

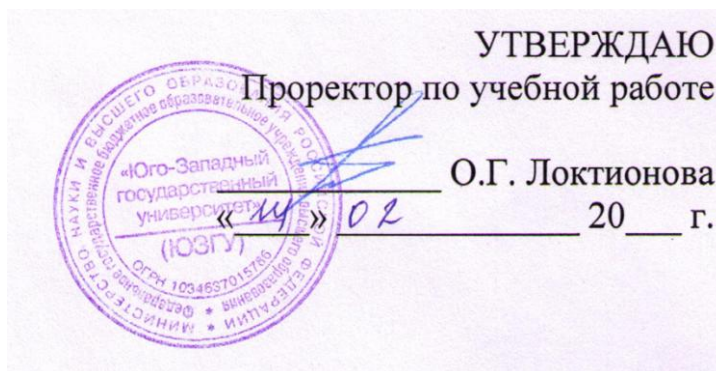
Документ подписан простой электронной подписью  
Информация о владельце:  
ФИО: Локтионова Оксана Геннадьевна  
Должность: проректор по учебной работе  
Дата подписания: 16.08.2020  
Уникальный программный ключ:  
0b817ca911e6668abb13a5d426d39e5f1c11eabbf75e943df4a4851fda56d089

## МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение высшего образования

«Юго-Западный государственный университет»  
(ЮЗГУ)

Кафедра программной инженерии



### ВЫЧИСЛИТЕЛЬНЫЕ СХЕМЫ РАЗРЕШЕНИЯ КОНФЛИКТОВ:

методические указания к лабораторным занятиям для студентов  
направления подготовки 02.03.03 Математическое обеспечение и  
администрирование информационных систем

Курс 20\_\_

УДК 004

Составитель: Е.А. Титенко

Рецензент

Кандидат технических наук А. В. Киселев

**Вычислительные схемы разрешения конфликтов:**  
методические указания к лабораторным занятиям / Минобрнауки  
России, Юго-Зап. гос. ун-т; сост.: Е.А. Титенко. - Курск, 2022. - 12 с. -  
Библиогр.: с. 17.

Приводится описание решетчатых вычислительных схем выбора приоритетного входа из входного конфликтного набора. Решетчатые вычислительные схемы используются в экспертных системах при планировании вычислений и выборе лучшей альтернативы. Приведены теоретические положения построения регулярной вычислительной структуры, имеющей память настройки, рассмотрено решение прямой и обратной задачи разрешения конфликт, приведены примеры и задания.

Методические рекомендации предназначены для студентов, обучающихся по направлению подготовки 02.03.03 Математическое обеспечение и администрирование информационных систем очной и заочной форм обучения.

Текст печатается в авторской редакции.

Подписано в печать

. Формат 60x84 1/16.

Усл.печ. л. 0,70 п.л. Уч.-изд. л. 0,63. Тираж 100 экз. Заказ.

Бесплатно.

Юго-Западный государственный университет.

305040, г. Курск, ул. 50 лет Октября, 94.

**Цель лабораторной работы:** Изучение основных принципов работы программируемых вычислительных схем разрешения конфликтов (настраиваемых арбитров), приобретение практических навыков управления системой приоритетов элементарных конфликтных ячеек.

## 1. Основные понятия

Работа современных экспертных систем опирается на многозадачный режим планирования вычислений, диспетчеризации и выполнения поисковых процессов в соответствии с определенными стратегиями их реализации. Экспертная система на основе принятой модели обработки знаний реализует в пространстве поиска выбор или отбор лучшего направления и проводит построение графа состояний приоритетным способом, т.е. выделяя наиболее перспективную альтернативу. В дальнейшем при сохранении входного набора вычислительная схема разрешения конфликтов может выделить второй, третий приоритет и т.д.

В связи с этим системный процесс в работе экспертных систем, состоящий в динамическом управлении доступом к неразделяемым ресурсам со стороны одновременно выполняющихся задач или потоков. Для разрешения возникающих конфликтов за общий неразделяемый ресурс вводятся вычислительные схемы разрешения конфликтов, которые осуществляют обработку поступающих запросов от конкурирующих альтернатив (рабочих процессов).

Вычислительная схема разрешения конфликта содержит  $k$ -битовый вектор запросов и выдает на выход  $k$ -разрядный вектор подтверждений с единственной логической единицей, соответствующей приоритетному входу.

Одним из возможных путей повышения гибкости арбитра является программная настройка приоритетов. Экспертная система в режиме планирования меняет очередность приоритетов по определенному алгоритму. Поэтому низкоприоритетные каналы периодически становятся высокоприоритетными, и наоборот. При хаотическом «перемешивании» приоритетов рабочих процессов все они оказываются в равных условиях. При целенаправленном перемешивании приоритетность в целом сохраняется.

Среди арбитров наибольший интерес представляют арбитры с изменяемым законом приоритетов запросов. Изменяемый закон

необходим для реализации рабочих процессов при построении графа состояний.

Отличительная особенность программируемых решетчатых арбитров заключается в том, что закон арбитража зависит не только от количества запросов, но и от структурного соотношения между ними, что делает их применение в экспертных системах оправданным. Вторая особенность – описание множества конфликтов через параллельно-последовательное соединение элементарных ячеек, разрешающих парный конфликт между  $i$ -ым,  $j$ -ым запросами ( $i \in 1 \dots N, j \in 1 \dots N, N=8$ ).

На рисунках 1,2,3 представлены схемы решетчатых (программируемых) арбитров, реализующих 128 различных приоритетных соотношений в 8-разрядном векторе запросов в зависимости от 28-разрядного кода настраиваемого регистра приоритетов  $Q$  (память приоритетов). Каждому элементарной ячейке арбитра соответствует один разряд  $Q_i$  программно-доступного регистра приоритетов (на рис. 1, 2, 3 не показан). При отсутствии конфликтов элементарная ячейка транслирует сигналы  $a$  и  $b$  на выходы  $e$  и  $d$  без изменения. При конфликте элементарная ячейка «руководствуется указаниями» бита управляющего разряда  $Q_i$ . Доказано, что любой конфликтной ситуации на входах арбитра независимо от кода в регистре приоритетов на выходы пройдет только один сигнал. Перестройка структуры приоритетов достигается изменением кода в управляющем регистре  $Q$ .

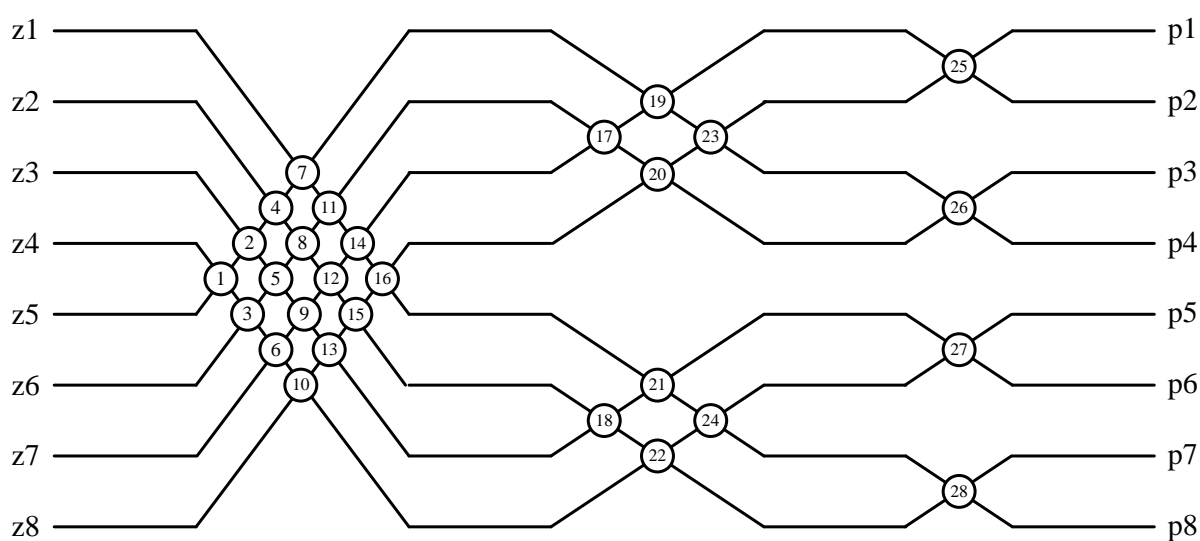


Рис. 1 Структура вычислительной схемы разрешения конфликта №1

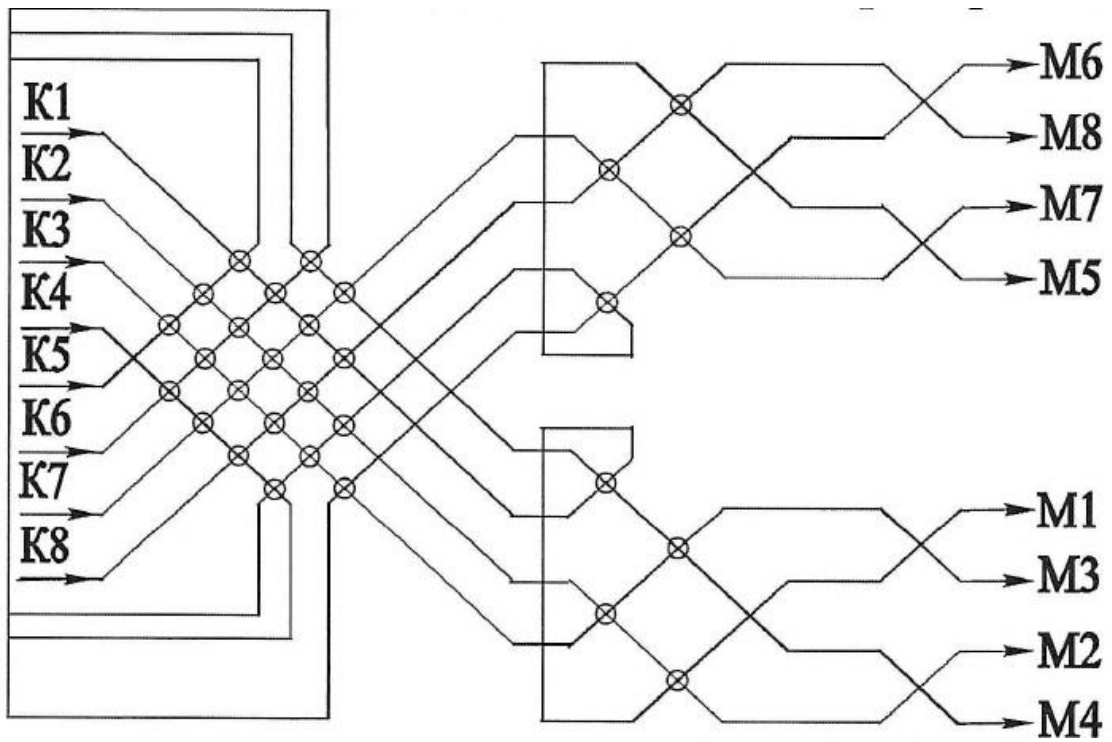


Рис. 2 Структура вычислительной схемы разрешения конфликта

№2

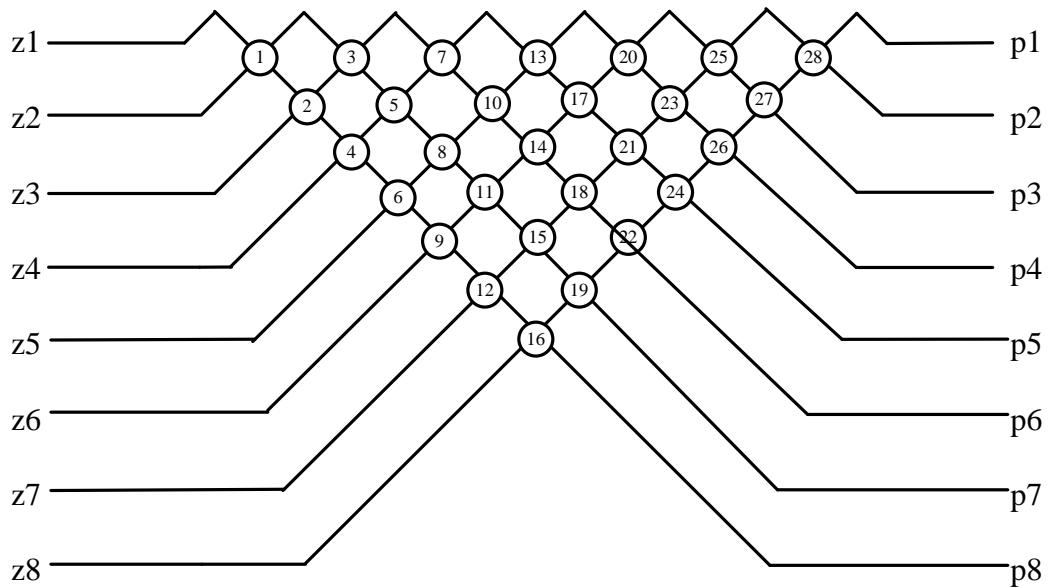


Рис. 3 Структура вычислительной схемы разрешения конфликта

№3

Взаимоотношения между любой парой конкурирующих запросов однозначно определяются битом  $Q_i$  управляющего регистра в соответствии таблицей работы, имеющей два варианта формирования выходов  $c-d$ ,  $c'-d'$  (табл.) элементарной ячейки (рис.4).

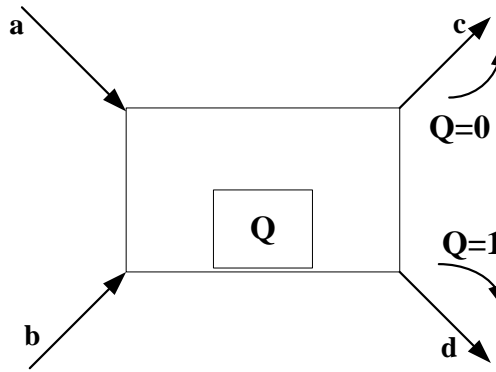


Рис. 4 Схема элементарной конфликтной ячейки

Работа конфликтной ячейки описывается следующей таблицей истинности (таблица 1), в которой бит  $Q_i$  из управляющего регистра разрешает простейший конфликт при  $a=1, b=1$ .

Таблица – Работа конфликтной ячейки

| a | b | $Q_i$ | c | d | $c'$ | $d'$ |
|---|---|-------|---|---|------|------|
| 0 | 0 | 0     | 0 | 0 | 0    | 0    |
| 0 | 0 | 1     | 0 | 0 | 0    | 0    |
| 0 | 1 | 0     | 0 | 1 | 0    | 1    |
| 0 | 1 | 1     | 0 | 1 | 0    | 1    |
| 1 | 0 | 0     | 1 | 0 | 1    | 0    |
| 1 | 0 | 1     | 1 | 0 | 1    | 0    |
| 1 | 1 | 0     | 0 | 1 | 1    | 0    |
| 1 | 1 | 1     | 1 | 0 | 0    | 1    |

Схема разрешения конфликтов устраняет любые конфликты при любом коде в регистре, однако в зависимости от кода он может быть настроен либо на один из жестких, либо на один из гибких режимов.

## 2. Режимы работы схемы разрешения конфликтов

В работе схем разрешения конфликтов выделяют 2 варианта настройки:

- по требуемой системе приоритетов определить значение настраиваемого регистра (прямая задача);
- при заданном значении регистра вычислить соответствующую систему приоритетов (обратная задача).

## Прямая задача

Пусть, например, требуется обеспечить следующую приоритетность запросов для арбитра на рис. 2 (прямая задача)

$K3 \rightarrow K2 \rightarrow K8 \rightarrow K6 \rightarrow K7 \rightarrow K1 \rightarrow K5 \rightarrow K4$ .

Стрелка в такой записи направлена от более приоритетного входа к менее приоритетному. Для обеспечения указанного режима код в управляющем регистре должен настроить узловые схемы следующим образом:

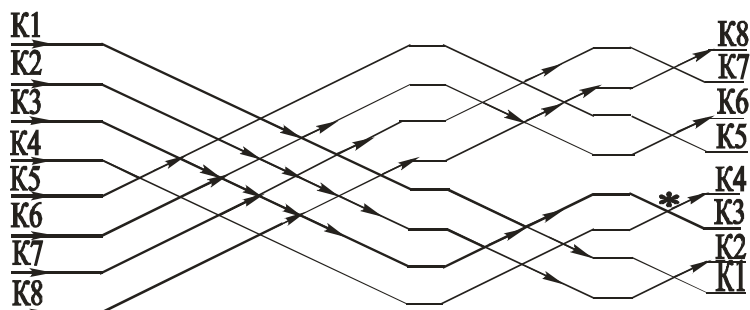
$K3 \rightarrow K2$ ,  $K3 \rightarrow K8$ ,  $K3 \rightarrow K6$ ,  $K3 \rightarrow K7$ ,  $K3 \rightarrow K1$ ,  $K3 \rightarrow K5$ ,  $K3 \rightarrow K4^*$ ,  
 $K2 \rightarrow K8$ ,  $K2 \rightarrow K6$ ,  $K2 \rightarrow K7$ ,  $K2 \rightarrow K1$ ,  $K2 \rightarrow K5$ ,  $K2 \rightarrow K4$ ,  
 $K8 \rightarrow K6$ ,  $K8 \rightarrow K7$ ,  $K8 \rightarrow K1$ ,  $K8 \rightarrow K5$ ,  $K8 \rightarrow K4$ ,  
 $K6 \rightarrow K7$ ,  $K6 \rightarrow K1$ ,  $K6 \rightarrow K5$ ,  $K6 \rightarrow K4$ ,  
 $K7 \rightarrow K1$ ,  $K7 \rightarrow K5$ ,  $K7 \rightarrow K4$ ,  
 $K1 \rightarrow K5$ ,  $K1 \rightarrow K4$ ,  
 $K5 \rightarrow K4$ .

Каждое из этих условий задается значением соответствующего бита в управляющем 28-разрядном регистре. Аналогично можно задать любой жесткий приоритетный порядок между каналами. Число таких режимов равно числу перестановок между номерами каналов  $P_8 = 40\ 320$ . Однако это число не исчерпывает все возможные режимы работы устройства. Действительно, поскольку число управляющих бит в арбитра в данном примере составляет  $C_8 = 28$ , то возможны  $2^{28} = 268\ 435\ 456$  вариантов настройки, которые, за исключением уже рассмотренных, характеризуются гибким распределением приоритетов между каналами.

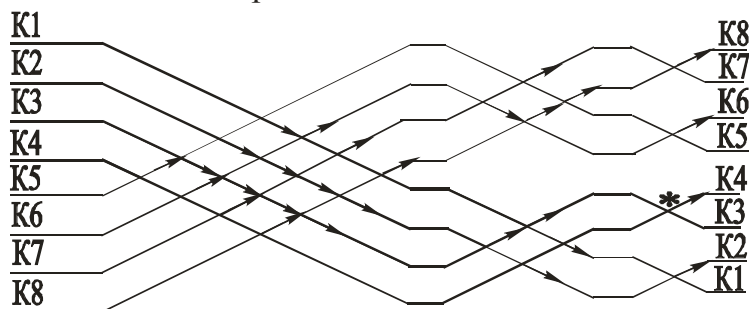
Смысл гибких режимов состоит в том, что приоритет между каналами определяется не только кодом в управляющем регистре, но также зависит от числа поступивших запросов и их распределения по входам арбитра. Если в предыдущем примере изменить значение только одного бита в управляющем коде, а именно бита, определяющего взаимоотношения между входами  $K3$  и  $K4$ , то условие  $K3 \rightarrow K4$ , помеченное ранее знаком  $*$ , изменится на противоположное. Такое изменение приводит к получению одного из гибких режимов.

Предположим, что задана исходная система приоритетов, но  $K4 \rightarrow K3$ . Пусть запросы ступили на все входы  $K1—K8$  и только на входы  $K1—K4$ . Для первого вектора запросов на рис.6 а) полужирными линиями показаны траектории распространения запросных сигналов через арбитра. Из рисунка сигнала видно, что при взаимодействии

запросов по входам K4 и K5 побеждает K5, так как соответствующая узловая схема решетчатой структуры настроена на приоритетную передачу сигнала, поступившего на вход K5. Для вектора запросов, имеющих логические единицы на входах K1—K4, приоритетным будем запрос K3, что отражено на рис.6 б)



при наличии всех  
запросных сигналов



при наличии четырех  
запросных сигналов  
по каналам K1-K4

а) при запросах K1-K8

б) при запросах K1-K4

Рис. 5 Траектории распространения запросов в арбитра.

Другая особенность решетчатых программируемых арбитров состоит в том, схемы являются двунаправленными, что позволяет рассматривать обратные варианты вычислительных схем разрешения арбитров. На рис. 6 и 7 показан вариант обратной вычислительной схемы для N=4 и N=8.

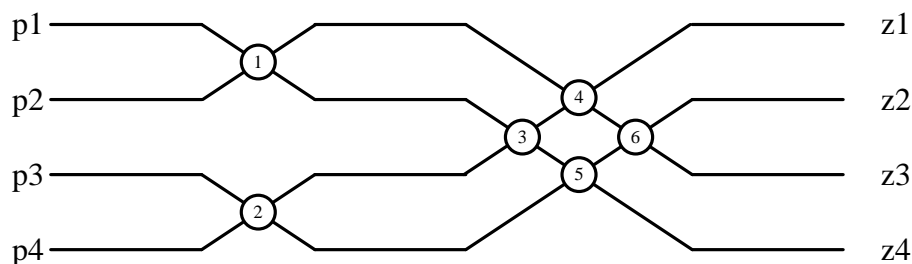




Рис. 6 Обратная схема разрешения конфликтов при N=4

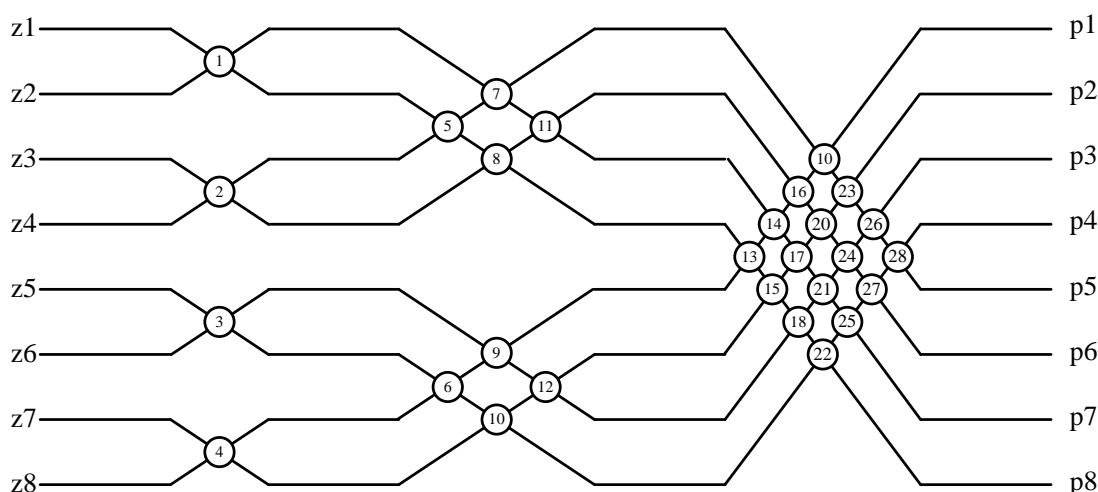


Рис. 7 Обратная схема разрешения конфликтов при N=8

### Обратная задача.

Суть обратной задачи состоит в задании настраиваемого регистра Q и вычислении по нем системы приоритетов. Обратная задача также решается на основе составления пар приоритетов, между i-ым, j-ым запросами ( $i \in 1 \dots N, j \in 1 \dots N$ ). На основе пар приоритетов путем анализа зависимостей между правыми и левыми частями (продукционный подход) составляется линейная система приоритетов или устанавливается факт невозможности построения единственной системы приоритетов.

Эта особенность связана с тем, что в общем случае по произвольному значению регистра Q всегда можно построить циклическую систему приоритетов, но не всегда линейную систему приоритетов.

Пусть вычислительная схема разрешения конфликтов согласно рис.6. Пусть значение настроечного регистра  $Q=010101$ . Пусть последовательность бит в настроечном регистре Q соответствует номерам элементарных конфликтных ячеек: 1, 2, 3, 4, 5, 6.

Далее для каждой конфликтной ячейки на основе таблицы 1 вычисляется пара приоритетов

1) конфликтная ячейка 1, анализируются входы 1 и 2. На основе таблицы 1 при  $q_1=0$  устанавливается пара приоритетов  $1 \rightarrow 2$ .

2) конфликтная ячейка 2, анализируются входы 3 и 4. На основе таблицы 1 при  $q_2=1$  устанавливается пара приоритетов  $4 \rightarrow 3$ .

3) конфликтная ячейка 3, анализируются входы 1 и 4. На основе таблицы 1 при  $q_3=0$  устанавливается пара приоритетов  $1 \rightarrow 4$ .

4) конфликтная ячейка 4, анализируются входы 2 и 4. На основе таблицы 1 при  $q_2=1$  устанавливается пара приоритетов  $4 \rightarrow 2$ .

5) конфликтная ячейка 5, анализируются входы 1 и 3. На основе таблицы 1 при  $q_3=0$  устанавливается пара приоритетов  $1 \rightarrow 3$ .

6) конфликтная ячейка 6, анализируются входы 2 и 3. На основе таблицы 1 при  $q_2=1$  устанавливается пара приоритетов  $3 \rightarrow 2$ .

Для составления цепочки приоритетов пары записываются «столбиком».

$1 \rightarrow 2$

$4 \rightarrow 3$

$1 \rightarrow 4$

$4 \rightarrow 2$

$1 \rightarrow 3$

$3 \rightarrow 2$

Далее из них выделяются вход, который располагается только в левой части пар приоритетов. Это будет вход с максимальным приоритетом. Из пар приоритетов выделяется вход наиболее часто входящий в их правую часть. Это будет вход с минимальным приоритетом.

Для вычислительной схемы разрешения конфликтов (рис.60 и значения настроечного регистра  $Q=010101$  максимальный приоритет для входа 1, минимальный приоритет – вход 2.

Оставшиеся входы 3 и 4 располагаются по числу вхождений в правую часть пар приоритетов: вход 3 справа дважды входит в правую часть, вход 4 один раз входит в правую часть. Значит, для  $Q=010101$  система приоритетов имеет единственный (линейный) вид  $1 \rightarrow 4 \rightarrow 3 \rightarrow 2$ .

Данную систему приоритетов можно восстановить путем формальным способом, разбив «столбик» пар приоритетов на упорядоченные подмножества с одинаковой левой частью:

$1 \rightarrow 2$

$1 \rightarrow 4$

$1 \rightarrow 3$

$4 \rightarrow 3$

$4 \rightarrow 2$

$3 \rightarrow 2$

Теперь единая система приоритетов составляется из левых частей каждого из подмножеств:  $1 \rightarrow 4 \rightarrow 3 \rightarrow 2$

### 3 Индивидуальное задание студента

| № варианта | Тип арбитра | Направление | Q     | Система приоритетов запросов |
|------------|-------------|-------------|-------|------------------------------|
| 1          | 1           | →           | -     | 8,7,1,3,4,2,6,5              |
| 2          | 2           | →           | FOOFh |                              |
| 3          | 3           | →           | 00FAh | -                            |
| 4          | 4           | ←           | 0AA0h | -                            |
| 5          | 1           | ←           | -     | 1,3,5,7,2,4,6,8              |
| 6          | 2           | ←           | 00F0h | -                            |
| 7          | 3           | →           | -     | 5,6,7,8,1,2,3,4,             |
| 8          | 4           | →           | -     | 1,8,2,7,3,6,4,5              |
| 9          | 1           | →           | 0FF0h |                              |
| 10         | 2           | ←           | -     | 2,3,7,8,6,1,5,4              |
| 11         | 3           | ←           | 000Fh | -                            |
| 12         | 4           | ←           | 00CEh | -                            |
| 13         | 1           | →           | FFFFh | -                            |
| 14         | 2           | →           | FFFFh | -                            |
| 15         | 3           | ←           | FF0F  | -                            |
| 16         | 4           | ←           | FF0F  | -                            |

где ←, → - направления входов/выходов в арбитра; «h» – шестнадцатеричный эквивалент значения настроечного регистра Q.

### 4 Контрольные вопросы

1. Назначение процессов разрешения конфликтов в экспертных системах и.
3. Какие недостатки вычислительных схем с жесткой структурой.
4. Какие преимущества вычислительных схем с настраиваемой структурой.
5. Таблица работы элементарной конфликтной ячейки.
6. Назначение, разрядность настроечного регистра.
7. Прямая задача в вычислительных схемах разрешения конфликтов.

8. Обратная задача в вычислительных схемах разрешения конфликтов.

9. Особенности гибкого режима вычислительных схем разрешения конфликтов.

10. Какие нестандартные структуры вычислительных схем разрешения конфликтов.

### **Библиографический список**

1. Сидоркина, И. Г. Системы искусственного интеллекта [Текст] : учебное пособие / И. Г. Сидоркина. - Москва : КНОРУС, 2016. - 246 с.

2. Автоматизированные информационные системы и интеллектуальные технологии [Текст] : учебное пособие / Е. А. Титенко [и др.] ; Минобрнауки России, Федеральное государственное бюджетное образовательное учреждение высшего профессионального образования "Юго-Западный государственный университет". - Курск : ЮЗГУ, 2013. - 133 с.

3. Основы построения интеллектуальных систем [Электронный ресурс]: учебное пособие/ Г.В. Рыбина. М. : Финансы и статистика, 2010. - 432 с.

4. Шевкопляс, Б.В. Микропроцессорные структуры. Инженерные решения: Справочник / Б.В. Шевкопляс - М.: Радио и связь, 2009. - 512 с.