

УДК 004.5

## ПРОЕКТИРОВАНИЕ СИСТЕМЫ ПОДДЕРЖКИ ПРИНЯТИЯ РЕШЕНИЯ ДЛЯ КОНТРОЛЯ ТЕКУЩЕГО УРОВНЯ ЖИДКОСТИ С ПОМОЩЬЮ УДАЛЕННОГО ВИДЕОАНАЛИЗА

Шустова К.П.

*ФГБОУ ВО «Казанский государственный энергетический университет»,  
Казань, e-mail: kseniyashustova@yandex.ru*

В настоящей статье спроектирована система для контроля текущего уровня жидкости с помощью удаленного видеоанализа с нескольких видеокамер. На видеопотоке, поступающем с каждой камеры, можно следить за уровнями жидкости в разных указанных пользователем областях интереса, находящихся в обзоре выбранной камеры. Приведена UML-диаграмма деятельности работы этой системы и предложен алгоритм определения текущего уровня жидкости в указанных пользователем объектах интереса. Предложенный алгоритм может быть применен как для пенящихся, так и непенящихся жидкостей. В системе можно реализовывать настройку параметров наблюдения, вида изображения на месте, т.е. в своих конкретных условиях освещенностей и шумов. В базу знаний разрабатываемой в настоящей статье системы заложен алгоритм, основанный на операциях удаления шумов и бинаризации с указанными пользователем через интерфейс верхними границами параметров методов и их текущих значений. В качестве примеров программирования ключевых моментов предложенного алгоритма приведены коды программы в системе Mathematica 11, в которых реализованы предложенные в работе приемы реализации этих моментов. Эта система спроектирована с учетом встройки ее в качестве модуля в прикладной раздел автоматизированной системы, разрабатываемой автором совместно с М.Д. Миссаровым, Е.П. Шустовой и А.Ю. Соловьяненко.

**Ключевые слова:** уровень жидкости, видеоанализ, система поддержки принятия решений, UML, диаграмма деятельности, Mathematica

## DESIGNING A DECISION SUPPORT SYSTEM FOR CONTROL OF THE CURRENT LIQUID LEVEL USING REMOTE VIDEO ANALYSIS

Shustova K.P.

*Kazan State Energy University, Kazan, e-mail: kseniyashustova@yandex.ru*

In this article, a system has been designed for monitoring the current liquid level using remote video analysis from several video cameras. On the video stream coming from each camera, you can monitor the liquid levels in different user-specified ROIs in the view of the selected camera. A UML diagram of the operation of this system is given and an algorithm for determining the current liquid level in the objects of interest specified by the user is proposed. The proposed algorithm can be applied to both foaming and non-foaming liquids. In the system, you can configure the monitoring parameters by seeing the image on the spot, i.e. in their specific lighting and noise conditions. The knowledge base of the system developed in this article contains an algorithm based on the operations of noise removal and binarization with the upper bounds of the method parameters and their current values specified by the user through the interface. As examples of programming the key points of the proposed algorithm, the codes of the program in the Mathematica 11 system are given, in which the methods for implementing these points proposed in the work are implemented. This system is designed taking into account its integration as a module into the applied section of the automated system, developed by the author together with M.D. Missarov, E.P. Shustova and Solovyanenko A.Yu.

**Keywords:** fluid level, video analysis, decision support system, UML, activity diagram, Mathematica

В настоящее время существуют различные приборы для определения уровня жидкости в емкости. Все они не предполагают удаленного видеоанализа. Еще в 2019 г. автор статьи пришла к выводу о необходимости создания системы для удаленного видеоконтроля текущего уровня жидкости для объекта интереса в видеопотоке, получаемом с камеры. Поэтому в рамках научно-исследовательской работы было предложено студенту КФУ А. Гильманову (направление «Прикладная математика и информатика») реализовать создание такой системы. В базе знаний этой системы было предложено реализовать алгоритм, основанный на обработке изображения,

полученного из текущего с помощью операции бинаризации. Им была создана такая система, но с нарушением некоторых правил проектирования интерфейсов систем. В настоящее время такая система им дорабатывается.

Автором настоящей статьи были замечены недостатки в точности срабатывания указанного выше алгоритма. Поэтому в настоящей статье этот алгоритм улучшен. А также добавлена возможность наблюдения с одной камеры сразу за несколькими объектами интереса. То есть сразу, например, за несколькими байпасными уровнями, попавшими в поле зрения одной камеры. Это поможет сэкономить средства

на закупку и установку камер. К тому же в настоящей работе система спроектирована так, чтобы можно было наблюдать за объектами интереса сразу несколькими камерами, указанными пользователем. Эта функция полезна, например, для случая, когда одна камера не может быть установлена так, чтобы захватить все нужные объекты интереса или для разных объектов интереса нужно разное разрешение.

Проектирование системы в настоящей работе проведено с учетом встройки ее в качестве модуля в прикладной раздел автоматизированной системы, разрабатываемой автором совместно с М.Д. Миссаровым, Е.П. Шустовой и А.Ю. Соловяненко [1–3].

Заметим, что события 2020–2021 гг., связанные с пандемией COVID-19, дополнительно показали актуальность разработки такой системы.

Цель исследования: проектирование системы поддержки принятия решения для контроля текущего уровня жидкости с помощью удаленного видеонализа с нескольких камер с возможностью контроля сразу нескольких объектов интереса, находящихся в поле зрения каждой камеры. Эта система должна быть спроектирована с учетом встройки ее в качестве модуля в прикладной раздел автоматизированной системы, разрабатываемой автором со-

вместно с М.Д. Миссаровым, Е.П. Шустовой и А.Ю. Соловяненко [1–3].

Объекты исследования: видеопотоки.

Предмет исследования: уровень жидкости.

*Алгоритм работы системы  
«Контроль уровня жидкости»*

Нашу систему спроектируем так, чтобы у пользователя была возможность анализировать видеопоток, поступающий как с указанной пользователем камеры, так и из файла. Текущее изображение с указанным на нем  $j$ -м объектом интереса обозначим через  $im[k,j]$  (здесь  $k=0$  означает, что изображение получено из файла, для остальных  $k$  – с камеры под меткой  $k$ ). Само текущее изображение  $im[k,j]$  будем выводить на форму пользователя  $F[k]$ , если видеопоток поступает с камеры, и  $Ffile$ , если видео получено из файла.

Как правило, текущее изображение, получаемое камерой в разных условиях съемки, нуждается в соответствующих методах предварительной его обработки, с целью получения изображения, пригодного для конкретного анализа. Нашу систему будем создавать, соблюдая принцип развития. А именно, например, в Mathematica 11 путем задания порогов параметров и значений этих параметров в виде функций:

```
zadatp[k_,j_,v_,i_] := Grid[{{"Задайте верхнюю грань порога p", i, " ",
  InputField[Dynamic[p2[k,j,v,i]], Number, FieldSize -> 4]
}},
slp[k_,j_,v_,i_] := Grid[{{"p", i,
  Manipulator[Dynamic[p[k,j,v,i]],
    {0, Dynamic[p2[k,j,v,i]]}
}},
}],
```

здесь  $k$  – тип (номер) камеры,  $j$  – номер области интереса,  $v$  – вид операции улучшения изображения (например, бинаризация, медианный фильтр),  $i$  – номер параметра для метода вида  $v$ ,  $p2[k,j,v,i]$  это верхний порог бинаризации,  $p[k,j,v,i]$  текущее значение порога бинаризации. Устанавливать верхний порог бинаризации и его текущее значение пользователь может, введя значение порога в поле ввода и двигая движок на форме во вкладке «Настройка».

Теперь создадим в Mathematica 11 панель  $pul[k,j,v,i]$  для улучшения  $j$ -го объекта интереса на текущем изображении  $im[k,l]$  с помощью движков на форме для параметров  $i$  метода  $v$ :

```
pul[k_,j_,v_,i_] := Panel[
  Column[
    v,
    Dynamic@Grid[{{zadatp[k,j,v,i]}}],
    Dynamic@Grid[{{" ", slp[k,j,v,i]}}]
  ],
  Background -> RGBColor[0.8, 0.8, 0.75]
].
```

Заметим, что в настоящей статье разрабатывается система для наблюдения:

- за показаниями байпасного уровнемера (рис. 1–2),
- за уровнем в прозрачной или полупрозрачной емкости.

Принцип работы байпасных уровнемеров изложен, например, на официальных сайтах компании «РусАвтоматизация Эксперт в датчиках уровня» и ООО «ОЛИЛ» [4, 5].

Видим, что контроль уровня, показываемого байпасным уровнемером, и контроль уровня в жидкости в прозрачной или полупрозрачной емкости может осуществляться по одному и тому же алгоритму. А именно, надо находить границу разделения сред (между повернувшимися и не повернувшимися пластинами в случае байпасного

уровнемера и между жидкостью и воздухом (пенной) в случае пенящихся жидкостей). В базу знаний нашей системы заложим нахождение этой границы снизу вверх в узком столбике. Левый верхний угол этого столбика указывается пользователем с помощью клика около каждого из объектов интереса  $I[k, j]$ , попавшего в обрабатываемый видеофайл при  $k = 0$  и в обзор указанной пользователем камеры для остальных значений  $k$ . На местах клика появляется соответствующий указатель (локатор), который пользователь может передвинуть в случае, если изначально он указал не тот левый угол. При этом формируется легенда, где пользователь может ввести наименование соответствующего объекта интереса  $I[k, j]$ .



Рис. 1. Байпасный измеритель уровня NivoFlip [4]

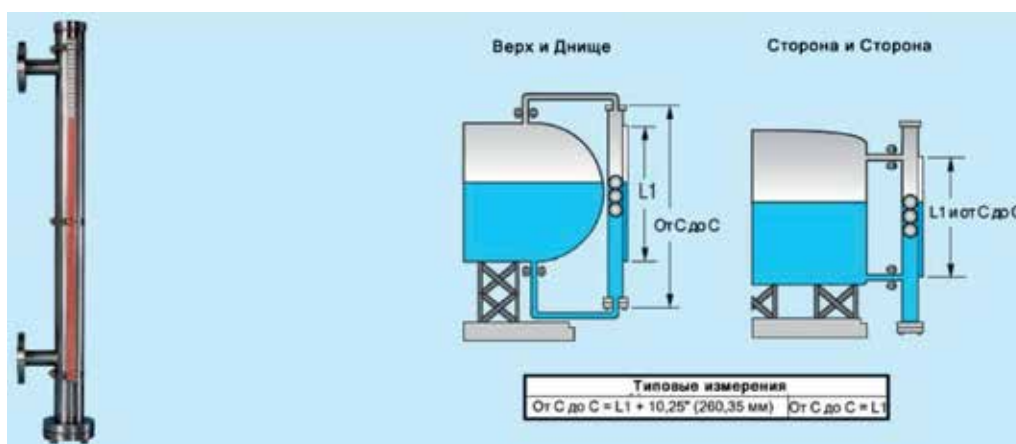


Рис. 2. Индикатор уровня жидкости серии VR [5]

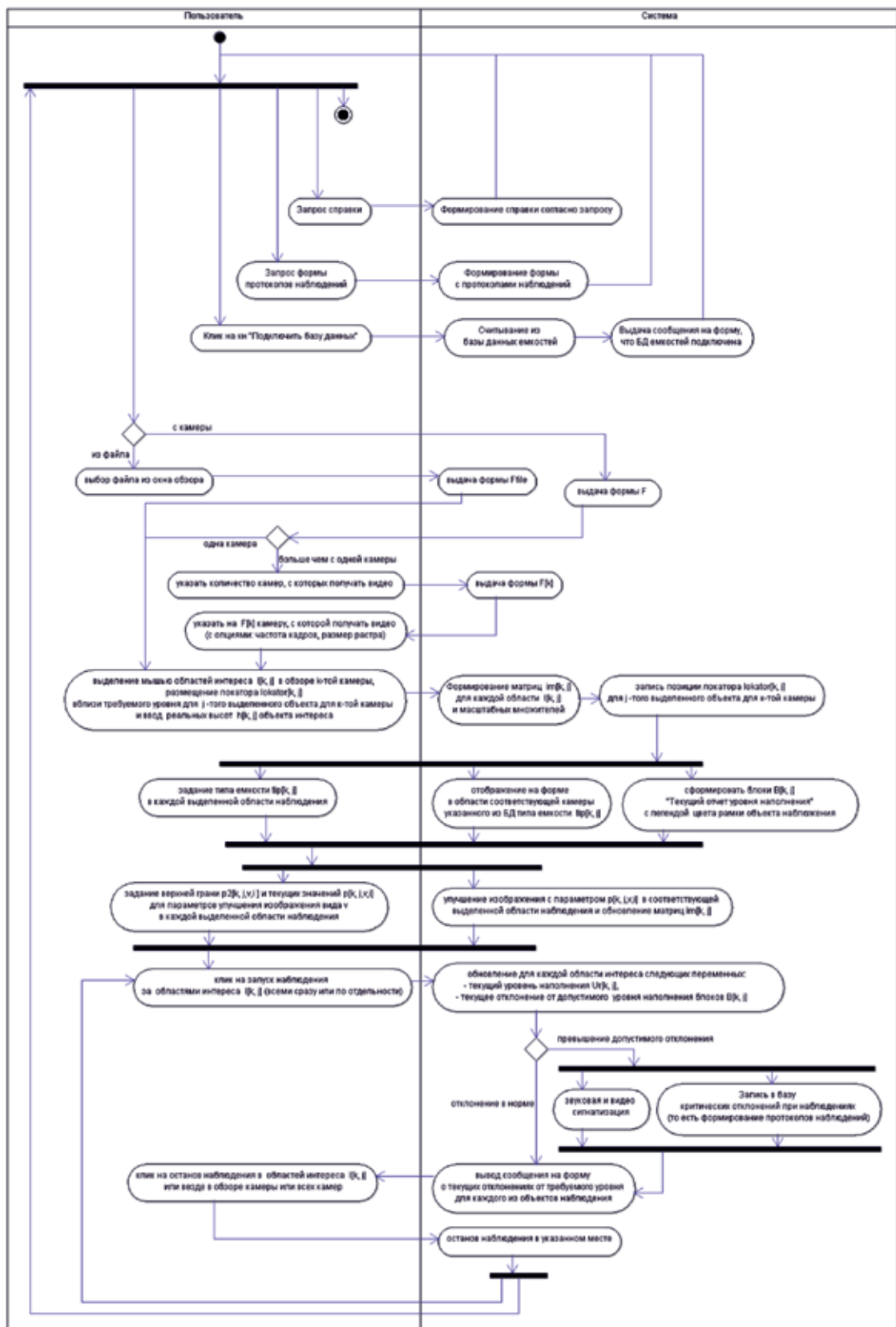


Рис. 3. Логика разрабатываемой системы

Приведена диаграмма деятельности работы в нашей СППР (рис. 3). Поясним приведенную на этом рисунке диаграмму. На ней представлен порядок действий в настоящей СППР для реализации указанного выше метода определения уровня жидкости:

- импорт данных из базы данных (емкостей или байпасных уровнемеров);
- получение текущего изображения (из файла через окно обзора или из камеры (камер)) и выдача соответственно форм *Ffile* или *F[k]*;
- автоматическая бинаризация изображения (порог бинаризации – среднее значение интенсивности пикселей изображения);
- захват области интереса (областей интересов)  $I[k, j]$  в обзоре  $k$ -й камеры и указание их реальных высот  $h[k, j]$  на выведенном видеопотоке (видеопотоках) соответственно на формах *Ffile* или *F[k]*.

Здесь стоит заметить, что в случае нескольких камер надо учитывать пропускную способность USB. Поэтому чтобы изображения с камер передавались, надо регулировать частоту кадров и разрешение (чем больше камер, тем меньше они должны быть). Например, в Mathematica 11 это можно сделать с помощью оператора:

```
Imagecapture[Imagesize -> 200].
```

Это если камеры подключены к одной шине. Заметим, что можно добавить при необходимости еще одну карту USB на свой компьютер, чтобы увеличить пропускную способность.

- осуществление системой поддержки принятия решений соответствующего с предыдущим действием, формирование матриц  $mI[k, j]$  для каждой области  $I[k, j]$  и соответствующих масштабных множителей, а также запись позиции локатора  $lokator[k, j]$  для  $j$ -го выделенного объекта для  $k$ -й камеры;

- улучшение пользователем бинарного изображения области (областей) интереса  $I[k, j]$  на формах *Ffile* или *F[k]* соответственно до четкой видимости наблюдаемого уровня жидкости (снизу вблизи вертикальной оси области интереса черные пиксели вплоть до границы разделения сред), а система в этот момент обновляет матрицу  $mI[k, j]$ ;

- определение масштаба (цены одного пикселя в сантиметрах): поделить реальную высоту объекта интереса  $h[k, j]$  на количество строк в матрице объекта интереса  $mI[k, j]$ , полученной (полученных) с *Ffile* или *F[k]* соответственно;

- задание пользователем типа емкости  $tip[k, j]$  в каждой выделенной области наблюдения (или для всех областей интереса сразу, если они одни и те же во всех этих областях интереса, а системой одновременно отображение на форме в области соответствующей камеры указанного из БД типа емкости  $tip[k, j]$  и сформирование блоков  $B[k, j]$  «Текущий отчет уровня наполнения» с легендой соответствующего цвета;

- задание пользователем верхней грани порога бинаризации  $p2[k, j, v, i]$  и его текущего значения  $p[k, j, v, i]$ , путем ввода в каждой выделенной области наблюдения значения верхней грани порога бинаризации в поле ввода и установки текущего его значения с помощью движка на форме во вкладке «Настройка». А системой одновременно осуществление бинаризации с этим заданным пользователем порогом  $p[k, j, v, i]$  в соответствующей выделенной области наблюдения  $I[k, j]$ ;

- после клика пользователя на запуск наблюдения за областями интереса  $I[k, j]$  (всеми сразу или по отдельности) система осуществляет обновление для каждой области интереса следующих переменных:

- текущий уровень наполнения  $Ur[k, j]$ ,
- текущее отклонение от допустимого уровня наполнения блоков  $B[k, j]$ .

Текущий уровень наполнения  $Ur[k, j]$  находится системой (в сантиметрах);

- если отклонение этого уровня от требуемого уровня наполнения превышает допустимое отклонение, то системой осуществляется звуковая и видеосигнализация об этой ситуации с одновременной записью в базу критических отклонений (дата, время, камера, область интереса, отклонение от нормы), а затем вывод сообщения на форму о текущих отклонениях от требуемого уровня для каждого из объектов наблюдения.

Например, в Mathematica 11 указанное выше звуковое сопровождение можно организовать с помощью кода:

```
Dynamic@If[Ur[k, j] != " ",
  If[Abs[turbd[k, j, tip[k, j]] - Ur[k, j]] > d[k, j, tip[k, j]],
  EmitSound[Play[Sum[Sin[2000 2^t n t], {n, 5}], {t, 0, 4}]]];
]
```

а вывод на форму сообщения запрограммировать следующим образом:

```
Column[{"Сообщение системы", ,
soobsh[k,j] =
Dynamic@If[Ur[k,j] != " ",
Dynamic@Piecewise[{
{
InputField["Отклонение от нормы больше допустимого", String,
Appearance -> "Frameless", ImageSize -> 200],
Abs[turbd[k,j,tip[k,j]] - Ur[k,j]] > d[k,j,tip[k,j]]
},
{
InputField["Уровень наполнения в норме", String,
Appearance -> "Frameless", ImageSize -> 250],
Abs[turbd[k,j,tip[k,j]] - Ur[k,j]] <= d[k,j,tip[k,j]]
}
}]]
```

Здесь  $tip[k, j]$  – тип указанной пользователем емкости (или байпасного уровнемера) для  $j$ -го объекта интереса, находящегося в обзоре  $k$ -й камеры,  $d[k, j, tip[k, j]]$  – допустимое отклонение от среднего уровня, которое считано из базы данных для указанной пользователем емкости (байпасного уровнемера) для  $j$ -го объекта интереса, находящегося в обзоре  $k$ -й камеры. В программе  $Ur[k, j] = "$  " по умолчанию, т.е. в том случае, когда пользователем не указан объект интереса или не выбрана камера.

Заметим, что в коде мы запоминаем текущее сообщение системы в виде значения функции  $soobsh[k, j]$ . Это сделано для того, чтобы в дальнейшем можно было записывать в протоколы наблюдений ситуации превышения допустимых отклонений.

– если отклонение этого уровня (в сантиметрах) от требуемого уровня наполнения (в сантиметрах) в норме, то системой осуществляется только вывод этого сообщения на форму и вывод информации о текущих отклонениях от требуемого уровня для каждого из объектов наблюдения;

– пользователь может остановить наблюдение в выбранной им области интереса  $I[k, j]$  или везде в обзоре камеры или всех камер. Система тогда отметит эту область как область с остановленным наблюдением и не будет рассчитывать информацию для этих областей;

– пользователь может так же запустить снова наблюдение в остановленных ранее местах.

### Заключение

В настоящей работе спроектирована СППР «Контроль уровня жидкости». Эта

система поддержки принятия решения предназначена для контроля текущего уровня жидкости с помощью удаленного видеонализа с нескольких видеокамер. Причем с видеопотока, поступающего с одной камеры, можно следить за уровнями жидкости в разных указанных пользователем областях интереса, находящихся в обзоре выбранной камеры. Система спроектирована с учетом встройки ее в качестве модуля в прикладной раздел автоматизированной системы, разрабатываемой автором совместно с М.Д. Миссаровым, Е.П. Шустовой и А.Ю. Соловяненко [1–3].

### Список литературы

1. Миссаров М.Д., Шустова Е.П., Шустова К.П. Интерфейс системы «Обработка и анализ изображений и видеопотоков» // Современные наукоемкие технологии. 2020. № 6–1. С. 67–73. DOI 10.17513/snt.38073.
2. Shustova E.P. Design principles and architecture of the system «Processing and analysis of images and video streams». Journal of Physics: Conference Series. 2019, Vol. 1158(4):042017. [Electronic resource]. URL: <https://iopscience.iop.org/issue/1742-6596/1158/4>, <https://iopscience.iop.org/article/10.1088/1742-6596/1158/4/042017> (date of access: 20.05.2021). DOI: 10.1088/1742-6596/1158/4/042017.
3. Соловяненко А.Ю. Система тестирования качества работы методов обработки и анализа изображений и видеопотоков // Материалы Международной молодежной научной конференции «XXIV Туполевские чтения» (Казань, 7–8 ноября 2019 г.). Казань: Изд-во ИП Сагиева А.Р., 2019. Т. 4. С. 417–423. [Электронный ресурс]. URL: <https://t4.kai.ru/material/24>, [https://t4.kai.ru/images/digest\\_2019/4.pdf](https://t4.kai.ru/images/digest_2019/4.pdf) (дата обращения: 20.05.2021).
4. Официальный сайт компании «РусАвтоматизация Эксперт в датчиках уровня». [Электронный ресурс]. URL: [https://rusautomation.ru/datchiki\\_urovnya/nivoflip](https://rusautomation.ru/datchiki_urovnya/nivoflip) (дата обращения: 20.05.2021).
5. Официальный сайт ООО «ОЛИЛ», страница «Визуальный индикатор уровня жидкости серии VR». [Электронный ресурс]. URL: <https://dwyer.ru/catalog/indikator-urovnya/vizualnyj-indikator-urovnya-vr> (дата обращения: 20.05.2021).