛОВ НАКЛОНА КОНТРОЛИРУЕМОГО ОБЪЕКТА, ИНЖЕНЕРНОГО ИЛИ ПРИРОДНОГО, ПРИ ГЕОДЕЗИЧЕСКОМ МОНИТОРИНГЕ ПОСРЕДСТВОМ ТЕХНОЛОГИИ КОМПЬЮТЕРНОГО ЗРЕНИЯ В РЕЖИМЕ РЕАЛЬНОГО ВРЕМЕНИ В ЭКСТРЕМАЛЬНЫХ КЛИМАТИЧЕСКИХ УСЛОВИЯХ.
21. Пастушков А. В., Калайда В. Т. Методы и алгоритмы поиска объекта на видеопотоке //Сборник научных трудов SWorld. – 2013. – Т. 6. – №. 3. – С. 38-42.

22. ГОСТ 7845-92. Система вращательного телевидения. Основные параметры. Методы измерений. – Москва: ИПК Издательство стандартов, 2001.
23. ГОСТ 53535-2009. Цифровое телевидение высокой четкости. Аналоговое и цифровое представление сигналов. Цифровые интерфейсы. Технические требования. – Москва: Стандартинформ, 2020.
24. ГОСТ р 54457-2011. Цифровая система телевидения высокой четкости. Цифровые интерфейсы. Основные параметры. – Москва: Стандартинформ, 2020.
25. Бакаева М. Р., Вареник Ю. А., Волик А. В. Устройство автоматизации механического угломера // Перспективы развития информационных технологий систем управления: сб. науч. ст. по материалам I науч.-техн. конф. молодых ученых, специалистов и студентов / под общ. ред. М. М. Бутаева. Пенза: Изд-во ПГУ, 2025. С. 80–87.
26. Padmanabhan T. R., Sundari B. B. T. Design Through Verilog HDL. – John Wiley & Sons, 2003.
27. Лушин К. И. Программируемые логические интегральные схемы //Москва. – 2025.
28. Вареник Ю.А., Гордеев И.А. Блок медианного фильтра для построения видеопроцессоров на ПЛИС//Информационные технологии в науке и образовании. Проблемы и перспективы. – 2017.- с.273-274.
29. Вареник Ю.А., Волик А.В., Шмельков А.В. Цифровой фильтр импульсных помех видеосигнала//Информационно-управляющие, телекоммуникационные системы, средства поражения и их техническое обеспечение. – 2019.- с.9-16.
30. Гашников М.В. и др. Методы компьютерной обработки изображений. -2003.- с.175.

## ПРИЛОЖЕНИЕ А

(обязательное)

Задание на магистерскую диссертацию

«Утверждаю»  
Заведующий кафедрой НиМЭ  
Пронин И.А.  
(фамилия, инициалы)  
«21» мая 2026г.

### З А Д А Н И Е на магистерскую диссертацию

1. Студент гр. 24ПДм1 факультета информационных технологий и электроники  
Направление подготовки 11.04.04. «Электроника и микроэлектроника»  
Магистерская программа Нано- и микроэлектроника  
Бакаева Мария Романовна  
(фамилия, имя, отчество)

2. Руководитель Вареник Юрий Александрович  
(фамилия, имя, отчество)

3. Тема выпускной работы Элементы интегральных видеопроцессоров средств технического зрения для автоматизации механических угломеров

Тема утверждена приказом ПГУ № 136% от «05» и 2026г.

4. Техническое задание на магистерскую диссертацию (вид работы: разработка, проектирование, исследование, и т.д.; назначение; основные параметры; условия функционирования и т.д.)

4.1 Методы обработки изображений для автоматизации считывания показаний шкал угломеров;

4.2 Разработать методику цифровой обработки видеосигнала для распознавания перемещения и считывания показаний шкал;

4.3 Разработать структурную схему видеопроцессора средств технического зрения для распознавания перемещения и считывания показаний шкал ;

4.4 Разработать поведенческие модели функциональных узлов видеопроцессора на языке Verilog, синтезировать с помощью современных программных средств их электрические схемы;

5. Объем и содержание основной части магистерской диссертации

5.1. Пояснительная записка (перечень вопросов, подлежащих разработке, обоснований, описаний, расчетов и т.д.)

|                                                                                                                                                    |          |
|----------------------------------------------------------------------------------------------------------------------------------------------------|----------|
| 1 Методы обработки изображений для автоматизации считывания показаний шкал угломеров                                                               |          |
| 1.1 Современные механические и оптико-электронные средства измерения углов                                                                         |          |
| 1.2 Методы обработки изображений для распознавания объектов                                                                                        |          |
| 2 Разработка методики цифровой обработки видеосигнала для распознавания перемещения и считывания показаний шкал                                    |          |
| 2.1 Элементы методики цифровой обработки видеосигнала для распознавания перемещения и считывания показаний шкал                                    |          |
| 2.2 Нормализация и фильтрация видеоизображения шкал угломера                                                                                       |          |
| 2.3 Детектирование границ и определение положения на видеоизображении делений шкал угломера                                                        |          |
| 2.4 Вычисление значений углов по положению на видеоизображении делений шкал угломера                                                               |          |
| 3 Разработка структурной схемы видеопроцессора средств технического зрения для распознавания перемещения и считывания показаний шкал               |          |
| 3.1 Структура телевизионной приставки для механического угломера                                                                                   |          |
| 3.2 Структура видеопроцессора средств технического зрения для распознавания перемещения и считывания показаний шкал                                |          |
| 4 Разработка поведенческих моделей и синтез электрических схем функциональных узлов видеопроцессора                                                |          |
| 4.1 Усредняющий фильтр видеоизображения                                                                                                            |          |
| 4.2 Медианный фильтр видеоизображения                                                                                                              |          |
| 4.3 Бинаризатор видеоизображения                                                                                                                   |          |
| 4.4 Детектор границ делений шкалы                                                                                                                  |          |
| 4.5 Вычислитель значений углов                                                                                                                     |          |
| 5.2. Иллюстративная часть (перечень и содержание чертежей и слайдов презентации)                                                                   |          |
| 1. Актуальность и практическая значимость работы                                                                                                   | Слайд 1  |
| 2. Цели и задачи работы                                                                                                                            | Слайд 2  |
| 3. Элементы методики цифровой обработки видеосигнала                                                                                               | Слайд 3  |
| 4. Структура видеопроцессора средств технического зрения для распознавания перемещения и считывания показаний шкал                                 | Слайд 4  |
| 5. Структура телевизионной приставки для механического угломера                                                                                    | Слайд 5  |
| 6. Усредняющий фильтр видеоизображения                                                                                                             | Слайд 6  |
| 7. Медианный фильтр видеоизображения                                                                                                               | Слайд 7  |
| 8. Бинаризатор видеоизображения                                                                                                                    | Слайд 8  |
| 9. Детектор границ делений шкалы                                                                                                                   | Слайд 9  |
| 10. Вычислитель значений углов                                                                                                                     | Слайд 10 |
| 11. Кадры видеоизображения дополненного измерительной информацией на выходе видеопроцессора                                                        | Слайд 11 |
| 12. Результат работы                                                                                                                               | Слайд 12 |
| 6. Консультанты и содержание дополнительных разделов                                                                                               |          |
| 6.1. Анализ влияния освещенности научно-исследовательской лаборатории на работу инженера при изготовлении и эксплуатации проектируемого устройства |          |
| Консультант: _____                                                                                                                                 |          |

7. Календарный график по выполнению магистерской диссертации

| Наименование этапов работы                                                                                                                    | Объем работы | Срок выполнения | Подпись руководителя, консультанта |
|-----------------------------------------------------------------------------------------------------------------------------------------------|--------------|-----------------|------------------------------------|
| Методы обработки изображений для автоматизации считывания показаний шкал угломеров                                                            | 15%          | 23.05           |                                    |
| Разработка методики цифровой обработки видеосигнала для распознавания перемещения и считывания показаний шкал                                 | 25%          | 25.05           |                                    |
| Разработка структурной схемы видеопроцессора средств технического зрения для распознавания перемещения и считывания показаний шкал            | 20%          | 27.05           |                                    |
| Разработка поведенческих моделей и синтез электрических схем функциональных узлов видеопроцессора                                             | 20%          | 29.05           |                                    |
| Анализ влияния освещенности научно-исследовательской лаборатории на работу инженера при изготовлении и эксплуатации проектируемого устройства | 10%          | 01.06           |                                    |
| Оформление ВКР                                                                                                                                | 10%          | 02.06           |                                    |

Руководитель магистерской диссертации   
подпись

Вареник Ю.А.  
фамилия, инициалы

Дата выдачи задания

«21» мая 2026 г.

Задание к исполнению принял

подпись

Бакаева М.Р.  
фамилия, инициалы

«21» мая 2026 г.

Магистерскую диссертацию к защите допустить

Декан факультета

подпись

Кревчик В.Д.  
фамилия, инициалы

«03» 06 2026 г.

## ПРИЛОЖЕНИЕ Б

(обязательное)

Форма отчета о поиске патентов

Таблица Б.1 – Патентная документация

| Предмет поиска                                            | Страна выдачи, вид и номер охранного документа, МПК | Заявитель, страна. Дата приоритета, дата публикации.                                                                                                                                                     | Название изобретения                                                                                        |
|-----------------------------------------------------------|-----------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------|
| Механические и оптико-электронные средства измерения угла | РФ, патент,<br>RU 2 655 024 C2                      | Федеральное государственное бюджетное учреждение «Научно-исследовательский испытательный центр подготовки космонавтов имени Ю.А. Гагарина»<br>Заявка: 2016130449, 26.07.2016<br>Опубликовано: 31.01.2018 | Способ и устройство измерения угла наклона                                                                  |
| Цифровая обработка видеосигнала                           | РФ, патент,<br>RU 2 769 371 C1                      | ЭлДжи Электроникс Инк.<br>Заявка: 2021122470, 02.01.2020<br>Дата регистрации: 30.03.2022                                                                                                                 | Способ и устройство для обработки видеосигнала посредством использования внешнего прогнозирования           |
| Интегральный видеопроцессор                               | РФ, патент,<br>RU 2 760 365 C2                      | Общество с ограниченной ответственностью «НТИ»<br>Заявка: 2020133913, 14.10.2020<br>Опубликовано: 24.11.2021                                                                                             | Интегрированная судовая система                                                                             |
| Цифровая обработка видеосигнала                           | РФ, патент,<br>RU 2 827 495 C1                      | Федеральное государственное бюджетное образовательное учреждение высшего образования «Юго-Западный государственный университет»<br>Заявка: 2023125300, 03.10.2023<br>Опубликовано: 27.09.2024            | Устройство для выделения периодических последовательностей цифровых сигналов                                |
| Интегральные видеопроцессоры                              | РФ, патент,<br>RU 2 766 053 C1                      | Козлов А.И.<br>Заявка: 2021109800, 08.04.2021<br>Опубликовано: 07.02.2022                                                                                                                                | Тепловизор на основе гибридного фотоприемника с оптическим выходом                                          |
| Средства технического зрения                              | РФ, патент,<br>RU 2 779 021 C1                      | АО «УКБТМ»<br>Заявка: 2021136701, 13.12.2021<br>Опубликовано: 30.08.2022                                                                                                                                 | Способ настройки точности передачи углов и перекрестной связи от прицела с техническим зрением к вооружению |
| Способ обработки видеосигнала                             | РФ, патент,<br>RU 2 647 664 C1                      | Севостьянов П.В.<br>Общество с ограниченной ответственностью «Заботливый город»<br>Заявка: 2017110817, 31.03.2017<br>Опубликовано: 16.03.2018                                                            | Способ обработки видеосигнала                                                                               |

## ПРИЛОЖЕНИЕ В

(обязательное)

Опубликованные статьи за 2024/2026 учебные года

Таблица В.1 - Опубликованные статьи за 2024/2026 года

| №<br>п/п | Наименование<br>работы и ее вид                                                                                                                                | Форма<br>работы | Выходные данные                                                                                                                                                         | Объем<br>в с. | Соавторы                                                                         |
|----------|----------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------|----------------------------------------------------------------------------------|
| 1        | <b>Устройство<br/>автоматизации<br/>механического<br/>угломера (статья)</b>                                                                                    | печатный        | <b>Перспективы<br/>развития<br/>информационных<br/>технологий систем<br/>управления. – Пенза,<br/>Изд-во ПГУ, 2025.<br/>- 8с.</b>                                       | 8/4           | <b>Вареник Ю.А.<br/>Волик А.В.</b>                                               |
| 2        | <b>Опыт разработки<br/>технологического<br/>терминала на<br/>основе<br/>отечественной<br/>ЭКБ для<br/>настройки<br/>сложного<br/>оборудования<br/>(статья)</b> | печатный        | <b>Актуальные<br/>проблемы и<br/>перспективы<br/>развития<br/>радиотехнических и<br/>инфокоммуникацион<br/>ных систем. – Москва,<br/>Радиоинфоком, 2025.<br/>– 6 с.</b> | 6/3           | <b>Волик А.В.<br/>Кожухов Е.В.<br/>Степаненко О.В.</b>                           |
| 3        | Исследование<br>конструктивных<br>особенностей и<br>повышение<br>характеристик<br>тонкопленочных<br>резистивных<br>элементов<br>(статья)                       | печатный        | Методы, средства и<br>технологии получения<br>и обработки<br>измерительной<br>информации. – Пенза,<br>изд-во ПГУ, 2026. – 4 с.                                          | 4/1           | Горбунцов А.А.<br>Антонов В.А.<br>Агафонов Д.В.<br>Печерская Е.А.<br>Левцов К.В. |
| 4        | <b>Цифровые блоки<br/>видеопроцессора<br/>автоматизации<br/>считывания<br/>показаний<br/>лимбов<br/>механических<br/>угломеров<br/>(статья)</b>                | печатный        | <b>Перспективы<br/>развития<br/>информационных<br/>технологий систем<br/>управления. – Пенза,<br/>изд-во ПГУ, 2026. – 4<br/>с.</b>                                      | 4/2           | <b>Вареник Ю.А.<br/>Рязяпов И.Н.</b>                                             |

## ПРИЛОЖЕНИЕ Г

(обязательное)

Заявление и протокол о проверке на антиплагиат

И.о. ректора Пензенского  
государственного университета  
С.М. Васину  
студента 2 курса  
магистратуры ФИТЭ  
очной формы обучения  
Бакаевой Марии  
Романовны

### ЗАЯВЛЕНИЕ.

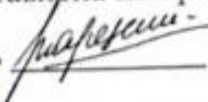
Я, Бакаева Мария Романовна, студент 2 курса магистратуры очной формы обучения, направления подготовки 11.04.04 «Электроника и наноэлектроника», магистерская программа «Нано- и микроэлектроника», прошу провести проверку с использованием системы «Антиплагиат» магистерской диссертации на тему «Элементы интегральных видеопроцессоров средств технического зрения для автоматизации механических угломеров», выполненной мной самостоятельно, на содержание элементов плагиата.

Все прямые заимствования из печатных и электронных источников, а также из защищенных ранее выпускных квалификационных работ, кандидатских и докторских диссертаций имеют соответствующие ссылки.

Я ознакомлен с действующим Стандартом университета «Выпускная квалификационная работа обучающихся по образовательным программам высшего образования – программ бакалавриата, программ специалитета и программ магистратуры», согласно которому обнаружение плагиата является основанием для недопуска ВКР к защите и отчисления из университета.

 Бакаева М.Р.  
« 27 » 05 2026г.

Работа представлена на проверку « 27 » 05 2026г.

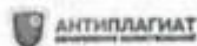
Руководитель  к.т.н., доцент кафедры «НиМЭ» Вареник Ю.А.

## Протокол

проверки на оригинальность в системе «Антиплагиат.ВУЗ» выпускной магистерской работы на тему «Элементы интегральных видеопроцессоров средств технического зрения для автоматизации механических угломеров» студента 2 курса Бакаевой Марии Романовны

направления подготовки 11.04.04 Электроника и микроэлектроника, профиль «Нано- и микроэлектроника»

Руководитель ВКР Вареник Юрий Александрович



### СПРАВКА

о результатах проверки текстового документа на наличие заимствований

Федеральное государственное бюджетное образовательное учреждение высшего образования «Томский государственный университет»

ПРОВЕРКА ВЫПОЛНЕНА В СИСТЕМЕ ANTIPLAGIAT.VUZ

Автор работы: Бакаева Мария Романовна  
Сектор проверки: Бакаева Мария Романовна  
реквизиты для: Бакаева Мария Романовна  
Название работы: Элементы интегральных видеопроцессоров средств технического зрения для автоматизации механических угломеров  
Тип работы: Магистерская диссертация  
Подразделение: Кафедра "Физики и микроэлектроники"

### РЕЗУЛЬТАТЫ

■ отчет о проверке опубликованном виде представлен результаты проверки документами

|                 |      |                 |      |
|-----------------|------|-----------------|------|
| СООПЛАДЕНИЕ     | 100% | СООПЛАДЕНИЕ     | 100% |
| ОРИГИНАЛЬНОСТЬ  | 100% | ОРИГИНАЛЬНОСТЬ  | 100% |
| ЦИТИРОВАНИЕ     | 0%   | ЦИТИРОВАНИЕ     | 0%   |
| САМОЦИТИРОВАНИЕ | 100% | САМОЦИТИРОВАНИЕ | 100% |
| ИНТЕРНЕТ        | 0%   |                 |      |

доля в общей копированности 0,00 0,00 100%

доля копированности 0,00 0,00 100%

### Структура документа:

Проверены разделы: основная часть 4.2-4, 14-17, введение 4.3-1.3, введение 4.76

### Методы поиска:

Полнотекстовый поиск по базам данных: Профессиональная литература, Медицина, Интернет; Профессиональная литература, Структурная профессиональная литература, АПК и биотех, РИМед, ИБЕ, Шаблонная форма; Секция коллекция научных работ Беларуси, Патенты СССР, РФ, СНГ; Публикации РГБ, ИС Адмет; Парфразирование по базе публикаций открытого доступа PubMed, OAI России и СНГ; Коллекция открытых публикаций международного масштаба в СРС ГАР-АИТ; аннотация; Перекладывание заимствования по коллекции Гарант; аннотация; Парфразирование по Коллекции открытых публикаций международного масштаба в Публикации РГБ (параллельно и параллельно); Парфразирование заимствования по коллекции Интернет в аннотации сегмента; Парфразирование по коллекции ИБЕ; Медицина; Публикации eLibrary; П...

Результат проверки: Голубовский Алексей Николаевич

ИДН: 0000000000

Дата подписания:

Подпись: 0000000000



Чтобы убедиться в подлинности документа, сканируйте QR-код, который находится на обратной стороне документа.

Если на экране монитора отображается сообщение об ошибке, значит документ не является оригиналом. Подлинность документа не подтверждается.

Руководитель ВКР

« 27 » 05 2026 год

*Вареник Ю.А.*  
подпись

Вареник Ю.А.  
фамилия, инициалы

## СПРАВКА

о результатах проверки текстового документа  
на наличие заимствований

Федеральное государственное бюджетное  
образовательное учреждение высшего  
образования "Пензенский государственный  
университет"

ПРОВЕРКА ВЫПОЛНЕНА В СИСТЕМЕ АНТИПЛАГИАТ.ВУЗ

Автор работы: Бакаева Мария Романовна  
Самоцитирование  
рассчитано для: Бакаева Мария Романовна  
Название работы: Элементы интегральных видеопроцессоров средств технического зрения для автоматизации  
механических угломеров  
Тип работы: Магистерская диссертация  
Подразделение: Кафедра "Нано- и микроэлектроника"

### РЕЗУЛЬТАТЫ

■ ОТЧЕТ О ПРОВЕРКЕ КОРРЕКТИРОВАЛСЯ: НИЖЕ ПРЕДСТАВЛЕНЫ РЕЗУЛЬТАТЫ ПРОВЕРКИ ДО КОРРЕКТИРОВКИ

|                 |        |                 |        |
|-----------------|--------|-----------------|--------|
| СОВПАДЕНИЯ      | 17.15% | СОВПАДЕНИЯ      | 17.15% |
| ОРИГИНАЛЬНОСТЬ  | 71.11% | ОРИГИНАЛЬНОСТЬ  | 71.11% |
| ЦИТИРОВАНИЯ     | 0%     | ЦИТИРОВАНИЯ     | 0%     |
| САМОЦИТИРОВАНИЯ | 11.74% | САМОЦИТИРОВАНИЯ | 11.74% |
| ИИ-КОНТЕНТ      | 0%     |                 |        |

ДАТА И ВРЕМЯ КОРРЕКТИРОВКИ: 15.06.2026 14:55

ДАТА ПОСЛЕДНЕЙ ПРОВЕРКИ: 15.06.2026

Структура  
документа:  
Модули поиска:

Проверенные разделы: основная часть с.2-4, 14-77, введение с.5-13, выводы с.76

Патенты СССР, РФ, СНГ; Профессиональная лексика. АПК и биотех; Рувики; Переводные заимствования; Профессиональная лексика. Медицина; Профессиональная лексика. Юриспруденция; Шаблоны фразы; PubMed; Цитирование; ИПС Адилет; IEEE; СМИ России и СНГ; Коллекция открытых публикаций международных издательства; Коллекция НБУ; Кольцо вузов; Публикации eLIBRARY; Сводная коллекция научных работ Беларуси; Медицина; Публикации РГБ; Перефразирования по базе публикаций открытого доступа PubMed; Публикации eLIBRARY (переводы и перефразирования); Сводная коллекция ЭБС; Перефразирования по коллекции IEEE; СПС ГАРАНТ; нормативно-правовая документация; СПС ГАРАНТ; аналитика; Перефразирования по Коллекции открытых публикаций международных издательства; Публикации РГБ (переводы...

Работу проверил: Головяшкин Алексей Николаевич

ФИО проверяющего

Дата подписи: 15.06.2026

Подпись проверяющего



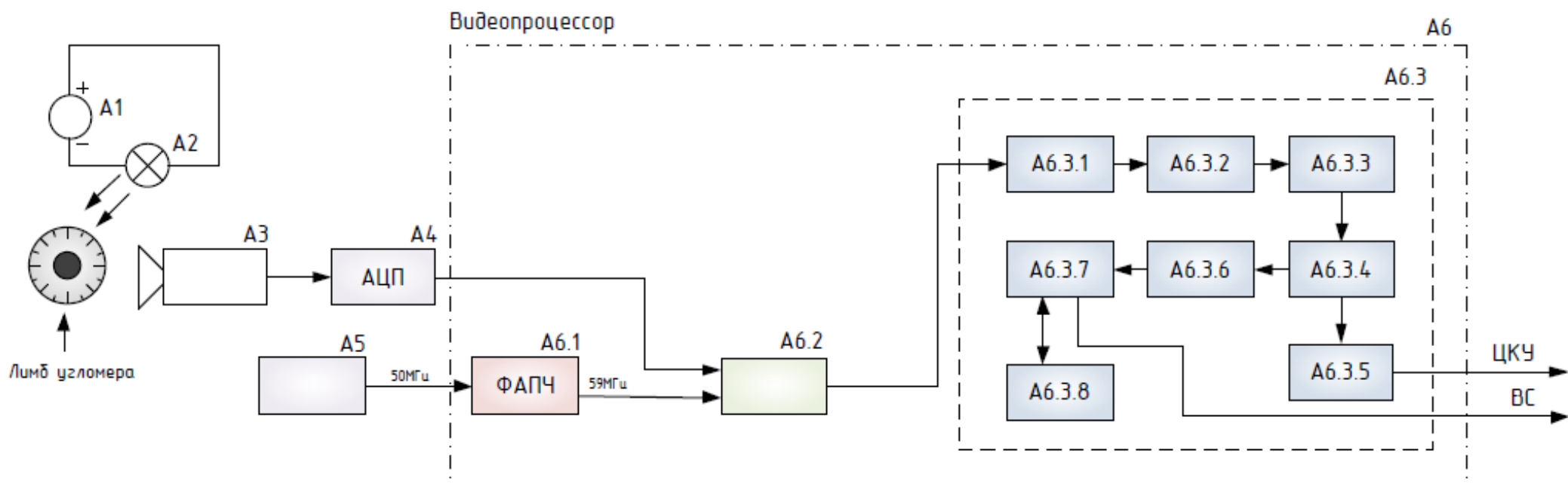
Чтобы убедиться  
в подлинности справки, используйте QR-код,  
который содержит ссылку на отчет.

Ответ на вопрос, является ли обнаруженное заимствование  
корректным, система оставляет на усмотрение проверяющего.  
Предоставленная информация не подлежит использованию  
в коммерческих целях.

## ПРИЛОЖЕНИЕ Е

(обязательное)

Структурная схема телевизионной приставки для механического угломера



## ПРИЛОЖЕНИЕ Ж

(обязательное)

Поведенческие модели функциональных узлов видеопроцессора на языке  
*Verilog*

## Усредняющий фильтр видеоизображения

```
module AvrFLT ( inHVblank, inSSC, inVData, inNpix, inNstr,inCLK, inFltOn,  
               outHVblank, outSSC, outVData, outNpix, outNstr );  
  
input inHVblank;  
input inSSC;  
input [9:0] inVData;  
input [10:0] inNpix;  
input [9:0] inNstr;  
input inCLK;  
input inFltOn;  
  
output reg outHVblank;  
output reg outSSC;  
output reg [9:0] outVData;  
output reg [10:0] outNpix;  
output reg [9:0] outNstr;  
  
reg [9:0] Stream [0:4];  
reg [22:0] SHV [0:2];  
  
wire [11:0] AVR = Stream [1] [9:0] + Stream [3] [9:0] + Stream [0] [9:0] + Stream  
[2] [9:0];  
always @( posedge inCLK )  
begin  
    Stream [0] [9:0] <= inVData;  
    Stream [1] [9:0] <= Stream [0] [9:0];  
    Stream [2] [9:0] <= Stream [1] [9:0];
```

```

Stream [3] [9:0] <= Stream [2] [9:0];

outVData <= ( inFltOn ) ? AVR [11:2] : Stream [2] [9:0];
SHV [0] [22:0] <= { inNstr, inNpix, inSSC, inHVblank };
SHV [1] [22:0] <= SHV [0] [22:0];
SHV [2] [22:0] <= SHV [1] [22:0];
outHVblank <= SHV [2] [0];
outSSC <= SHV [2] [1];
outNpix <= SHV [2] [12:2];
outNstr <= SHV [2] [22:13];

end
endmodule

```

## Медианный фильтр видеоизображения

```
module MedianFLT ( inHVblank, inSSC, inVData, inNpix, inNstr, inCLK, inFltOn,  
                  outHVblank, outSSC, outVData, outNpix, outNstr );
```

```
input inHVblank;
```

```
input inSSC;
```

```
input [9:0] inVData;
```

```
input [10:0] inNpix;
```

```
input [9:0] inNstr;
```

```
input inCLK;
```

```
input inFltOn;
```

```
output reg outHVblank;
```

```
output reg outSSC;
```

```
output reg [9:0] outVData;
```

```
output reg [10:0] outNpix;
```

```
output reg [9:0] outNstr;
```

```
reg [9:0] Stream [0:4];
```

```
wire [2:0] Summ [0:4];
```

```
assign Summ [0] [2:0] = ( Stream [1] [9:0] < Stream [0] [9:0] ) +  
                        ( Stream [2] [9:0] < Stream [0] [9:0] ) +  
                        ( Stream [3] [9:0] < Stream [0] [9:0] ) +  
                        ( Stream [4] [9:0] < Stream [0] [9:0] );
```

```
assign Summ [1] [2:0] = ( Stream [0] [9:0] < Stream [1] [9:0] ) +  
                        ( Stream [2] [9:0] < Stream [1] [9:0] ) +  
                        ( Stream [3] [9:0] < Stream [1] [9:0] ) +
```

```

        ( Stream [4] [9:0] < Stream [1] [9:0] );

assign Summ [2] [2:0] = ( Stream [0] [9:0] < Stream [2] [9:0] ) +
        ( Stream [1] [9:0] < Stream [2] [9:0] ) +
        ( Stream [3] [9:0] < Stream [2] [9:0] ) +
        ( Stream [4] [9:0] < Stream [2] [9:0] );

assign Summ [3] [2:0] = ( Stream [0] [9:0] < Stream [3] [9:0] ) +
        ( Stream [1] [9:0] < Stream [3] [9:0] ) +
        ( Stream [2] [9:0] < Stream [3] [9:0] ) +
        ( Stream [4] [9:0] < Stream [3] [9:0] );

assign Summ [4] [2:0] = ( Stream [0] [9:0] < Stream [4] [9:0] ) +
        ( Stream [1] [9:0] < Stream [4] [9:0] ) +
        ( Stream [2] [9:0] < Stream [4] [9:0] ) +
        ( Stream [3] [9:0] < Stream [4] [9:0] );

wire [2:0] Num0x = ( Summ [0] == 3'd0 ) +
        ( Summ [1] == 3'd0 ) +
        ( Summ [2] == 3'd0 ) +
        ( Summ [3] == 3'd0 ) +
        ( Summ [4] == 3'd0 );

wire [2:0] Num1x = ( Summ [0] == 3'd1 ) +
        ( Summ [1] == 3'd1 ) +
        ( Summ [2] == 3'd1 ) +
        ( Summ [3] == 3'd1 ) +
        ( Summ [4] == 3'd1 );

```

```

wire [9:0] Mediana = ( Summ [0] == 3'd2 ) ? Stream [0] [9:0] :
    (( Summ [1] == 3'd2 ) ? Stream [1] [9:0] :
    (( Summ [2] == 3'd2 ) ? Stream [2] [9:0] :
    (( Summ [3] == 3'd2 ) ? Stream [3] [9:0] :
    (( Summ [4] == 3'd2 ) ? Stream [4] [9:0] :
    ((Num0x > 3'd2 ) ? (
        ( Summ [0] == 3'd0 ) ? Stream [0] [9:0] :
        (( Summ [1] == 3'd0 ) ? Stream [1] [9:0] : Stream [0] [9:0])
        ) :
    (( Num1x > 3'd1 ) ? (
        ( Summ [0] == 3'd1 ) ? Stream [0] [9:0] :
        (( Summ [1] == 3'd1 ) ? Stream [1] [9:0] :
        (( Summ [2] == 3'd1 ) ? Stream [2] [9:0] : Stream [3] [9:0]))
        ) : Stream [2] [9:0])]]]]));

wire [9:0] Difference12x = ( Stream [2] [9:0] > Stream [1] [9:0]) ? (Stream [2] [9:0] -
Stream [1] [9:0]) :
    ( Stream [1] [9:0] - Stream [2] [9:0] );

wire [9:0] Difference23x = ( Stream [2] [9:0] > Stream [3] [9:0] ) ? ( Stream [2] [9:0]
- Stream [3] [9:0] ) :
    ( Stream [3] [9:0] - Stream [2] [9:0] );

reg [22:0] SHV [0:2];
always @( negedge inCLK )
begin
    Stream [0] [9:0] <= inVData;
    Stream [1] [9:0] <= Stream [0] [9:0];
    Stream [2] [9:0] <= Stream [1] [9:0];
    Stream [3] [9:0] <= Stream [2] [9:0];
    Stream [4] [9:0] <= Stream [3] [9:0];

```

```

outVData <= (inFltOn) ? Mediana : Stream [2] [9:0];

SHV [0] [22:0] <= { inNstr, inNpix, inSSC, inHVblank };
SHV [1] [22:0] <= SHV [0] [22:0];
SHV [2] [22:0] <= SHV [1] [22:0];

outHVblank <= SHV [2] [0];
outSSC <= SHV [2] [1];
outNpix <= SHV [2] [12:2];
outNstr <= SHV [2] [22:13];

end
endmodule

```

## Бинаризатор видеоизображения

```
module Binarizator( inHVblank, inHS, inY, inNpix, inNstr, inCLK,
                   outHVB, outHS, outY, outNpix, outNstr,
                   LeftVLine, RightVLine, TopHLine, BotHLine );

Input inHVblank;
Input inHS;
input [9:0] inY;
input [10:0] inNpix;
input [9:0] inNstr;
input inCLK;

output reg outHVB;
output reg outHS;
output reg [9:0] outY;
output reg [10:0] outNpix;
output reg [9:0] outNstr;
output [10:0] LeftVLine = 11'd692;
output [10:0] RightVLine = 11'd820;
output [9:0] TopHLine = 10'd260;
output [9:0] BotHLine = 10'd362;

wire [9:0] CenterLine = ( TopHLine + BotHLine ) >> 1;

parameter YavrTop = 10'd150;
parameter YavrBot = 10'd170;

wire TopBar = ( inNpix > LeftVLine ) & ( inNpix < RightVLine ) & ( inNstr <
CenterLine );
wire BotBar = ( inNpix > LeftVLine ) & ( inNpix < RightVLine ) & ( inNstr >
CenterLine );

wire [9:0] TopY = ( inY > YavrTop ) ? 10'd716 : 10'd0;
wire [9:0] BotY = ( inY > YavrBot ) ? 10'd716 : 10'd0;
```

```
always @( posedge inCLK )
begin

outY <= TopBar ? TopY : ( BotBar ? BotY : inY );

outHVB <= inHVblank;
outHS <= inHS;
outNpix <= inNpix;
outNstr <= inNstr;

end
endmodule
```

## Детектор границ делений шкалы

```
module EdgeDetector( inHVblank, inHS, inY, inNpix, inNstr, inCLK,  
                    LeftVLine, RightVLine, TopHLine, BotHLine,  
                    outHVB, outHS, outY, outNpix, outNstr,  
                    TopEdgeNpix, BotEdgeNpix );
```

```
input inHVblank;
```

```
input inHS;
```

```
input [9:0] inY;
```

```
input [10:0] inNpix;
```

```
input [9:0] inNstr;
```

```
input inCLK;
```

```
input [10:0] LeftVLine;
```

```
input [10:0] RightVLine;
```

```
input [9:0] TopHLine;
```

```
input [9:0] BotHLine;
```

```
output reg outHVB;
```

```
output reg outHS;
```

```
output reg [9:0] outY;
```

```
output reg [10:0] outNpix;
```

```
output reg [9:0] outNstr;
```

```
output reg [10:0] TopEdgeNpix;
```

```
output reg [10:0] BotEdgeNpix;
```

```
reg DetectEn;
```

```
reg oldMsbY;
```

```
reg TopEdge;
```

```

reg BotEdge;

always @( posedge inCLK )
begin
oldMsbY <= inY[9];

DetectEn <= (( inNpix > LeftVLine ) & ( inNpix < RightVLine )) ? ( oldMsbY & ~
inY[9]) : 1'b0;

if ( DetectEn & (( inNstr == TopHLine ) | ( inNstr == TopHLine + 1'b1 )) & ( inNpix
>= BotEdgeNpix )) TopEdgeNpix <= inNpix;

if ( DetectEn & (( inNstr == BotHLine ) | ( inNstr == BotHLine + 1'b1 ))) BotEdgeNpix
<= inNpix;

TopEdge <= ( TopEdgeNpix[10:1] == inNpix[10:1] );

BotEdge <= ( BotEdgeNpix[10:1] == inNpix[10:1] );


outY <= ( TopEdge | BotEdge ) ? ( inNstr[3] ? 10'd716 : 10'd0) : inY;
outHVB <= inHVblank;
outHS <= inHS;
outNpix <= inNpix;
outNstr <= inNstr;


end

endmodule

```

## Вычислитель значений углов

```
module DUcounter( inCLK, inTopEdgeNpix, inBotEdgeNpix, inNresetDU,
                  outMDU, outDecBDU, outEdinBDU, outDecMDU, outEdMDU,
                  outDesDiff, outEdDiff );

input inCLK;
input [10:0] inTopEdgeNpix;
input [10:0] inBotEdgeNpix;
input inNresetDU;

output reg [12:0] outMDU;
output reg [3:0] outDecBDU;
output reg [3:0] outEdinBDU;
output reg [3:0] outDecMDU;
output reg [3:0] outEdMDU;
output reg [3:0] outEdDiff;
output reg [3:0] outDesDiff;

reg [10:0] difEdgeTime;
reg [10:0] oldDifEdgeTime;
reg [10:0] ModuleDifOldNewEdgesTime;
reg incDifOldNewEdgesTime;
reg [10:0] DifOldNewEdgesTime;

wire [16:0] difEdgeTimeCorrDiv = { difEdgeTime,4'd0 } + { difEdgeTime,3'd0 } + {
difEdgeTime,1'd0 } + difEdgeTime;
reg [10:0] difEdgeTimeTimeCorr; wire [10:0] EdDiff = difEdgeTimeTimeCorr -
(( outDesDiff == 4'd9 ) ? 7'd90 :
```

```

        (( outDesDiff == 4'd8 ) ? 7'd80 :
        (( outDesDiff == 4'd7 ) ? 7'd70 :
        (( outDesDiff == 4'd6 ) ? 7'd60 :
        (( outDesDiff == 4'd5 ) ? 7'd50 :
        (( outDesDiff == 4'd4 ) ? 7'd40 :
        (( outDesDiff == 4'd3 ) ? 7'd30 :
        (( outDesDiff == 4'd2 ) ? 7'd20 :
        (( outDesDiff == 4'd1 ) ? 7'd10 : 7'd0 )))))));

always @( posedge inCLK )
begin
difEdgeTime <= inTopEdgeNpix - inBotEdgeNpix;
oldDifEdgeTime <= difEdgeTime;
incDifOldNewEdgesTime <= oldDifEdgeTime < difEdgeTime;
DifOldNewEdgesTime <= ( oldDifEdgeTime < difEdgeTime ) ? ( difEdgeTime -
oldDifEdgeTime ) : ( oldDifEdgeTime - difEdgeTime );
difEdgeTimeTimeCorr <= difEdgeTimeCorrDiv [15:5];
outDesDiff <= (difEdgeTimeTimeCorr > 11'd89) ? 4'd9 :
((( difEdgeTimeTimeCorr > 11'd79 ) & ( difEdgeTimeTimeCorr < 11'd90)) ? 4'd8 :
        ((( difEdgeTimeTimeCorr > 11'd69 ) &
( difEdgeTimeTimeCorr < 11'd80 )) ? 4'd7 :
        ((( difEdgeTimeTimeCorr > 11'd59 ) &
( difEdgeTimeTimeCorr < 11'd70 )) ? 4'd6 :
        ((( difEdgeTimeTimeCorr > 11'd49 ) &
( difEdgeTimeTimeCorr < 11'd60 )) ? 4'd5 :
        ((( difEdgeTimeTimeCorr > 11'd39 ) &
( difEdgeTimeTimeCorr < 11'd50 )) ? 4'd4 :
        ((( difEdgeTimeTimeCorr > 11'd29 ) &
( difEdgeTimeTimeCorr < 11'd40 )) ? 4'd3 :
        ((( difEdgeTimeTimeCorr > 11'd19 ) &

```

```

( difEdgeTimeTimeCorr < 11'd30 )) ? 4'd2 :
                ((( difEdgeTimeTimeCorr > 11'd9) &
( difEdgeTimeTimeCorr < 11'd20 )) ? 4'd1 : 4'd0
                ))))));

```

```

outEdDiff <= EdDiff [3:0];
end

```

```

reg [3:0] oldEdMDU;
reg [3:0] oldDecMDU;
reg [3:0] oldEdBdu;

```

```

always @( posedge inCLK or negedge inNresetDU )

```

```

begin

```

```

if ( ~inNresetDU )

```

```

begin

```

```

    outEdMDU <= 4'd0;
    oldEdMDU <= 4'd0;
    outDecMDU <= 4'd0;
    outEdinBDU <= 4'd3;
    outDecBDU <= 4'd3;
    oldEdBdu <= 4'd0;
    oldDecMDU <= 4'd0;

```

```

end

```

```

else

```

```

begin

```

```

    oldEdMDU <= outEdMDU;
    oldDecMDU <= outDecMDU;

```

```

oldEdBdu <= outEdinBDU;

if ( incDifOldNewEdgesTime & ( DifOldNewEdgesTime > 11'd60 )) outEdMDU <=
(( outEdMDU == 4'd0 ) ? 4'd9 : ( outEdMDU - 1'b1 ));

if ( ~incDifOldNewEdgesTime & ( DifOldNewEdgesTime > 11'd60 )) outEdMDU <=
(( outEdMDU == 4'd9 ) ? 4'd0 : ( outEdMDU + 1'b1 ));

if (( oldEdMDU == 4'd9 ) & ( outEdMDU == 4'd0 )) outDecMDU <= ((
outDecMDU == 4'd9 ) ? 4'd0 : ( outDecMDU + 1'b1 ));

if (( oldEdMDU == 4'd0 ) & ( outEdMDU == 4'd9 )) outDecMDU <= ((
outDecMDU == 4'd0 ) ? 4'd9 : ( outDecMDU - 1'b1 ));

if (( oldDecMDU == 4'd9 ) & ( outDecMDU == 4'd0 )) outEdinBDU <= ((
outEdinBDU == 4'd9 ) ? 4'd0 : ( outEdinBDU + 1'b1 ));

if (( oldDecMDU == 4'd0 ) & ( outDecMDU == 4'd9 )) outEdinBDU <= ((
outEdinBDU == 4'd0 ) ? 4'd9 : ( outEdinBDU - 1'b1 ));

if (( oldEdBdu == 4'd9 ) & ( outEdinBDU == 4'd0 )) outDecBDU <= (( outDecBDU
== 4'd5 ) ? 4'd0 : ( outDecBDU + 1'b1 ));

if (( oldEdBdu == 4'd0 ) & ( outEdinBDU == 4'd9 )) outDecBDU <= (( outDecBDU
== 4'd0 ) ? 4'd5 : ( outDecBDU - 1'b1 ));

end

end

endmodule

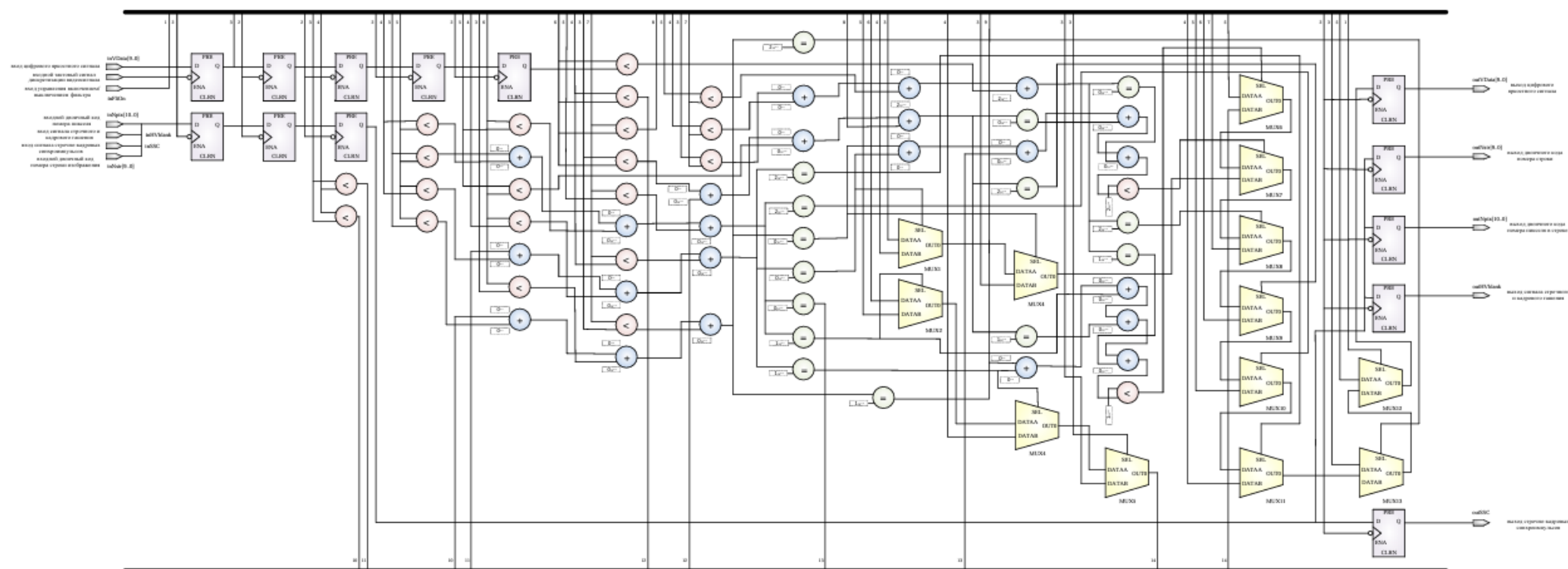
```

## ПРИЛОЖЕНИЕ 3

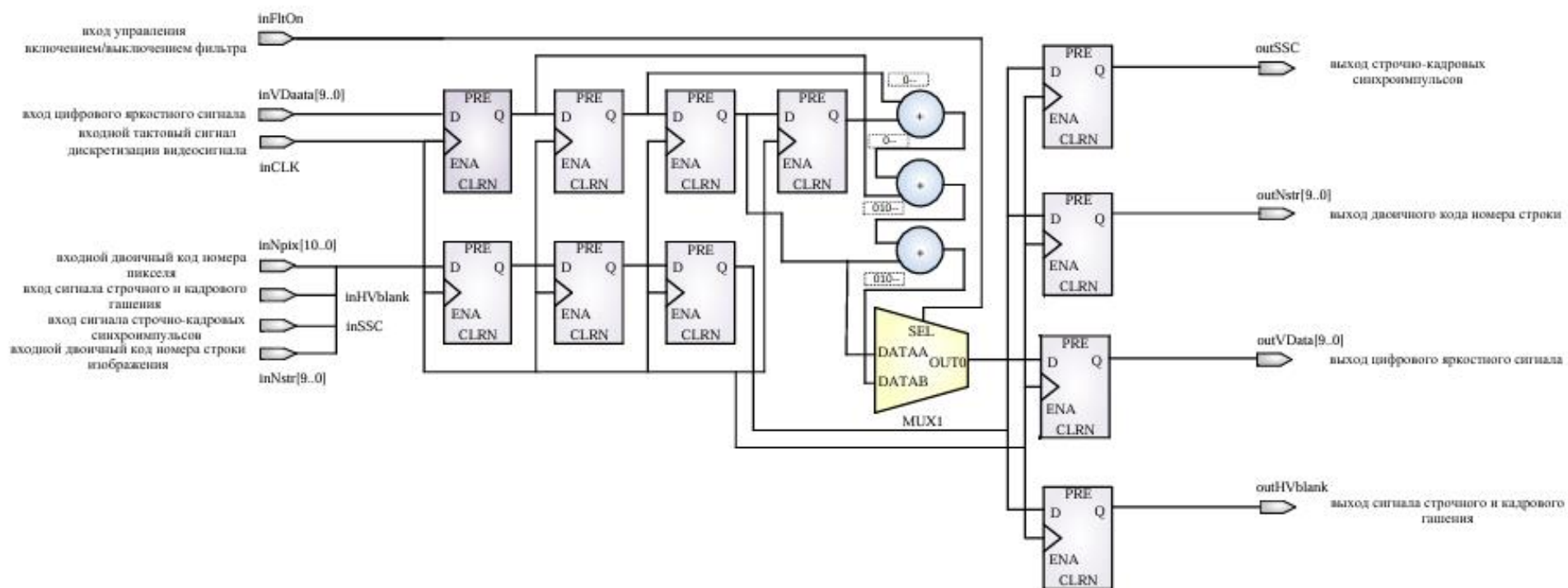
(обязательное)

Электрические схемы функциональных узлов видеопроцессора,  
синтезированные по поведенческим моделям

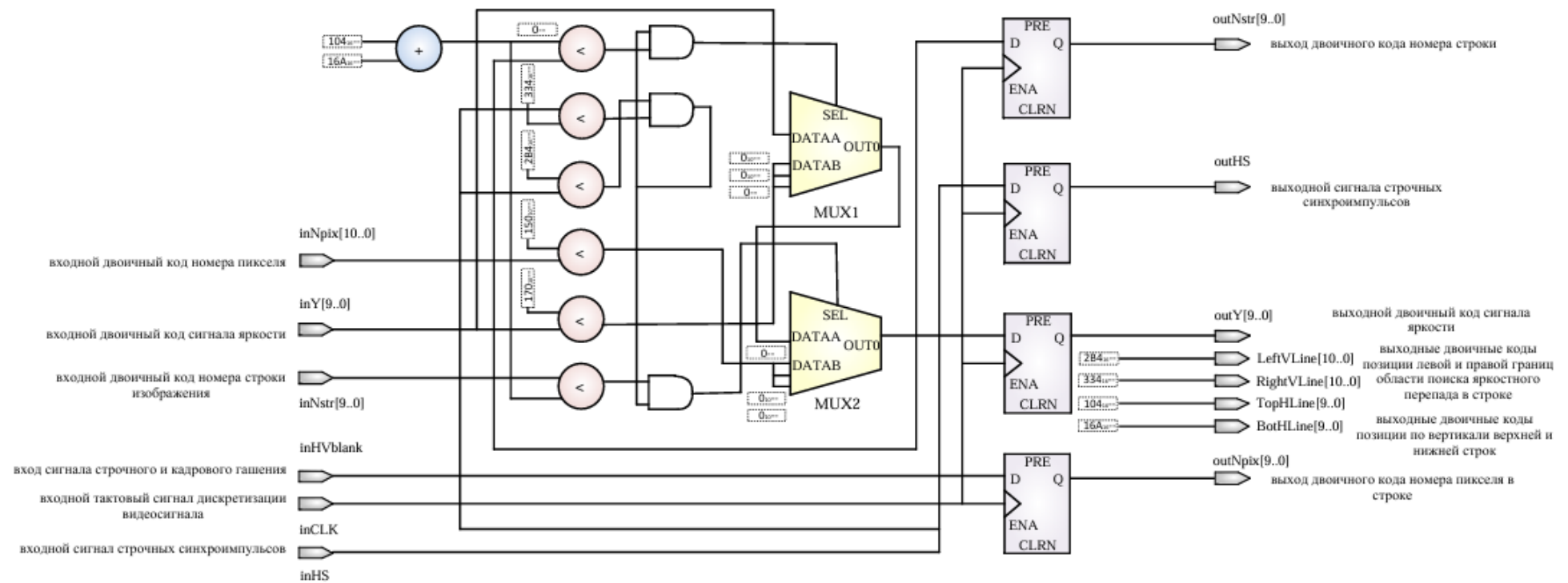
# Медианный фильтр видеоизображения



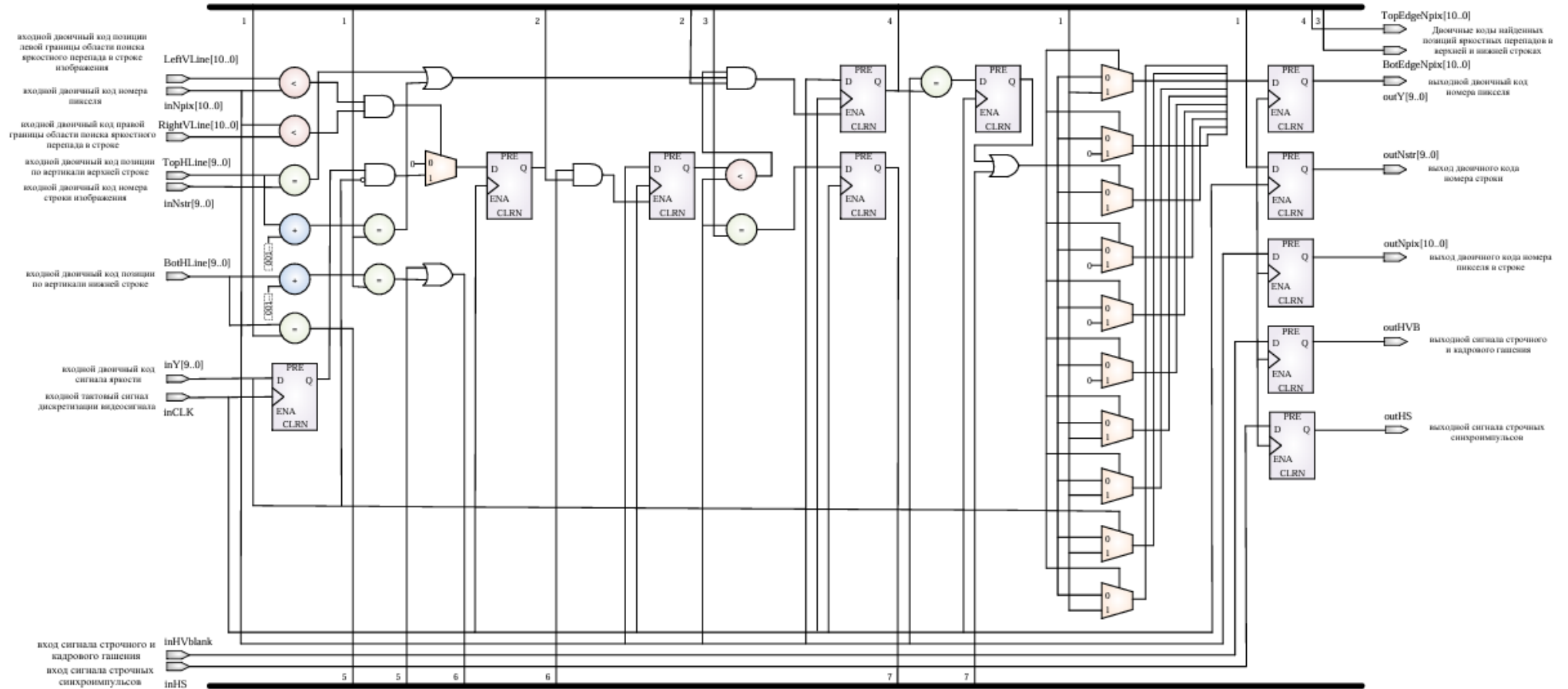
## Усредняющий фильтр видеоизображения



# Бинаризатор видеоизображения

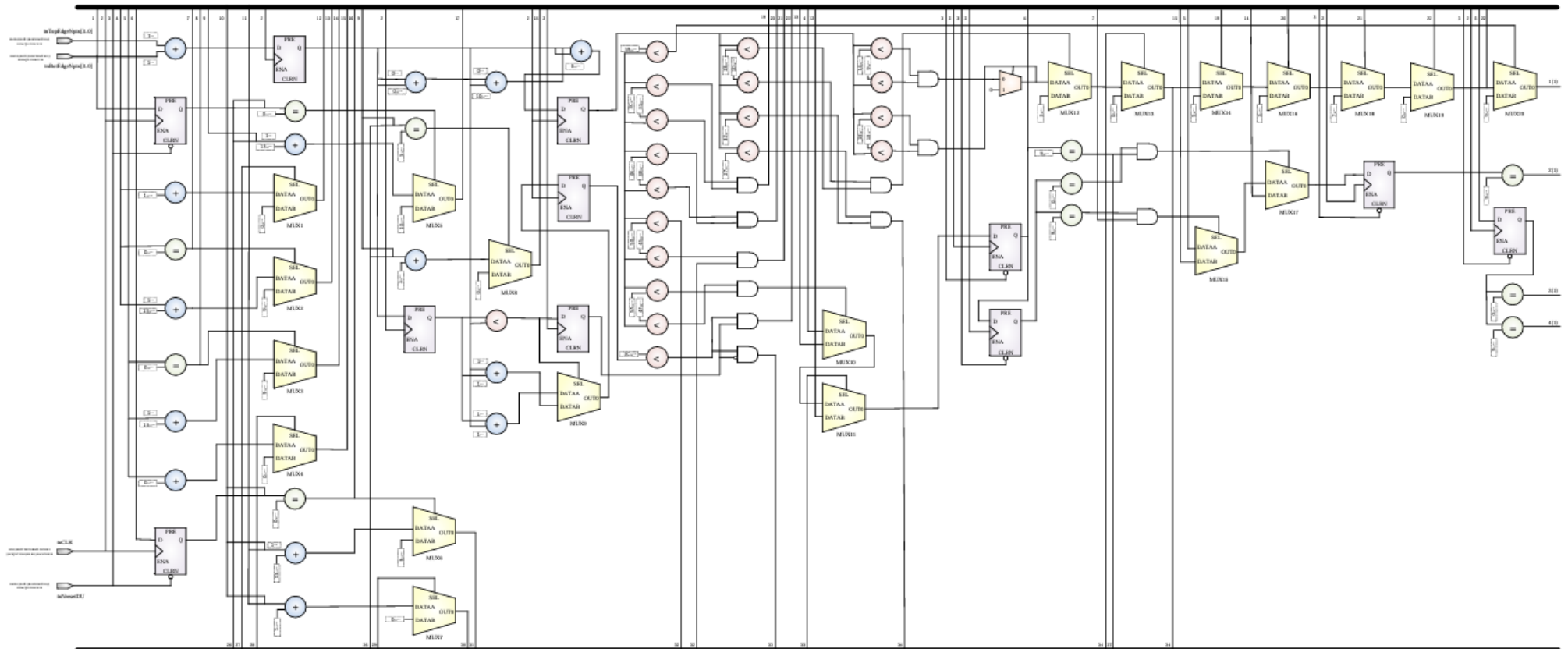


## Детектор границ делений шкалы



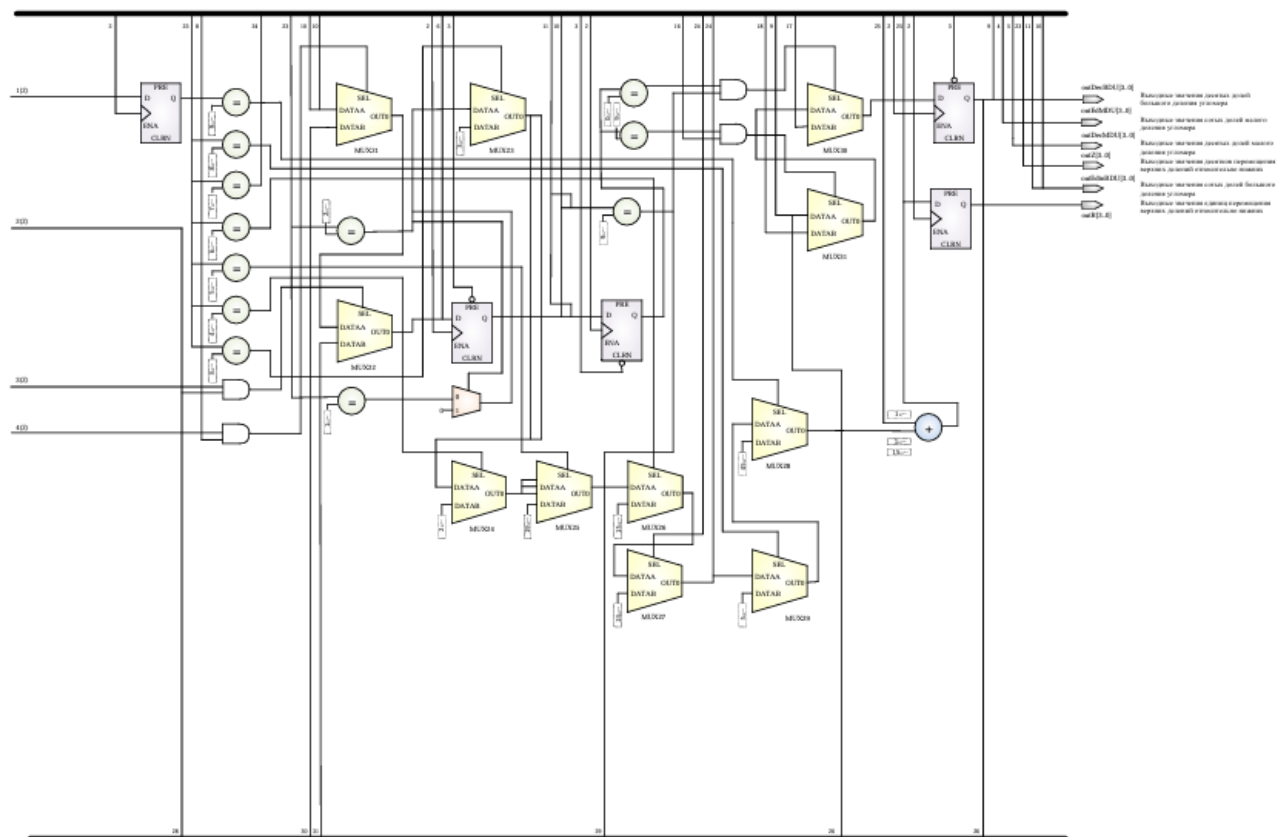
# Вычислитель значений углов

## 1 часть



# Вычислитель значений углов

## 2 часть



Условные обозначения



## ПРИЛОЖЕНИЕ И

(обязательное)

Материал, иллюстрирующий результаты магистерской диссертации



## **Элементы интегральных видеопроцессоров средств технического зрения для автоматизации механических угломеров**

Студент группы 24ПДм1 Бакаева М.Р.

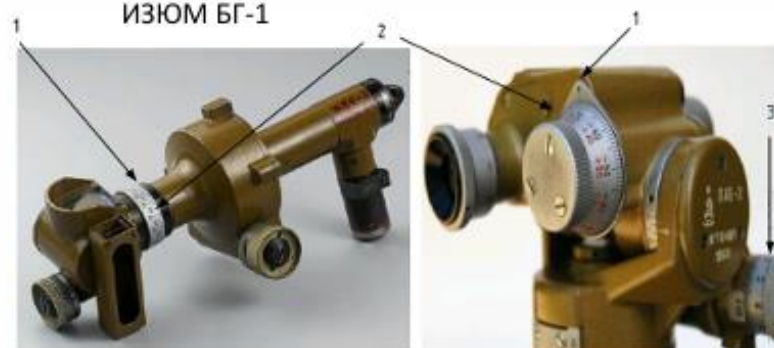
Руководитель - к.т.н., Вареник Ю.А.

## Актуальность и практическая значимость работы



Строительная буссоль  
ИЗЮМ БГ-1

Нониусная шкала на стенке



Панорама Герца

Артиллерийская буссоль ПАБ-2М

- 1 - опорная риска;
- 2 - лимб малых делений угломера;
- 3 - лимб больших делений угломера

Механические угломеры в составе изделий

1

## Цель и задачи работы

### Цель проведенной работы:

совершенствование программно-технических средств автоматизации измерения углов.

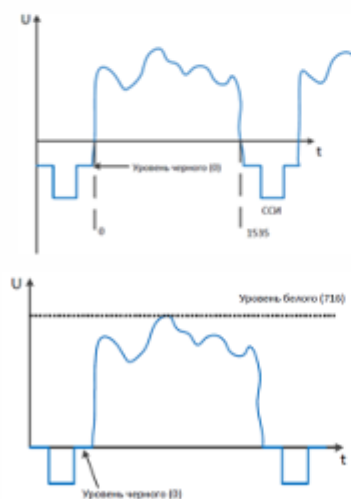
### Задачи:

1. Выполнить анализ:
  - современных механически и оптико-электронных средств измерения углов;
  - методов обработки изображений для распознавания объектов;
2. Разработать элементы методики цифровой обработки видеосигнала для распознавания перемещения и считывания показаний шкал;
3. Разработать структурную схему видеопроцессора средств технического зрения для распознавания перемещения и считывания показаний шкал;
4. Разработать поведенческие модели функциональных узлов видеопроцессора на языке *Verilog*, синтезировать с помощью современных программных средств их электрические схемы.

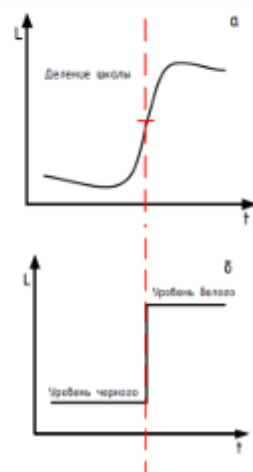
## Элементы методики цифровой обработки видеосигнала

Алгоритм предлагаемой методики включает следующие этапы:

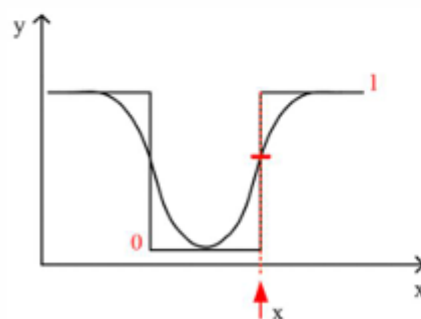
1. Нормализация видеосигнала для получения значений яркости элементов изображения в диапазоне, необходимом для дальнейших вычислений;
2. Фильтрация видеосигнала для ослабления шумов на изображении и его бинаризации;
3. Детектирование границ изображения делений шкал механического угломера и фиксация координат их положения на изображении;
4. Вычисление угла поворота лимба по линейным координатам положения делений;
5. Дополнение изображения значениями вычисленных углов, передача их через цифровой интерфейс в персональный компьютер.



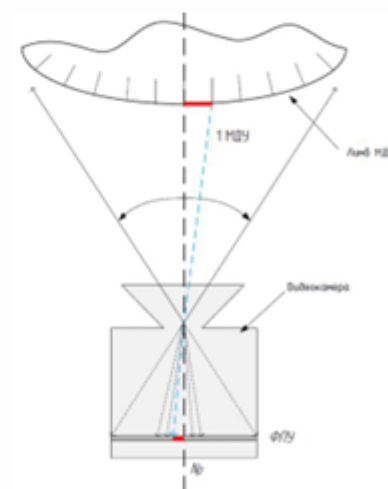
Нормализация видеосигнала



Процесс бинаризации

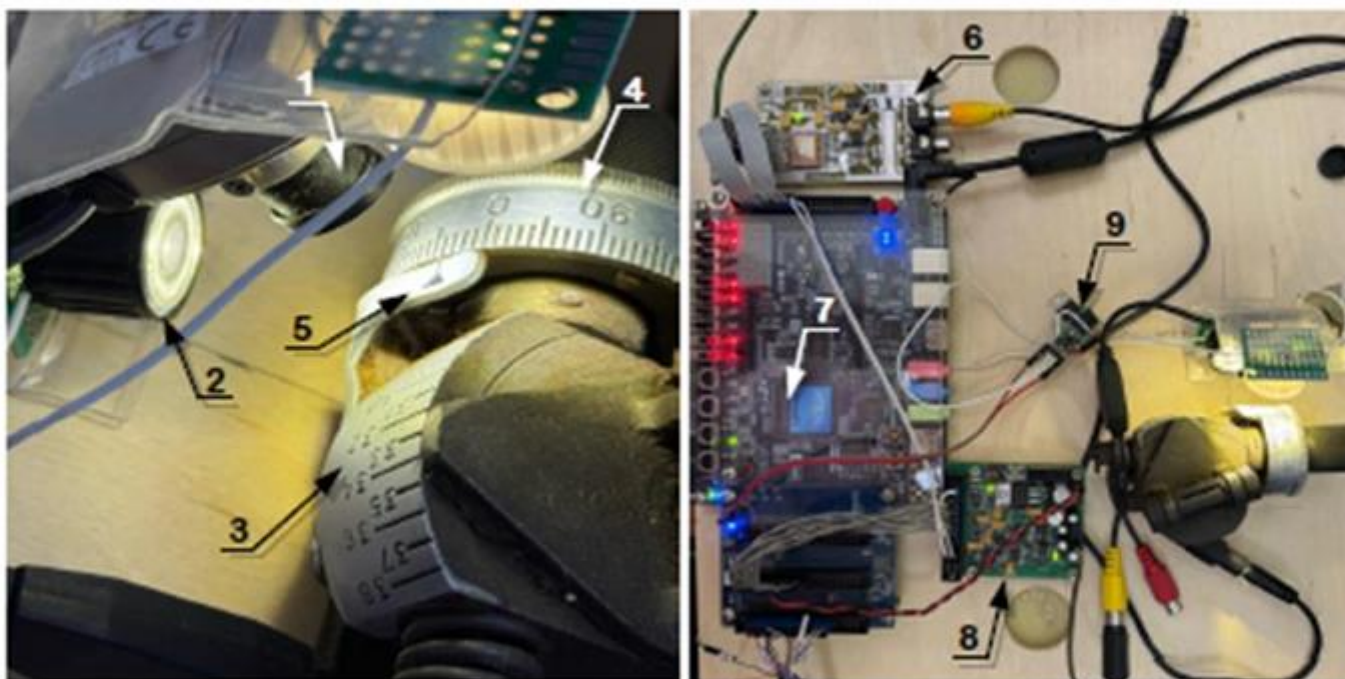


Определения перепада среза и фронта видеосигнала-детектирование границ



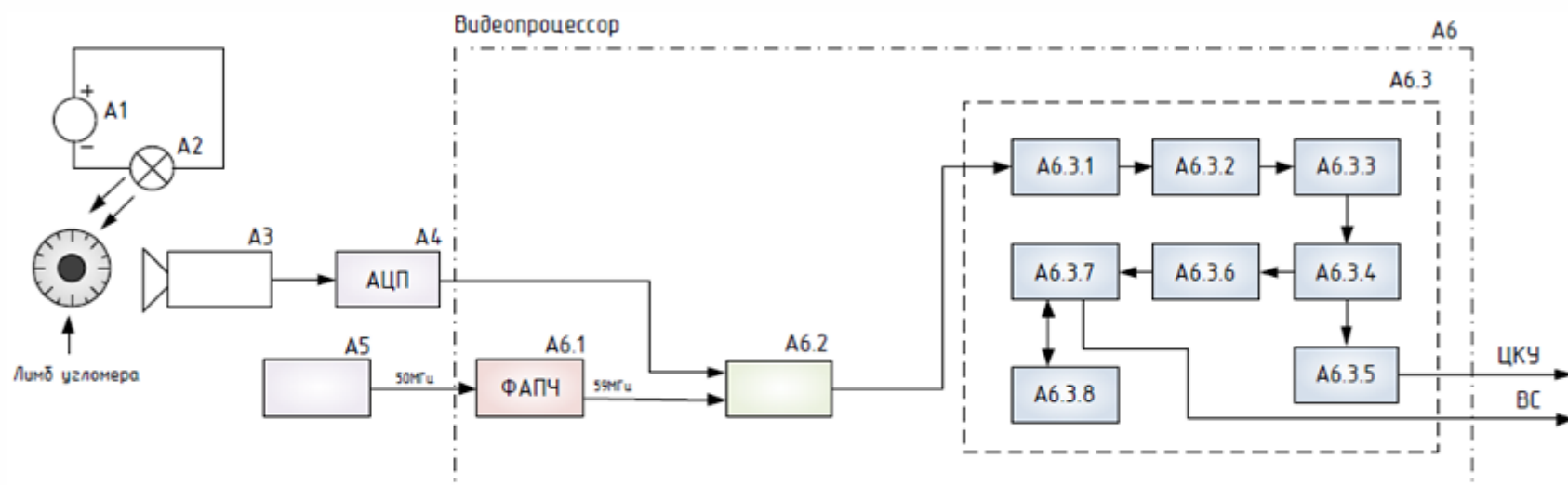
Вычисление угла поворота лимба

## Структура видеопроцессора средств технического зрения для распознавания перемещения и считывания показаний шкал



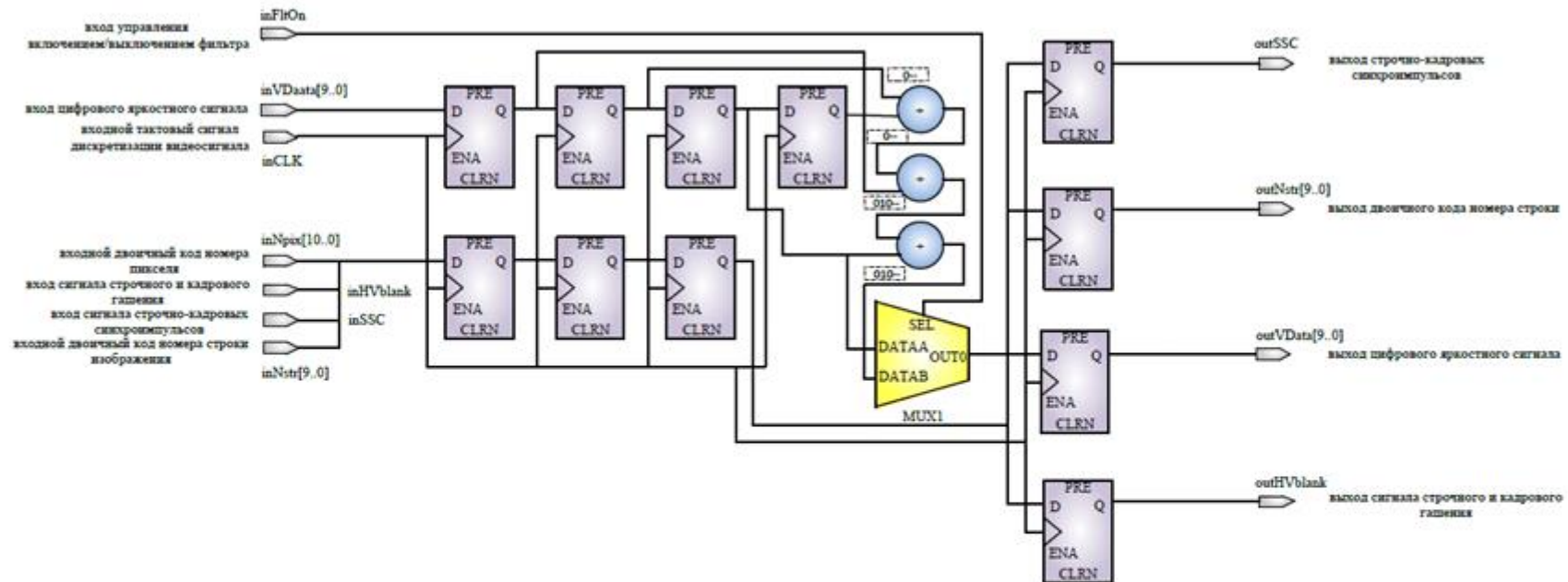
а) 1 – видеокамера; 2 - источники света; 3 - лимб БДУ; 4 - лимб МДУ;  
5 - опорная риска; б) 6 - цифроаналоговый преобразователь; 7- ПЛИС;  
8 - видео АЦП; 9 - источник тока для питания светодиодов

## Структура телевизионной приставки для механического угломера



A1-источник питания; A2-светодиодный источник света;  
 A3-видеокамера; A4-аналогово – цифровой преобразователь (АЦП);  
 A5 - задающий генератор; A6-видеопроцессор;  
 A6.1 - фазовая автоподстройка частоты (ФАПЧ);  
 A6.2 – блок видеозахвата; A6.3.6-блок вывода графической информации;  
 ЦКУ – вывод цифрового кода угла; ВС - вывод видеосигнал

## Усредняющий фильтр видеоизображения



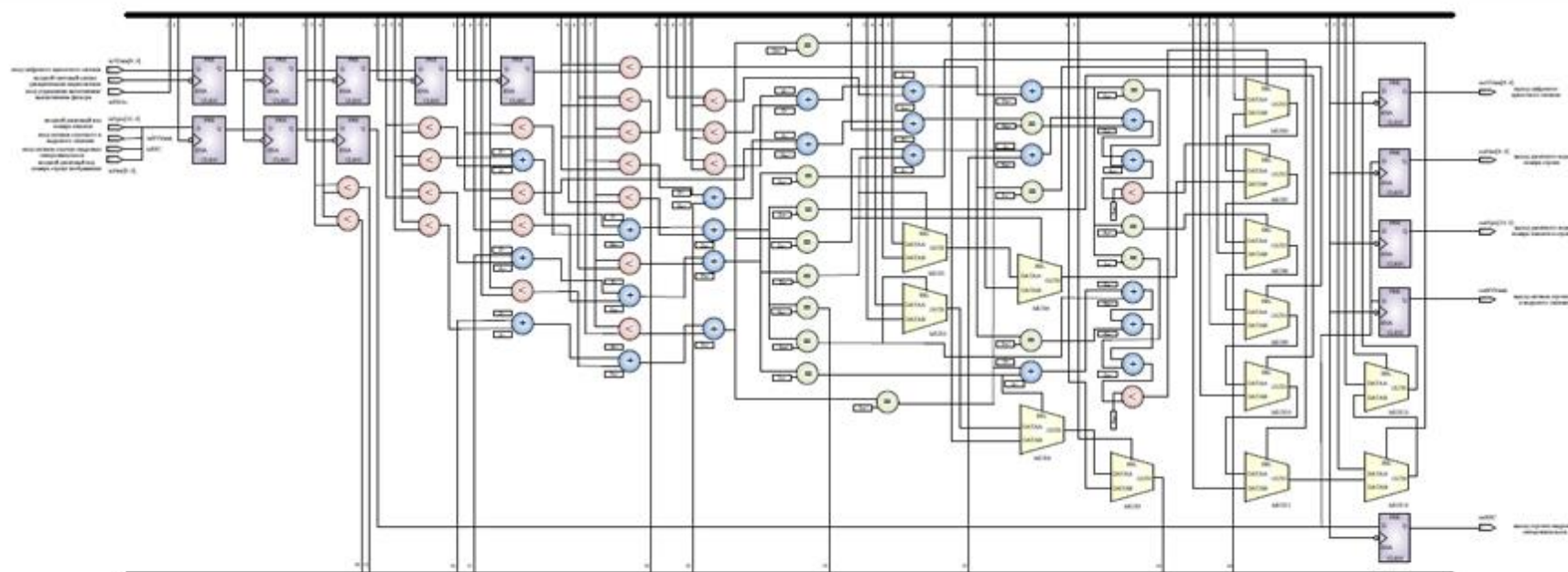
### Входные сигналы:

**inFltOn** - вход управления включением/выключением фильтра;  
**inCLK** - входной тактовый сигнал дискретизации видеосигнала;  
**inNpix[10..0]** - входной двоичный код номера пикселя;  
**inHVblank** - вход сигнала строчного и кадрового гашения;  
**inSSC** - вход сигнала строчно-кадровых синхроимпульсов;  
**inNstr[9..0]** - входной двоичный код номера строки изображения.

### Выходные сигналы:

**outSSC** - выход строчно-кадровых синхроимпульсов;  
**outNstr[9..0]** - выход двоичного кода номера строки;  
**outVData[9..0]** - выход цифрового яркостного сигнала;  
**outHVblank** - выход сигнала строчного и кадрового гашения.

## Медианный фильтр видеоизображения



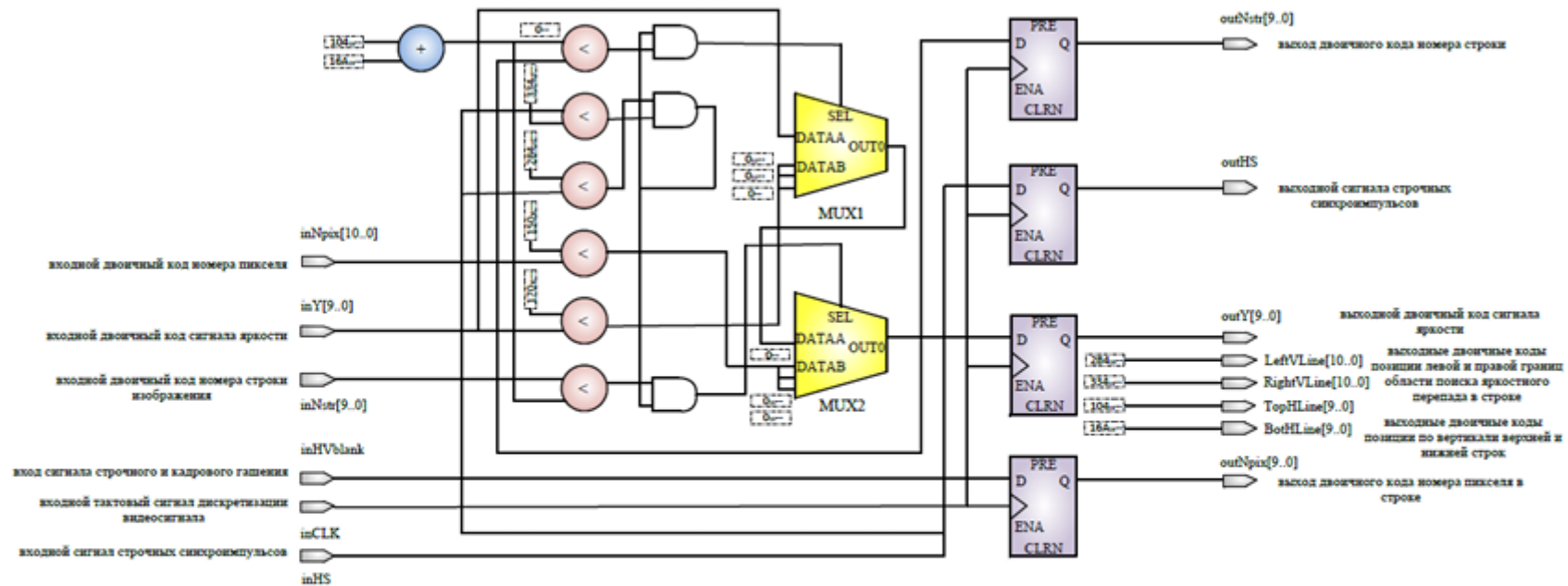
### Входные сигналы:

**inHVblank** – вход сигнала строчного и кадрового гашения;  
**inSSC** – вход сигнала строчно-кадровых синхроимпульсов;  
**inVData** – вход цифрового яркостного сигнала;  
**inNpix** – входной двоичный код номера пикселя;  
**inNstr** – входной двоичный код номера строки изображения;  
**inCLK** – входной тактовый сигнал дискретизации видеосигнала;  
**inFltOn** – вход управления включением/выключением фильтра.

### Выходные сигналы:

**outHVblank** – выход сигнала строчного и кадрового гашения;  
**outSSC** – выход строчно-кадровых синхроимпульсов;  
**outVData** – выход цифрового яркостного сигнала;  
**outNpix** – выход двоичного кода номера пикселя в строке;  
**outNstr** – выход двоичного кода номера строки.

## Бинаризатор видеоизображения



### Входные сигналы:

**inHVblank** – вход сигнала строчного и кадрового гашения;

**inY** – входной двоичный код сигнала яркости;

**inNpix** – входной двоичный код номера пикселя;

**inNstr** – входной двоичный код номера строки изображения;

**inCLK** – входной тактовый сигнал дискретизации видеосигнала;

**inHS** – вход сигнала строчных синхроимпульсов.

### Выходные сигналы:

**outY** – выходной двоичный код сигнала яркости;

**outNpix** – выход двоичного кода номера пикселя в строке;

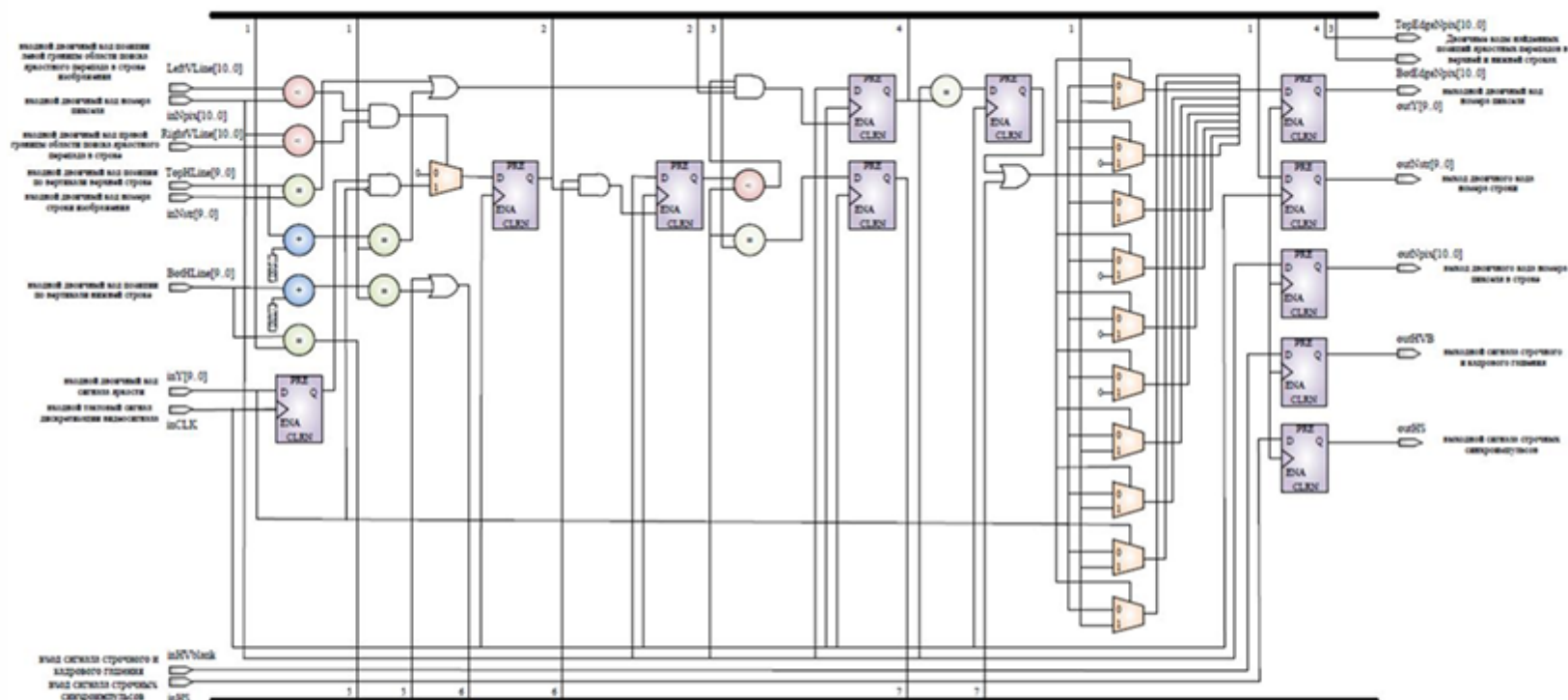
**outNstr** – выход двоичного кода номера строки;

**outHS** – выход сигнала строчных синхроимпульсов;

**outHVB** – выходной сигнала строчного и кадрового гашения.

8

## Детектор границ делений шкалы



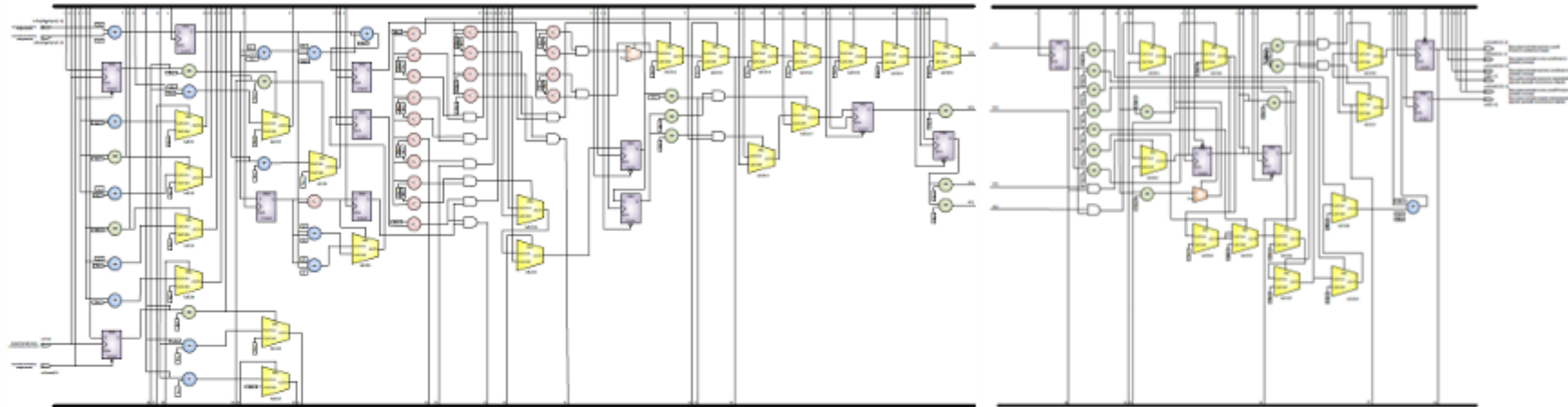
### Входные сигналы:

**LeftVLine[10.0]** - входной двоичный код позиции левой границы области поиска яркостного перепада в строке изображения;  
**inNpix[10.0]** - входной двоичный код номера пикселя;  
**RightVLine[10.0]** - входной двоичный код правой границы области поиска яркостного перепада в строке;  
**TopHLine[9.0]** - входной двоичный код позиции по вертикали верхней строке;  
**inNstr[9.0]** - входной двоичный код номера строки изображения;  
**BotHLine[9.0]** - входной двоичный код позиции по вертикали нижней строке;  
**inY[9.0]** - входной двоичный код сигнала яркости;  
**inCLK** - входной тактовый сигнал дискретизации видеосигнала;  
**inHVblank** - вход сигнала строчного и кадрового гашения;  
**inHS** - вход сигнала строчных синхроимпульсов.

### Выходные сигналы:

**TopEdgeNpix[10.0]** и **BotEdgeNpix[10.0]** - двоичные коды найденных позиций яркостных перепадов в верхней и нижней строках;  
**outY[9.0]** - выходной двоичный код номера пикселя;  
**outNstr[9.0]** - выход двоичного кода номера строки;  
**outNpix[10.0]** - выход двоичного кода номера пикселя в строке;  
**outhVB** - выходной сигнала строчного и кадрового гашения;  
**outhS** - выходной сигнала строчных синхроимпульсов.

## Вычислитель значений углов



### Входные сигналы:

**inTopEdgeNpix[3..0]**-входной двоичный код номера пикселя в верхней строке;

**inBotEdgeNpix[3..0]**-входной двоичный код номера пикселя в нижней строке;

**inCLK**- входной тактовый сигнал дискретизации видеосигнала;

**inNresetDU**-выходной двоичный код номера пикселя.

### Выходные сигналы:

**outDecBDU[3..0]**-выходные значения десятых долей большого деления угломера;

**outEdMDU[3..0]**-выходные значения сотых долей малого деления угломера;

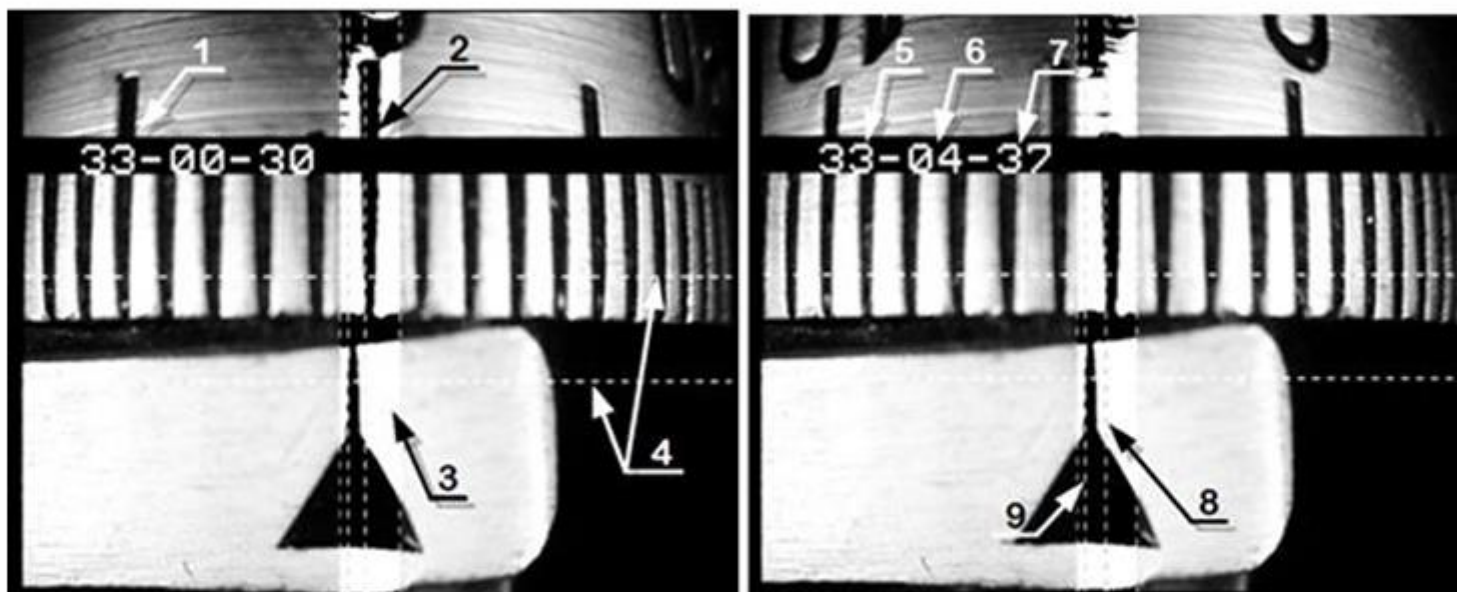
**outDecMDU[3..0]**- выходные значения десятых долей малого деления угломера;

**outEdinBDU[3..0]**- выходные значения сотых долей большого деления угломера;

**outZ[3..0]**- выходные значения десятков перемещения верхних делений относительно нижних;

**outR[3..0]**- выходные значения единиц перемещения верхних делений относительно нижних.

## Кадры видеозображения дополненного измерительной информацией на выходе видеопроцессора



- а) 1 – отметка лимба 5МДУ; 2 – отметка лимба 0МДУ; 3 – контрольный участок изображения;  
4 – строки отслеживания делений лимба;
- б) измеренное значение угла: 5 – БДУ; 6 – МДУ; 7 – сотые доли МДУ;  
8 – распознанное положение делений лимба; 9 – распознанное положение опорной риски

## Результат работы

| №<br>п/п                                     | Наименование работы и ее вид                                                                                          | Форма работы      | Выходные данные                                                                                     | Объем в с. | Соавторы                     |
|----------------------------------------------|-----------------------------------------------------------------------------------------------------------------------|-------------------|-----------------------------------------------------------------------------------------------------|------------|------------------------------|
| Статьи, посвященные теме работе              |                                                                                                                       |                   |                                                                                                     |            |                              |
| 1                                            | Устройство автоматизации механического угломера (статья, конференция)                                                 | печатная<br>очная | Перспективы развития информационных технологий систем управления. – Пенза, Изд-во ПГУ, 2025. - 8с.  | 8/4        | Вареник Ю.А.<br>Волик А.В.   |
| 2                                            | Цифровые блоки видеопроцессора автоматизации считывания показаний лимбов механических угломеров (статья, конференция) | печатная<br>очная | Перспективы развития информационных технологий систем управления. – Пенза, изд-во ПГУ, 2026. – 4 с. | 4/2        | Вареник Ю.А.<br>Рязяпов И.Н. |
| Научные конференции, посвященные теме работе |                                                                                                                       |                   |                                                                                                     |            |                              |
| 1                                            | Устройство автоматизации механического угломера (статья, конференция)                                                 | очная             | I Научно-техническая конференция молодых ученых, специалистов и студентов                           |            |                              |
| 2                                            | Цифровые блоки видеопроцессора автоматизации считывания показаний лимбов механических угломеров (статья, конференция) | очная             | II Научно-техническая конференция молодых ученых, специалистов и студентов                          |            |                              |

Полученные результаты использованы в научно-исследовательских и опытно-конструкторских работах «АО «НПП Рубин»

## РЕЦЕНЗИЯ

на магистерскую диссертацию  
студента гр.24ПДм1 факультета ИТЭ  
направление подготовки 11.04.04 – Электроника и наноэлектроника  
Бакаевой Марии Романовны

**На тему: «Элементы интегральных видеопроцессоров средств технического зрения для автоматизации механических угломеров»**

Магистерская диссертация посвящена актуальной научно-технической задаче – разработке элементов интегрального видеопроцессора средств технического зрения для автоматизации механических угломеров. Целью рецензируемой работы является совершенствование программно-технических средств автоматизации измерения углов. Актуальность темы обусловлена тем, что традиционные механические угломеры, несмотря на широкое распространение в машиностроении, строительстве, геодезии и других сферах, имеют в современных условиях применения ряд недостатков, таких как: невозможность без участия оператора передачи измеренных значений в программно-технические комплексы автоматизации контроля, измерения и управления, а также высокая вероятность субъективных ошибок при считывании показаний шкал оператором. В связи с этим задача внедрения средств телевизионного считывания – то есть использование видеокамеры и алгоритмов цифровой обработки изображения для автоматического определения и цифровой индикации угла – является востребованной для измерительной техники. Особое внимание уделено архитектуре разработанного специализированного видеопроцессора, его основным функциональным узлам и особенностям обработки изображения в реальном времени.

Структура диссертации логична и последовательно раскрывает тему работы. В первой главе описаны различные виды угломеров и методы обработки видеоизображения, применяемые для решения задач распознавания объектов. Во второй главе подробно описаны элементы методики цифровой обработки видеосигнала для распознавания перемещения и считывания показаний шкал. Третья глава носит прикладной характер и содержит структуру телевизионной приставки для механического угломера, а также результаты моделирования поведенческих моделей и синтеза электрических схем функциональных узлов видеопроцессора.

К достоинствам работы следует отнести: актуальность выбранной темы, наличие практической реализации, высокий уровень теоретической проработки. Вместе с тем работа не лишена отдельных недостатков: отсутствие расширенного анализа устойчивости системы к внешним помехам и ограниченное рассмотрение вопросов энергоэффективности действующего макета. Указанные замечания не носят принципиального характера и не снижают общей высокой оценки диссертационной работы. Они носят рекомендательный характер и могут быть учтены при дальнейших исследованиях.

В целом магистерская диссертация «Элементы интегральных видеопроцессоров средств технического зрения для автоматизации механических угломеров» является законченной квалификационной работой, выполненной на актуальную тему, содержащей элементы научной новизны и практическую значимость. Тема работы автором раскрыта полностью. Магистром были верно определены задачи, необходимые для достижения поставленной цели. Работа соответствует требованиям, предъявленным к работам магистра, а ее автор Бакаева М.Р. заслуживает присуждения степени магистра по направлению подготовки 11.04.04 «Электроника и наноэлектроника» по профилю «Нано- и микроэлектроника» с оценкой «отлично».

Начальник отдела,  
АО «ПО «Электроприбор»



А.В. Комиссаров

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РФ  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования

«ПЕНЗЕНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

**ОТЗЫВ**  
**РУКОВОДИТЕЛЯ ВЫПУСКНОЙ КВАЛИФИКАЦИОННОЙ РАБОТЫ**  
**МАГИСТРА**

Выпускная работа магистра на тему:

Элементы интегральных видеопроцессоров средств технического зрения для автоматизации механических угломеров

Выполнена студентом: М.Р. Бакаева

Факультет: Информационных технологий и электроники

Кафедра: «Нано- и микроэлектроника»

Группа: 24ПДм1

Направление подготовки, профиль:

11.04.04 «Электроника и наноэлектроника», «Нано- и микроэлектроника»

Квалификация (степень): магистр

Руководитель: Вареник Юрий Александрович

к.т.н., доцент кафедры «НиМЭ»

Качества выпускника, выявленные в ходе выполнения выпускной квалификационной работы

При выполнении ВКР Бакаева М.Р. проявила себя грамотным специалистом, способным самостоятельно решать сложные технические задачи, продемонстрировала умение работать с современной элементной базой, знание компьютерных технологий и программ автоматизированного проектирования, а также владение математическим аппаратом. За время выполнения ВКР Марию Романовну изучены современные методы цифровой обработки видеозаписей, элементы современной цифровой телевизионной техники, методы и программные средства проектирования полупроводниковых интегральных микросхем на основе ПЛИС. Выпускником самостоятельно разработана методика обработки видеосигнала и цифровые электронные узлы видеопроцессора на базе ПЛИС, позволяющие распознавать и бесконтактно считывать показания лимбов механических угломеров, а также разработан, изготовлен и испытан действующий макет цифровой системы технического зрения для решения поставленной задачи. Результаты ВКР Бакаевой М.Р. успешно использованы в опытно-конструкторской работе АО «НПП «Рубин». Марию Романовну рекомендую для приёма в аспирантуру ввиду проявленных ею высокого уровня интеллекта, ответственности, самостоятельности и способности к решению исследовательских задач.

#### Заключение

Выполненная М.Р. Бакаевой ВКР полностью соответствует техническому заданию. Степень соответствия подготовленности выпускника соответствует требованиям к результатам освоения программы магистратуры ФГОС ВО по направлению подготовки «Электроника и нанoeлектроника». На основании вышеизложенного считаю, что ВКР заслуживает оценки «отлично», а её автор, Бакаева Мария Романовна, присвоения квалификации «магистр» по направлению подготовки 11.04.04 «Электроника и нанoeлектроника».

«28» мая 2026г.

Руководитель



Ю.А. Вареник