# Особенности разработанного имитатора ЖУК

1. Имитируется работа не всей установки ЖУК, а только хранилищ углекислоты.
2. Активные элементы управления:

на мнемосхеме:

* - клапан на общей линии подачи и входные/выходные клапаны хранилищ,
* - переключатели А/М (автомат/ручной),
*  - счетчики отгрузки на хранилищах,

на трендах:

*  - цветовой индикатор в легенде,
* - переключатель режимов отображения,
* - переключатель масштабов оси X,
* - переключатель масштабов оси Y,
* - смещение начала оси X влево,
* - смещение начала оси X вправо,
* - смещение начала оси Y вверх,
* - смещение начала оси Y вниз,
*  - отображение данных в виде таблицы,

в режиме отображения данных в виде таблицы:

*  - возврат к трендам,
* - на первую страницу,
* - на предыдущую страницу,
* - на следующую страницу,
* - на последнюю страницу.

1. Имитатор реализован в виде отдельной html-страницы; протестирован в браузерах Microsoft Internet Explorer 11, Microsoft Edge, Google Chrome для Windows (версия 41) и Safari (iPad 2).
2. При открытии страницы в браузере Microsoft Internet Explorer 11 при определенных его настройках может выводиться предупреждение «Internet Explorer запретил выполнение сценариев и элементов ActiveX на этой странице» – необходимо разрешить заблокированное содержимое.
3. При открытии страницы в браузере Microsoft Edge возможно искажение мнемосхемы, устраняется обновлением страницы.
4. Максимально допустимое масштабирование – 175%. При большем масштабировании возникает рассогласование координат трендов и табличного представления данных.
5. Мнемосхема нарисована в Microsoft Visio с последующим сохранением в формате SVG и дополнительной постобработкой с помощью авторской утилиты.
6. Код имитатора реализован в виде набора локальных объектов JavaScript с использованием фреймворков Knockout.js и jqGrid.js – см. секцию <script>..</script> в коде html-страницы.
7. Использование фреймворка Knockout.js обусловлено его богатыми возможностями биндинга данных и хорошей совместимостью с другими JavaScript- фреймворками. Использование фреймворка jqGrid.js обусловлено простотой его использования для отображения данных в виде таблиц.
8. Для исключения необходимости развертывания дополнительных сервисов:

в коде имитатора серверные технологии доступа к данным АСУ ТП не используются; в рабочей конфигурации взаимодействие JavaScript-кода с web-сервисом легко реализуется, например, с использованием фреймворка jQuery.

1. Для того, чтобы не развертывать приложение в web-сервере, а открывать страницу из файловой системы, используется внедренный в страницу SVG и JavaScript; тем не менее, для корректной работы страницы из-за необходимости подгрузки фреймворков требуется соединение с интернет (в рабочей конфигурации фреймворки могут развертываться в локальном web-сервере и соединение с интернет не нужно).
2. Использование внедренных в страницу SVG и JavaScript исключает возможность их локального кэширования и приводит к незначительному замедлению загрузки страницы.

# Требуется доработка

1. Размер SVG может быть существенно уменьшен, т.к. Microsoft Visio генерирует его неэффективно (в частности, вместо стандартных примитивов SVG типа <circle>, <ellipse>, <line>, <polygon>, <polyline> и <rect> везде генерируется <path>, слишком много вложенных групп и т.д.). Для оптимизации SVG необходимо или подобрать существующую утилиту, или написать свою.
2. В разработанном имитаторе все данные хранятся в памяти, в реальной же системе их нужно брать с сервера. Как следствие, необходимо отработать технологию виртуализации табличных данных.