

**Тихомиров В. А., Александров С. Ю.**  
**V.A.Tikhomirov, S.Y.Aleksandrov**

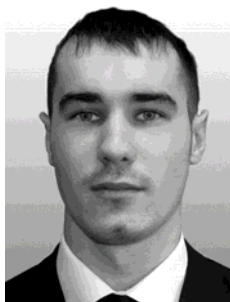
**АВТОМАТИЗАЦИЯ ТРАССИРОВКИ ЭЛЕКТРОЖГУТОВ  
МАШИНОСТРОИТЕЛЬНЫХ ИЗДЕЛИЙ**

**CABLE ROUTING AUTOMATION IN MECHANICAL ENGINEERING PRODUCTS**



**Тихомиров Владимир Александрович** – кандидат технических наук, доцент, заведующий кафедрой математического обеспечения и применения ЭВМ Комсомольского-на-Амуре государственного технического университета (Россия, Комсомольск-на-Амуре). E-mail: kmopevm@knastu.ru.

**Mr. Vladimir A. Tikhomirov** – PhD in Engineering, Associate Professor, Head of the Department of Software and Computing Technology, Komsomolsk-on-Amur State Technical University. E-mail: kmopevm@knastu.ru



**Александров Сергей Юрьевич** – аспирант Комсомольского-на-Амуре государственного технического университета (Россия, Комсомольск-на-Амуре). E-mail: sandrov@mail.ru.

**Mr. Sergey Y. Aleksandrov** – PhD Candidate, Komsomolsk-on-Amur State Technical University. E-mail: sandrov@mail.ru

**Аннотация.** В статье описываются алгоритм и его программная реализация для автоматизации поиска расположения трасс электрожгутов в сложных машиностроительных изделиях. Приводится общая математическая модель алгоритма трассировки электрожгута и нахождения множества клемм, расположенных на искомой трассе. Описывается программный модуль, используемый при тестировании и наладке функционирования электросистем, позволяющий резко сократить временные затраты на выполнение названных операций.

**Summary.** The paper describes an algorithm and its software implementation for automation of electric cable routing in complicated mechanical engineering products. A general model of the algorithm is proposed for cable routing and finding many connection terminals existing on the rout being searched. A software module is described that is used for check-up and maintenance/adjustment of electric systems; the module helps sharply reduce time spent on these operations.

**Ключевые слова:** электрожгут, трасса электрожгута, алгоритм обработки графа, математическая модель, программный модуль.

**Key words:** electric cable, cable routing, graph processing algorithm, mathematical model, software module.

УДК 32.973.26

Изделия специального машиностроения часто содержат множество блоков электрического и электронного оборудования, наладка и испытания которого требуют многократного поиска трасс, соединяющих клеммы этого оборудования [4]. Количество агрегатов в таких изделиях может достигать тысяч единиц, а количество клемм на них – десятки тысяч. Выполнять поиск контролируемых трасс по бумажным электросхемам оборудования – долго и не эффективно. Таким образом, разработка программных средств автоматизации процесса

поиска трасс проводов, соединяющих заданные клеммы заданных агрегатов, является задачей актуальной для повышения эффективности контрольно-наладочных работ в сфере общего и специального машиностроения [3].

### Математическая модель алгоритма поиска трасс электрожгутов

В общем случае раскладка трасс жгутов электрооборудования машиностроительного изделия может быть промоделирована сетевым неориентированным графом (см. рис. 1), в узлах которого находятся клеммы –  $k$ . Клеммы принадлежат определенным агрегатам –  $A$ , так что, с математической точки зрения, можно говорить о некотором множестве агрегатов  $A\{a_i\}$ , где  $i$  – индекс агрегата, пробегает значения от 1 до  $n$  – количества агрегатов в системе изделия.

В свою очередь, каждый агрегат (элемент множества)  $a_i$  содержит множество клемм  $K_i\{k_{ij}\}$ , где  $j$  – индекс клеммы, пробегает значения от 1 до  $m_i$  – количество клемм в агрегате с индексом  $i$ .

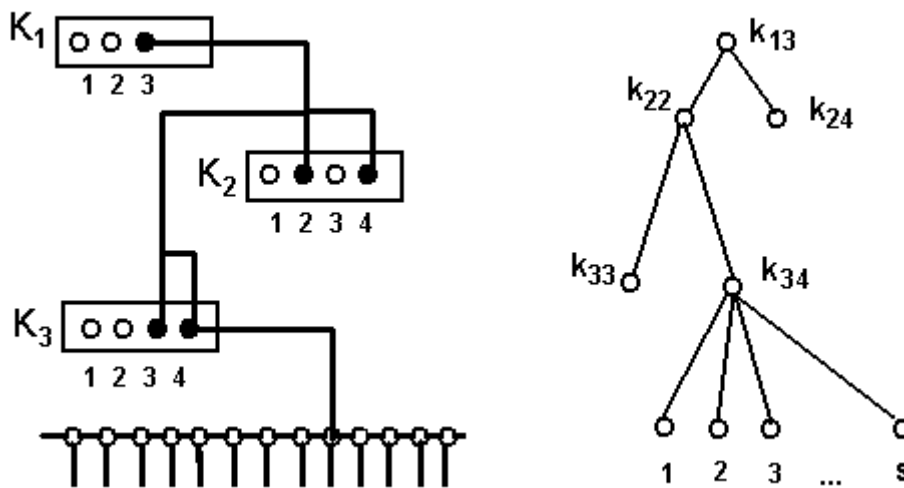


Рис. 1. Схема и граф трассы, соединяющей заданную клемму в электрической цепи изделия

Задача определения трассы, связывающей заданную клемму  $k_{rp}$  с клеммами других агрегатов, может быть записана как поиск бинарного отношения  $T$ , определенного на множестве клемм агрегатов:

$$T_{rp} \subset \prod_{i=1}^n K_i \text{ при } k_{ij} = k_{rp}, \text{ где } 1 \leq r \leq n \text{ и } 1 \leq p \leq m_r \quad (1)$$

и означающего, что во множество  $T$  попадут только те пары множеств  $K_i$ , между элементами которых имеет место указанное отношение. Множество  $T$  будем называть трассой клеммы  $k$ , а пары соединённых клемм в этой трассе – сегментами трассы. В электросистему контролируемого изделия, как правило, входит множество трасс.

Для практического использования выражения (1) при построении алгоритмов поиска трасс электрожгутов требуется дополнительно описание графа соединений (см. рис. 1) в виде некоторой матрицы, определяющей эквивалентность (трассу соединений) элементов множеств. Бинарное отношение (1) может быть задано  $(n, m_{max})$  – матрицей (таблицей), в которой элемент (клемма)  $k_{ij}$ , стоящий на пересечении  $i$ -й строки (номер агрегата) и  $j$ -го столбца (номер клеммы), равен 1, если эта клемма связана с заданной исходной клеммой  $k_{lp}$  искомой трассы, то есть имеет место отношение  $T$ , или 0 в противном случае. Например, для графа, представленного на рис. 1, такая матрица отношения для контакта  $K_{13}$  будет иметь вид:

$$M_{13} = \begin{Bmatrix} 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \end{Bmatrix},$$

а само отношение (трасса), в случае его полного представления, примет вид:

$$T_{13} = \{(k_{13}, k_{22}), (k_{13}, k_{24}), (k_{22}, k_{33}), (k_{22}, k_{34})\}.$$

$m_{max}$  в этом примере (максимальное количество клемм на агрегате изделия) равно 4.

В общем случае описанные рассуждения запишутся уравнением:

$$T_{ij} = \{k \mid M_{ij} = 1\}.$$

Если внимательно посмотреть на схему рис. 1, то явно видно, что рассматриваемая трасса после клеммы  $K_{34}$  тянется дальше, разветвляясь на множество других направлений. Это характерно при выходе трассы, например, на корпус изделия. Очень часто при исследовании трассировки такие ответвления не представляют интереса и должны быть исключены из автоматизированного поиска. Тогда на соответствующую клемму в алгоритме поиска трассы накладывается «блокировка» и считается, что начиная с этой клеммы не может существовать ни одной трассы:

$$T_{bp} \subset \prod_{i=1}^n K_i \equiv \emptyset \text{ если } k_{ij} \in B,$$

где  $B$  – множество заблокированных клемм.

Не редко граф, описывающий структуру контролируемой электросети, носит циклический характер [2] (см. рис. 2).

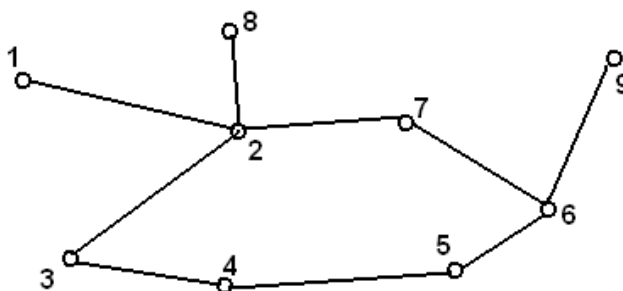


Рис. 2. Циклический граф трассировки электрожгута

Чтобы алгоритм автоматизированного поиска трасс не зациклился на таких кольцах, в математическую модель алгоритма следует ввести еще одно ограничение: блокирование повторного обхода ребра графа независимо от направления этого обхода:

$$\text{если } T_{rp} \subset K_r \times K_p, \text{ то } T_{pr} \subset K_p \times K_r \equiv \emptyset.$$

Для реализации такой блокировки в ходе выполнения алгоритма поиска трассы электрожгута необходимо формировать множество (массив) клемм, уже обработанных методом (1) и вошедших в искомую трассу. Это множество назовём  $P$ .

В этом случае общая математическая модель алгоритма трассировки электрожгута и нахождения множества клемм, расположенных на искомой трассе, может быть представлена уравнением

$$T_{ij} = \begin{cases} \{k \mid M_{ij} = 1\} \\ \{\emptyset \mid k_{ij} \in B\} \\ \{\emptyset \mid k_{ij} \in P\} \end{cases} \quad (2)$$

### Программная реализация алгоритма трассировки жгутов

Реализация разработанного алгоритма может быть проведена на основе базы данных, хранящей информацию о структуре графа электротрасс контролируемого изделия и множестве заблокированных клемм. В общем случае структура таблиц такой базы данных может иметь вид, представленный на рис. 3.

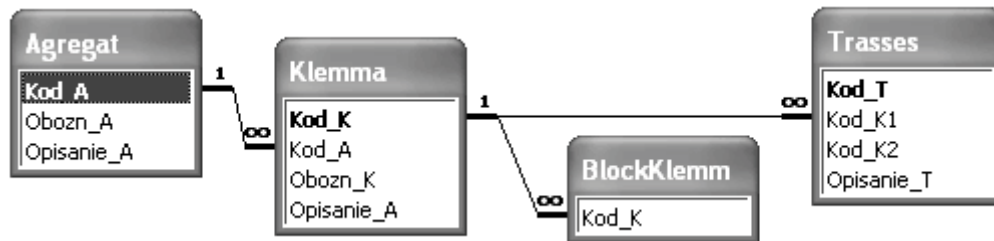


Рис. 3. Общая структура базы данных с информацией для трассировки жгутов

Основная таблица, хранящая информацию о каждом ребре графа электротрасс, – Trasses. Ребра описываются путём задания их концевых клемм. Вторая информационная таблица – BlockKlemm – хранит данные множества В, описанного выше. В полях таблиц с префиксом Opisanie\_ по необходимости может располагаться дополнительная информация для имеющихся агрегатов (таблица Agregat), клемм (таблица Klemma) и трасс (таблица Trasses).

Ядром разработанного программного модуля автоматизации поиска трасс электрожгутов по заданной клемме является процедура SearchTrasses, реализующая описанный выше алгоритм поиска через модуль. Метод алгоритма – рекурсивный обход графа с поиском в глубину. От известных подобных алгоритмов [1] он отличается наличием обработки заблокированных узлов. Графическое представление алгоритма работы процедуры SearchTrasses показано на рис. 4.

При обходе в глубину мы посещаем первый узел (клемму), а затем идем вдоль ребер графа, пока не упрямся в тупик. Узел неориентированного графа является тупиком, если мы уже посетили все примыкающие к нему узлы либо если узел является заблокированным.

После попадания в тупик мы возвращаемся назад вдоль пройденного пути, пока не обнаружим вершину, у которой есть еще не посещенный соседний узел, и двигаемся в этом новом направлении. Процесс оказывается завершенным, когда мы вернулись в отправную точку, а все примыкающие к ней вершины уже оказались посещенными.

Реализацию вычисления отношения  $T$  (формула (2)) лучше всего выполнять SQL запросом. Для рассматриваемой задачи такой запрос имеет вид:

```

SELECT Agregat.Obozn_A, Klemma.Obozn_K, Trasses.Kod_K1, Trasses.Kod_K2
FROM Trasses, Agregat INNER JOIN Klemma ON Agregat.Kod_A = Klemma.Kod_A
WHERE (((Agregat.Obozn_A)="<ОбознАгрегата>") AND
(((Klemma.Obozn_K)="<ОбознКлеммы>") AND ((Trasses.Kod_K1)=[Kod_K])) OR
((Trasses.Kod_K2)=[Kod_K]))
    
```

При каждом вызове этот запрос собирает подмножество рёбер искомой трассы для заданных в полях <ОбознАгрегата> и <ОбознКлеммы> значениях. Повторяя этот запрос в цикле для всех концевых клемм сегментов трассы, найденных на предыдущем запросе, процедура формирует полное отношение, определяющее величину  $T$  по формуле (2).

Интерфейс разработанного программного модуля автоматизированной трассировки электрожгутов представлен на рис. 5.

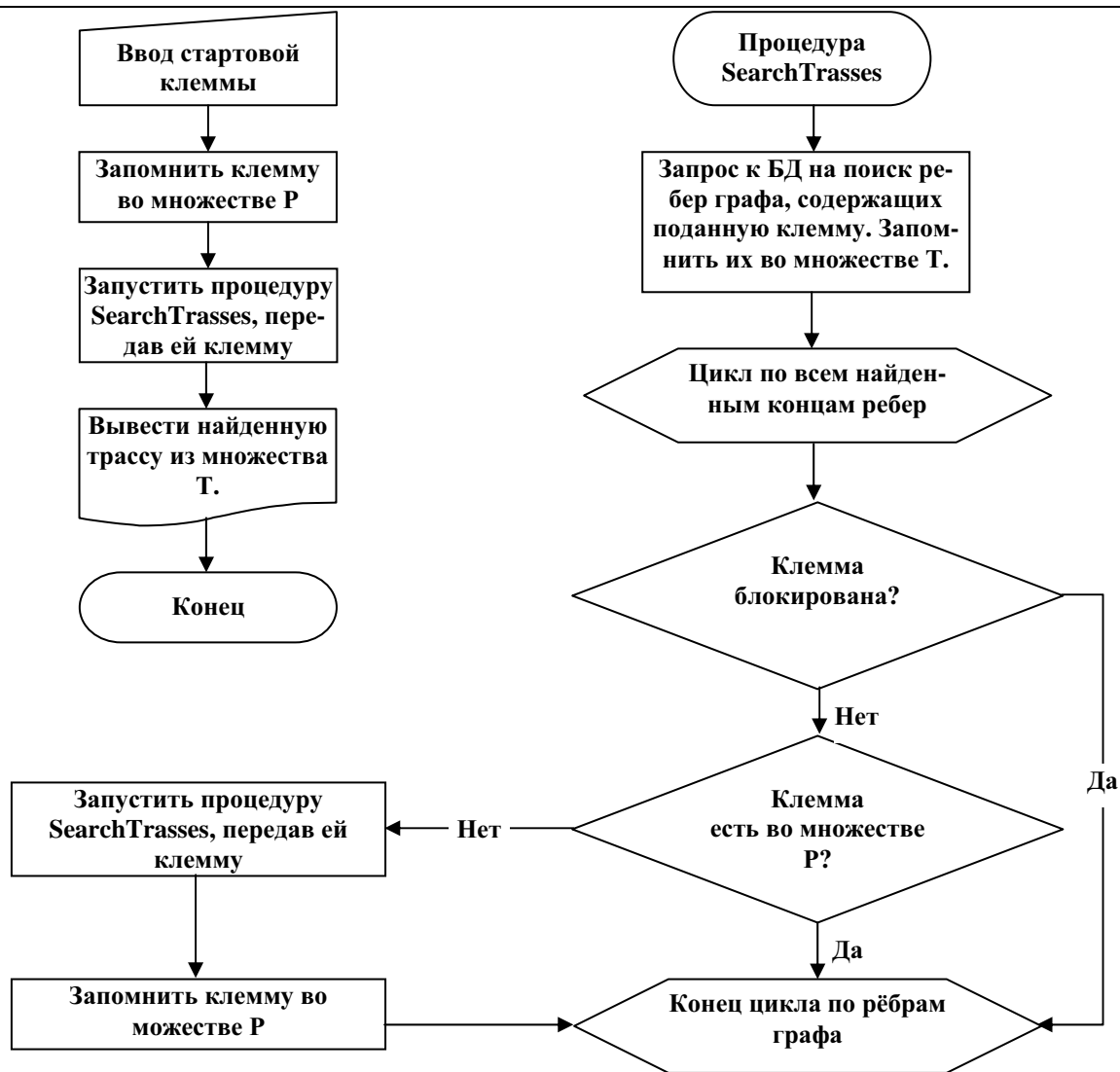


Рис. 4. Алгоритм работы модуля автоматизированного поиска электротрасс

Программный модуль поставляется в виде одного исполняемого файла (Trasses14\_13.exe) и прилагаемого к нему файла базы данных с таблицами множеств агрегатов, клемм и заблокированных клемм. Модуль разработан в среде C++ Builder 2009. Дизайн интерфейса выполнялся с использованием бесплатного набора компонентов AlphaControls v.10.1 [5]. При запуске исполняемого файла происходит автоматический поиск и подключение базы данных и открывается главное окно программы (см. рис. 5).

В полях списков 1, 2 и 3 (см. рис. 5) пользователь имеет возможность указать интересующие его, соответственно, номер электросхемы, обозначение агрегата на этой схеме и номер клеммы на агрегате для поиска трассы, в которую входит указанная клемма. Поиск основан на вышеописанном алгоритме и запускается при двойном щелчке мышью на выбранной клемме списка 3. В результате работы алгоритма в строковый компонент 4 выводится таблица рёбер графа найденной трассы (множество  $T$ ), а в окне 5 организуется вывод графического представления этой таблицы в виде прямоугольников с наименованиями найденных по трассе агрегатов и их клемм, соединенных проводниками (линиями) со связанными клеммами.

Для более тщательного изучения созданного изображения имеется виртуальная увеличительная лупа 6, перемещаемая по экрану курсором мыши. Имеется возможность распечатки результатов поиска заданных трасс как в одиночном режиме, так и в групповом, когда

на распечатку направляется несколько найденных трасс по количеству заданных на поиск клемм.

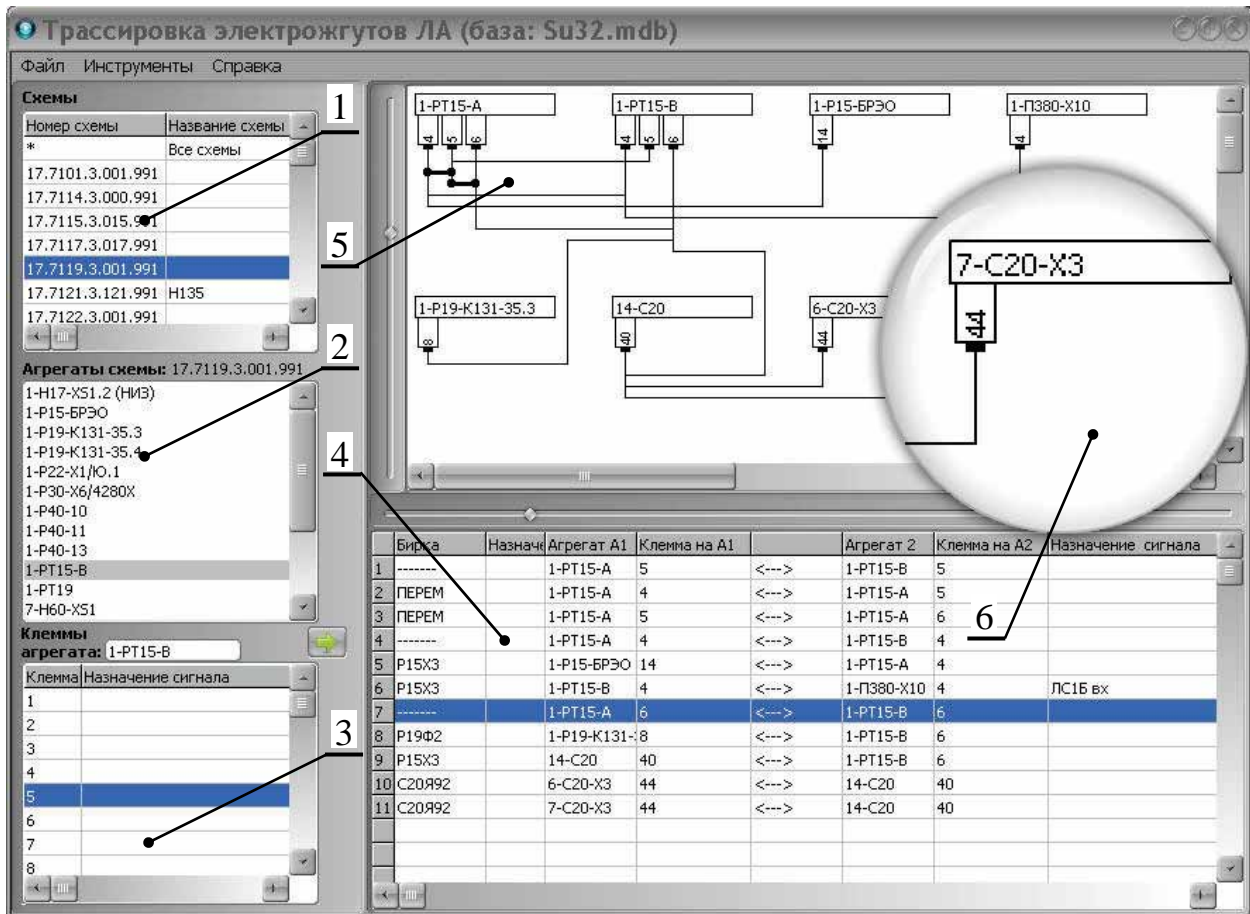


Рис. 5. Главное окно программного модуля автоматизированного поиска трасс электрожгутов

При наведении мыши на любой из прорисованных на экране сегментов найденной трассы в таблице сегментов 4 подсвечивается соответствующая строка с обозначениями оконечных клемм и агрегатов. И наоборот, при щелчке мышью на любой строке таблицы сегментов в графическом окне на экране подсвечивается соответствующий сегмент трассы.

Кроме того, при щелчке правой клавишей мыши на любом из прорисованных в графическом окне 5 агрегатов выводится дополнительная таблица (см. рис. 6) с информацией о активированном агрегате, взятая из полей базы данных с префиксом *Opisani\_*.

Скорость работы алгоритма высокая и составляет доли секунды даже при нескольких десятках тысяч записей в базе данных.

Представленная разработка внедрена в производство и используется при наладке, контроле и испытании бортовых электросистем изделий специального машиностроения.

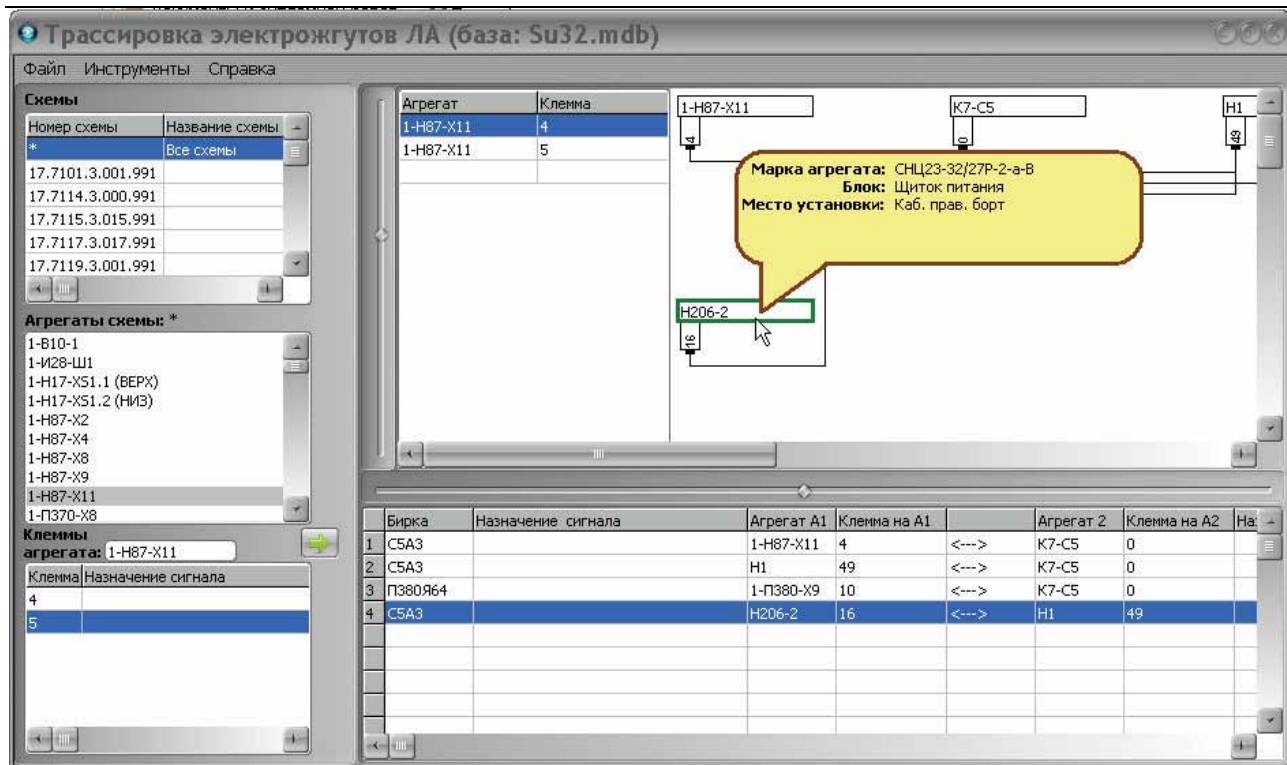


Рис. 6. Дополнительные информационные элементы программы

**ЛИТЕРАТУРА**

1. Алгоритмы обхода графов / Copyright © 2004-2005 «AlgoLib.narod.ru». URL: [http:// AlgoLib.narod.ru](http://AlgoLib.narod.ru). (дата обращения: 13.06.2011).
2. Бурдонов, И. Б. Неизбыточные алгоритмы обхода ориентированных графов. Детерминированный случай / И. Б. Бурдонов, А. С. Косачев, В. В. Кулямин // Программирование. – 2010. – № 4. – С. 13-19.
3. Бурдонов, И. Б. Использование конечных автоматов для тестирования программ / И. Б. Бурдонов, А. С. Косачев, В. В. Кулямин // Программирование. – 2000. – № 2. – С. 18-22.
4. Albers, S. Exploring Unknown Environments / S. Albers and M.R. Henzinger // SIAM J. Comput. – 2000. – № 4. – Vol. 29. – pp. 1164-1188.
5. Набор стандартных и некоторых уникальных компонентов AlphaControls, поддерживающих скинны. URL: [http://www.alphaskins.com/index\\_rus.php](http://www.alphaskins.com/index_rus.php) (дата обращения: 12.06.2011).