**Задание Ш05.** В пакете прикладных программ *National Instruments LabView* создать виртуальный прибор для решения системы линейных алгебраических уравнений (СЛАУ) матричным методом.

**Общая часть задания:**

1. Выполнить подключение и настройку стандартной функции *Solve Linear Equations* для решения СЛАУ.

2.1. Учащиеся с чётными номерами вариантов реализуют решение СЛАУ методом Крамера при использовании стандартной функции поиска определителя матрицы (*Determinant*).

2.2. Учащиеся с нечётными номерами вариантов реализуют метод решения СЛАУ при использовании стандартной функции поиска обратной матрицы (*Inverse Matrix*).

3.1. Учащиеся с чётными номерами вариантов создают собственную подпрограмму для поиска определителя матрицы, упаковывают её в процедуру и подменяют ею стандартные.

3.2. Учащиеся с нечётными номерами вариантов создают собственную подпрограмму для поиска обратной матрицы, упаковывают её в процедуру и подменяют ею стандартные.

На блок-диаграмме все функциональные, то есть значащие пересечения линий связи (узлы) обозначить точками. Для этого в меню перейти к настройкам «*Tools > Options…*», далее в выпадающем списке перейти к настройкам блок-диаграммы (*Block Diagram*) и выставить галочку напротив пункта «*Show dots at wire junctions*».

По итогам выполнения работы сдаются следующие файлы:

- отчёт, выполненный в текстовом редакторе *Microsoft Office Word* (*\*.doc* или *\*.docx*);

- файл виртуального прибора со стандартными элементами решения СЛАУ *National Instruments LabView* (*\*.vi*);

- файл виртуального прибора со пользовательскими элементами решения СЛАУ *National Instruments LabView* (*\*.vi*);

- файл процедурного виртуального прибора (обратная матрица / определитель) *National Instruments LabView* (*\*.vi*) по вариантам.

Отправленные поодиночке файлы проверке не подлежат. При отсутствии одного из упомянутых файлов зачёт по заданию не выставляется.

**Требования к именам файлов:**

**Общий вид формата имени файла:** «*Дата. Задание. Фамилия.mcdx*»

**Формат записи даты:** «*ГГГГММДД*», где *ГГГГ* – четыре цифры текущего года, *ММ* – две цифры текущего месяца, *ДД* – две цифры текущего дня.

**Формат записи задания:** «Задание *NNk*», где *NN* – две цифры номера задания, *k* – обозначение «о», если файл содержит общую часть; обозначение «и», если файл содержит индивидуальную часть; обозначение «ои», если файл содержит как общую, так и индивидуальную части.

**Если устранить замечания по работе удаётся в тот же день:** после фамилии ставится пробел и в круглых скобках записывается номер попытки исправления.

**Примеры правильных имён файлов, которые сдаются на проверку впервые:**

«*20190329. Задание 05ш. Иванов.docx*»

«*20190329. Задание 05ш. Иванов.vi*»

**Внимание!** Не забудьте выполнить автоматическую нумерацию страниц в отчёте.

Отчёт по выполненной работе должен содержать:

0. Титульный лист.

1. Формулировку цели работы.

2. Описание задачи согласно выданному варианту.

2.1. Общая часть.

2.2. Индивидуальная часть.

3. Составление блок-схемы алгоритма программы.

3.1. Общая часть.

3.2. Индивидуальная часть.

4. Подбор и расчёт тестовых примеров.

4.1. Общая часть.

4.2. Индивидуальная часть.

5. Листинг кода составленного программного обеспечения (блок-диаграммы *LabView*).

5.1. Общая часть.

5.2. Индивидуальная часть.

6. Графический пользовательский интерфейс программного обеспечения (передняя панель виртуального прибора *LabView*) и его описание.

6.1. Общая часть.

6.2. Индивидуальная часть.

7. Расчёт тестовых примеров с использованием составленного программного обеспечения.

7.1. Общая часть.

7.2. Индивидуальная часть.

8. Формулировку вывода о проделанной работе (обезличено – исключить из вывода местоимения, такие как «я», «мы» и другие).

Рекомендации к отчёту, доказывающие самостоятельность выполнения работы и упрощающие процедуру проверки отчёта преподавателем:

1. Выполнение дополнительных скриншотов для случаев, когда текстовое описание проделанных действий становится громоздким или трудным к восприятию.

2. Нумерация рисунков (если есть) с подписями, содержащими названия рисунков, например, «Рисунок 1 – Пользовательский интерфейс *Microsoft Office Excel*».

**Цель работы (одна из возможных формулировок)**: освоение навыков работы с матрицами. Закрепление раздела «Линейная алгебра» высшей математики.

**Памятка программисту:**

В методе Крамера для решения СЛАУ важен определитель исходной (главной) матрицы системы, а также определители модифицированных матриц, полученных в результате подмены соответствующих столбцов столбцом свободных членов. Расположение функции для поиска определителя матрицы в пакете прикладных программ *National Instruments LabView* показано на Рисунке 1.

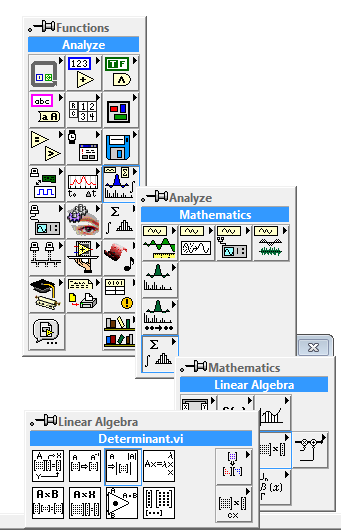


Рисунок 1 – Место расположения стандартного компонента для расчёта определителя матрицы (*Determinant*)

Расположение функции для получения сведений о размерности матрицы в пакете прикладных программ *National Instruments LabView* показано на Рисунке 2.

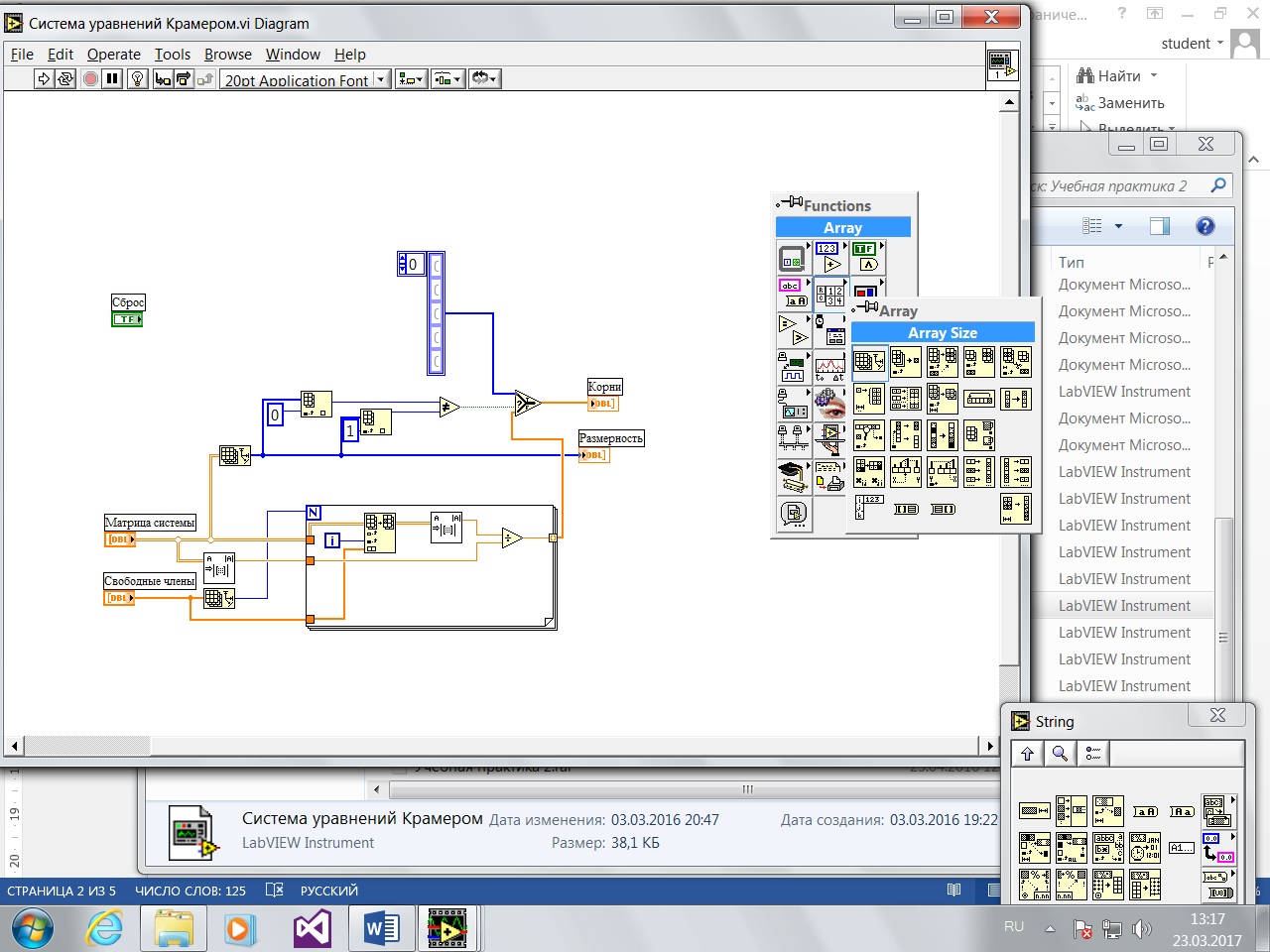


Рисунок 2 – Место расположения функции для получения размерности массива (*Array Size*)

Для одномерных массивов функция напрямую выдаёт целочисленное значение, равное размерности одномерного массива, для *N*-мерного массива функция выдаёт в качестве ответа одномерный массив размерностей, который необходимо разбирать на составляющие (Рисунок 3).

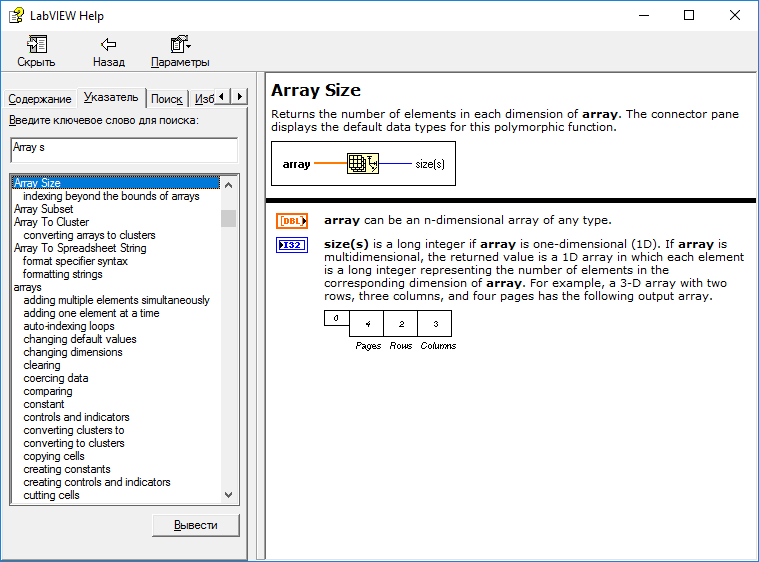


Рисунок 3 – Фрагмент справки пакета прикладных программ *National Instruments LabView* о работе функции для получения размерности массива (*Array Size*)

Пусть даны два листа (две страницы) матриц [3 x 4] (трёхмерный массив) вида:



Для ввода такого массива на передней панели виртуального прибора необходимо перейти в раздел массивов и кластеров, где выбрать массив (*Array*). Соответствующий выбор показан на Рисунке 4.

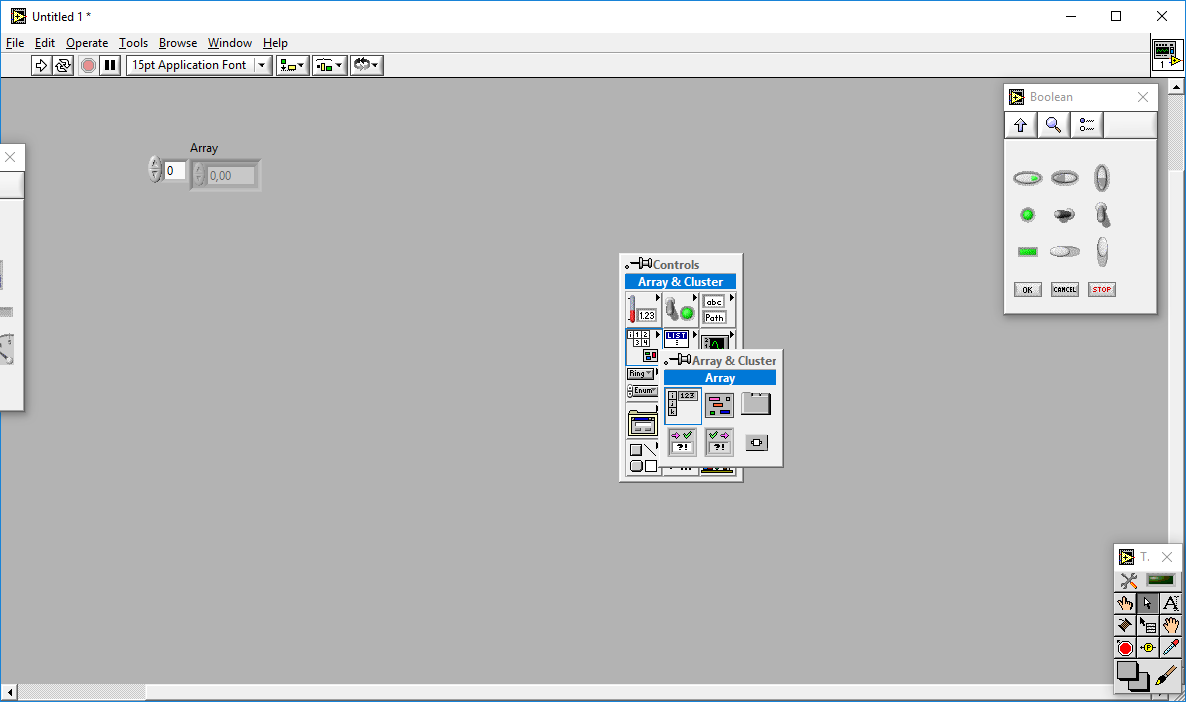


Рисунок 4 – Выбор массива из перечня интерфейсных элементов управления

Размещённый на передней панели нетипизированный массив выглядит следующим образом (Рисунок 5):

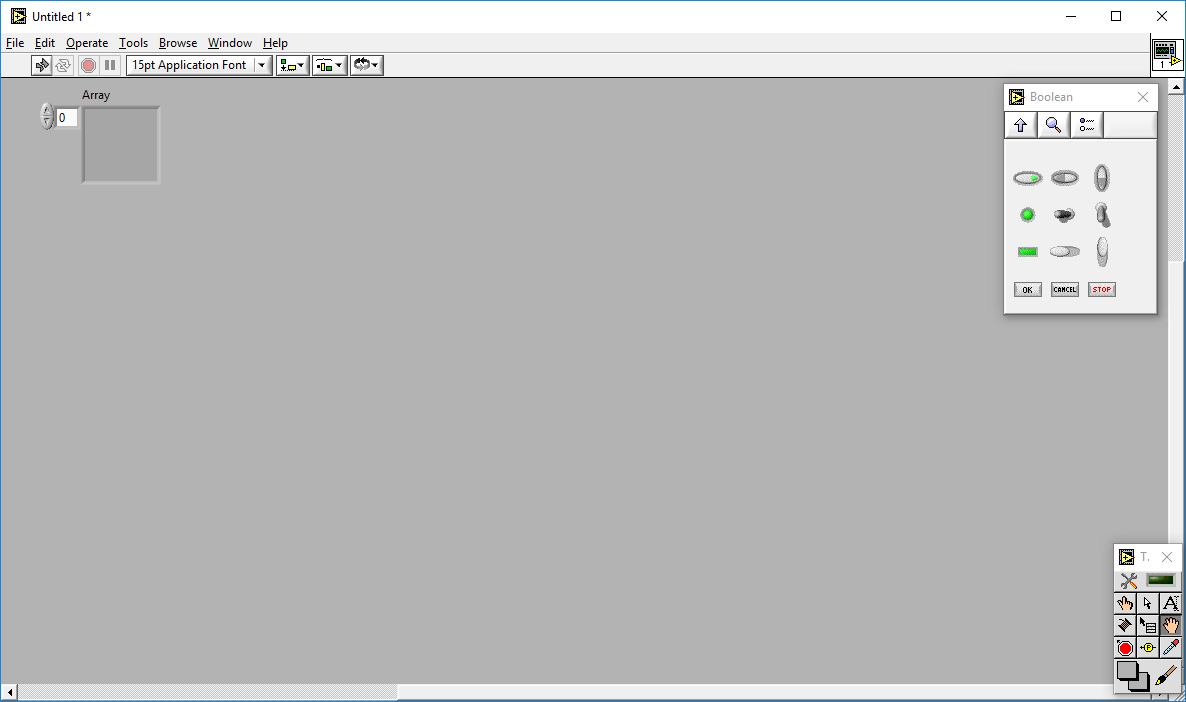


Рисунок 5 – Отображение нетипизированного массива на передней панели виртуального прибора

Далее в утопленную область необходимо перенести числовой контроллер и массив станет вещественным (Рисунок 6).

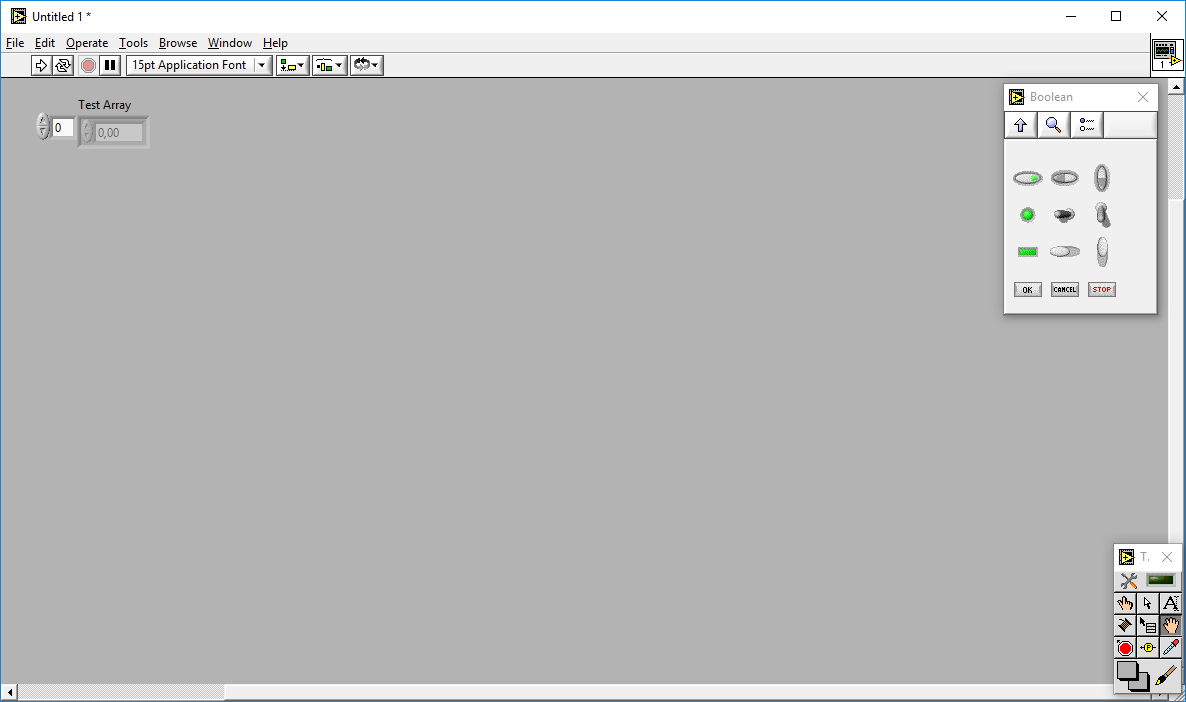


Рисунок 6 – Отображение вещественного массива на передней панели виртуального прибора

В основном примере рассматривается массив целых чисел, потому для удобства последующей работы необходимо изменить представление данных в полученном массиве с вещественного типа на целый (Рисунок 7).

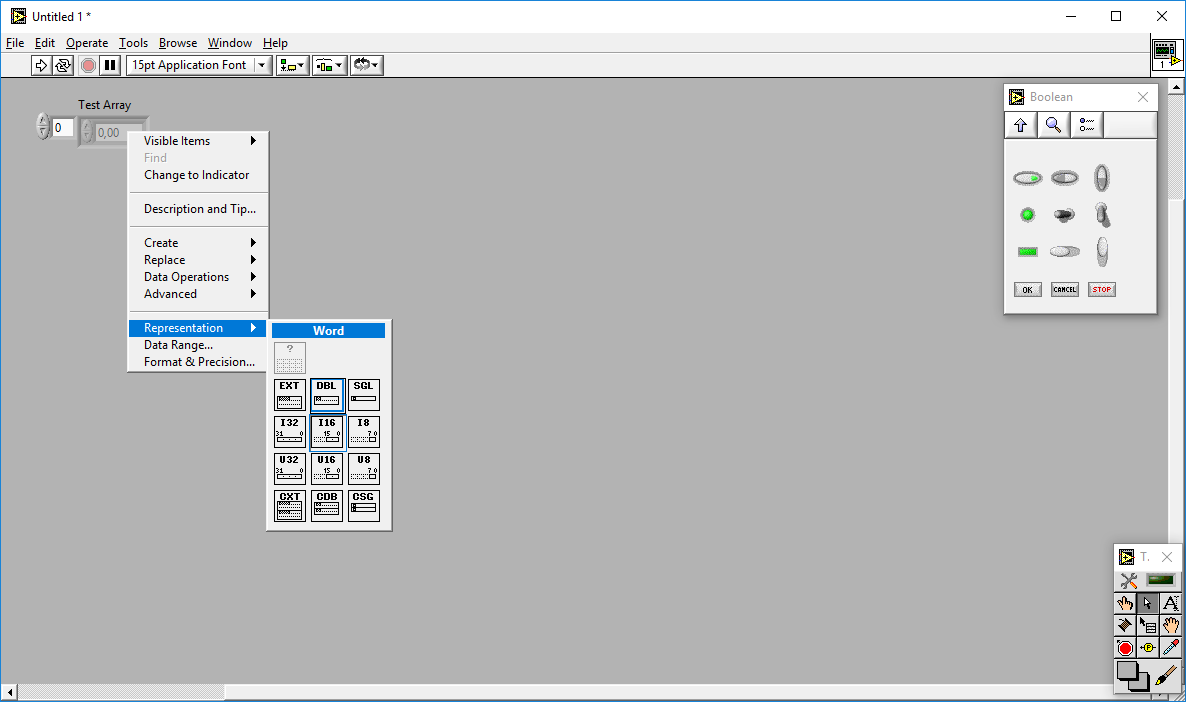


Рисунок 7 – Изменение представления данных в массиве с вещественного типа на целый тип

В рассматриваемом случае на передней панели расположен одномерный массив. Для создания трёхмерного массива, необходимо дважды проделать операцию добавления размерности (Рисунок 8). Эта операция (*Add Dimension*) расположена в контекстном меню того фрагмента массива, который отвечает за сдвиг отображаемых на передней панели элементов относительно нулевого индекса.

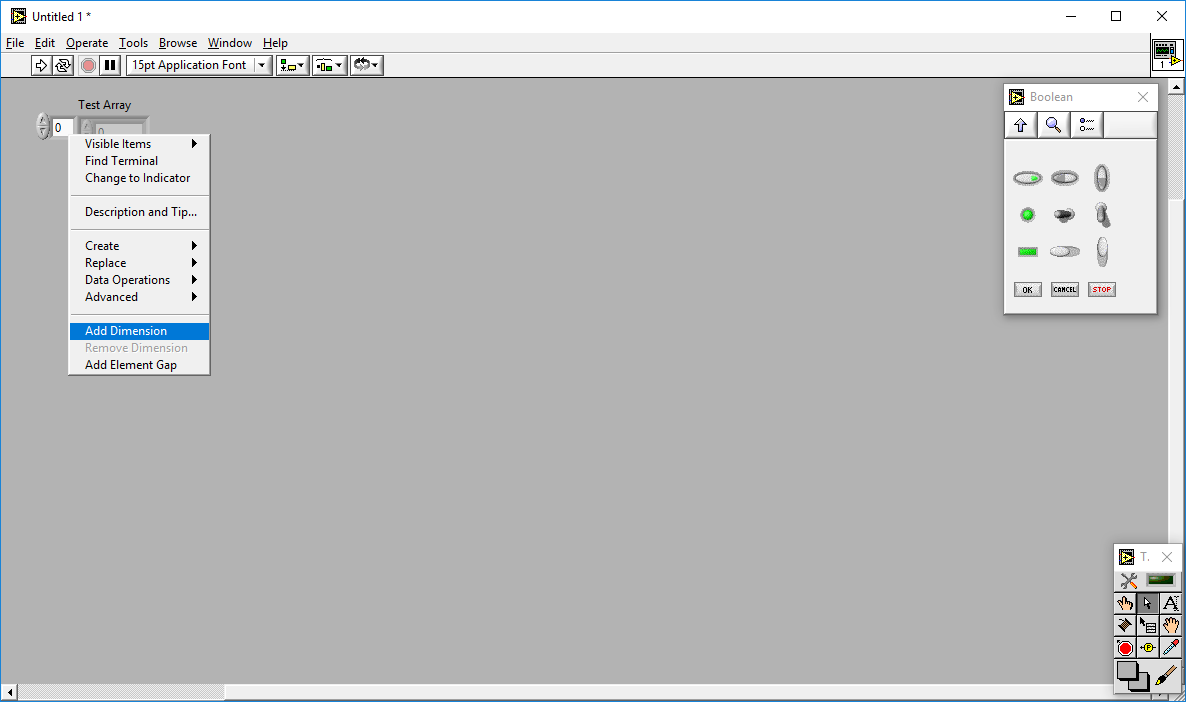


Рисунок 8 – Добавление размерности массива (*Add Dimension*) через контекстное меню элемента управления

После выполнения операции массив будет выглядеть следующим образом (Рисунок 9).

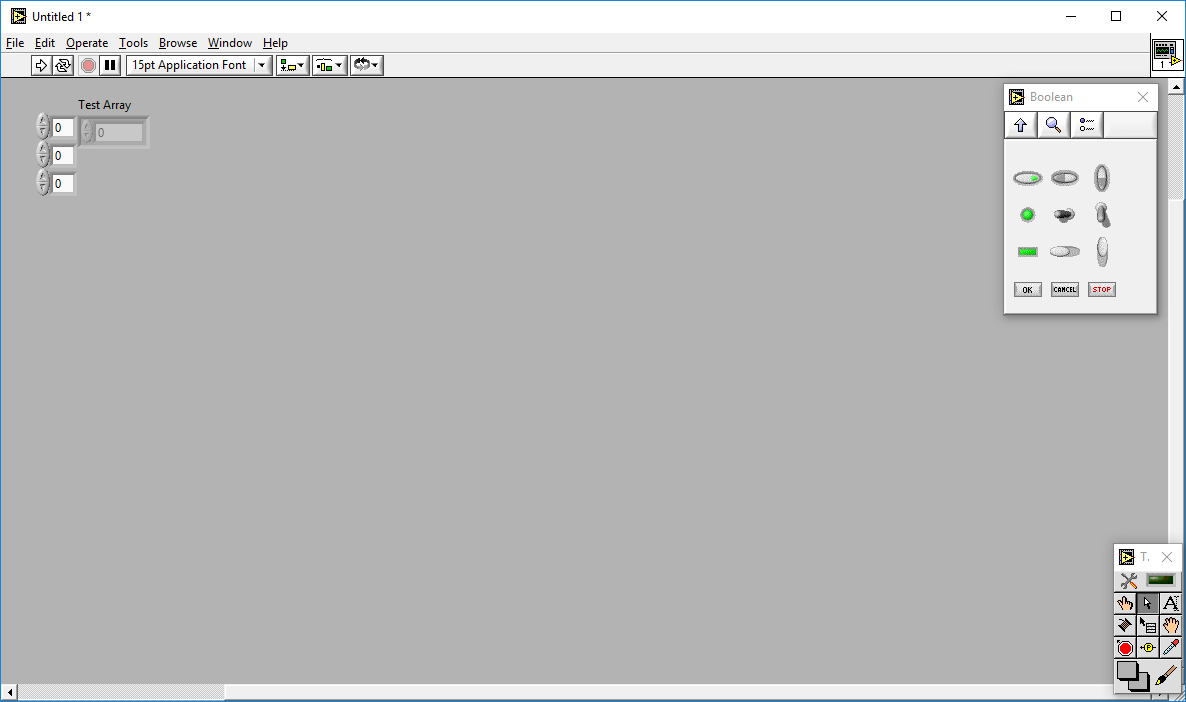


Рисунок 9 – Трёхмерный массив, размещённый на передней панели виртуального прибора

Посредством «вытягивания» интерфейсного элемента управления за правый нижний угол необходимо добиться табличного вида массива, состоящего из трёх строк и четырёх столбцов (Рисунок 10).

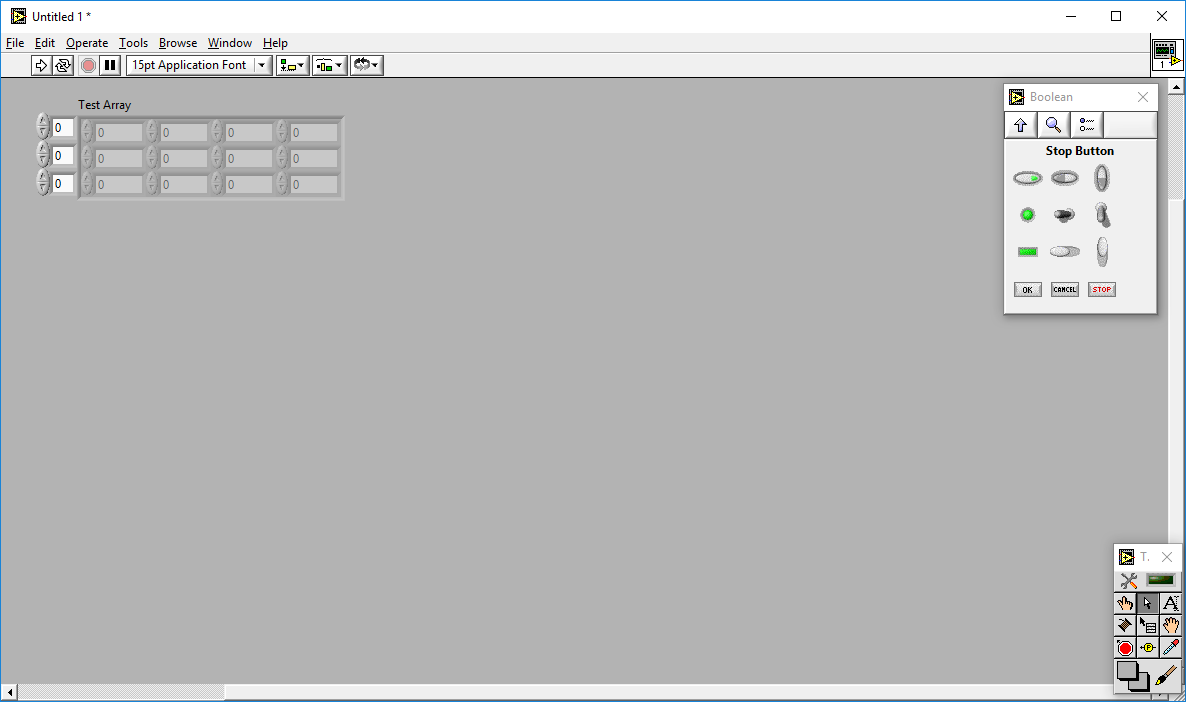


Рисунок 10 – Отображение незаполненного нулевого листа (нулевой страницы) созданного трёхмерного массива, состоящего из трёх строк и четырёх столбцов

Далее отображён нулевой лист массива, который заполнен целочисленными значениями, взятыми в порядке их использования при счёте (Рисунок 11).

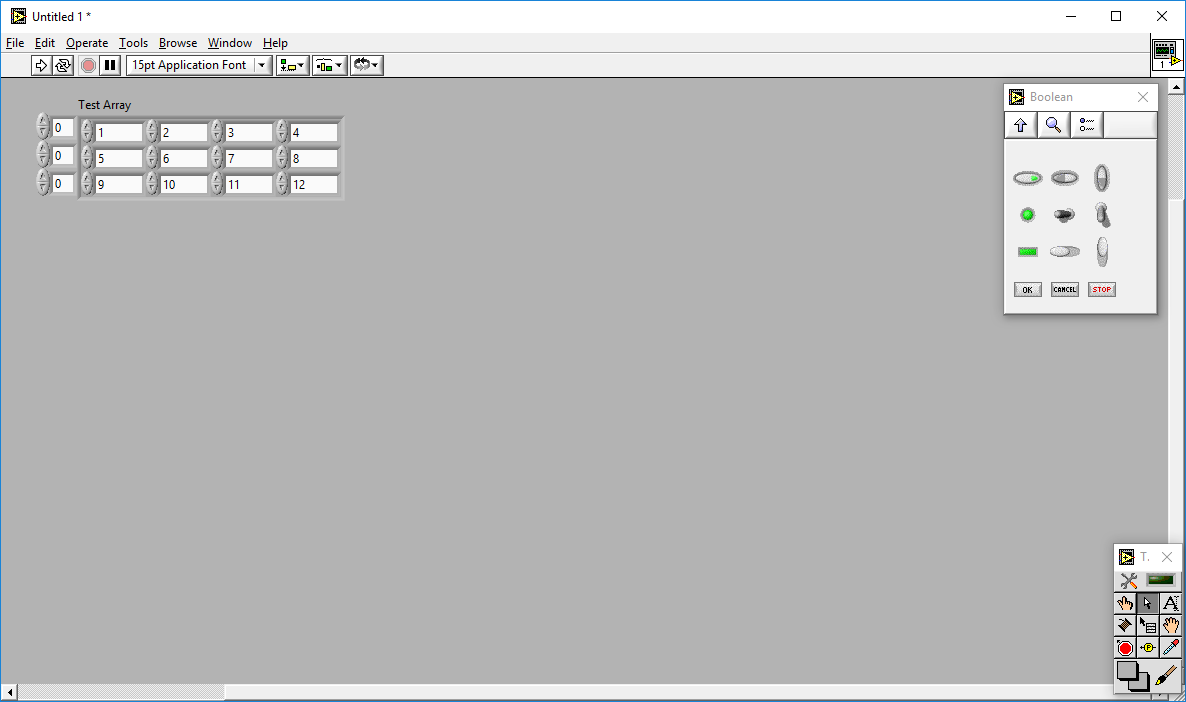


Рисунок 11 – Отображение заполненного нулевого листа (нулевой страницы) созданного трёхмерного массива

Для перехода к заполнению следующего листа массива необходимо выполнить сдвиг первой из представленных размерностей на единицу (Рисунок 12).

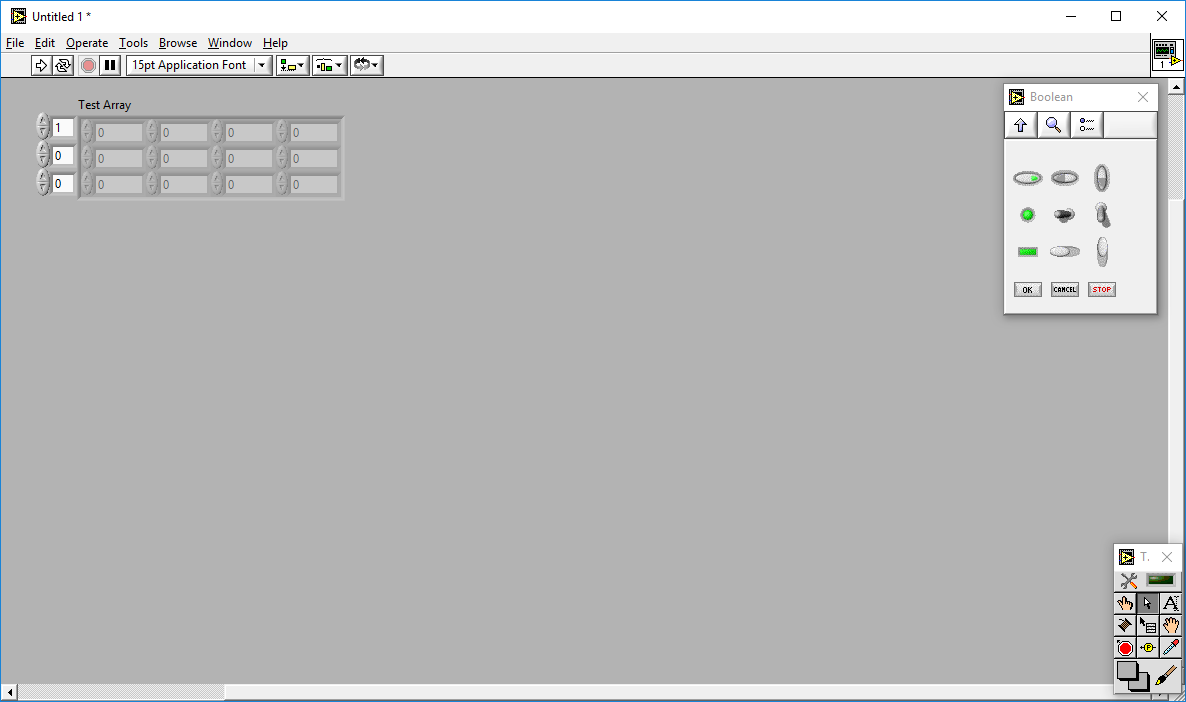


Рисунок 12 – Переход к первому листу трёхмерного массива

Для инициализации листа необходимо выполнить и его заполнение целочисленными значениями. С целью демонстрации того, что следующий лист является логическим продолжением предыдущего (как текст в книге) счёт элементов в нём продолжен (Рисунок 13).

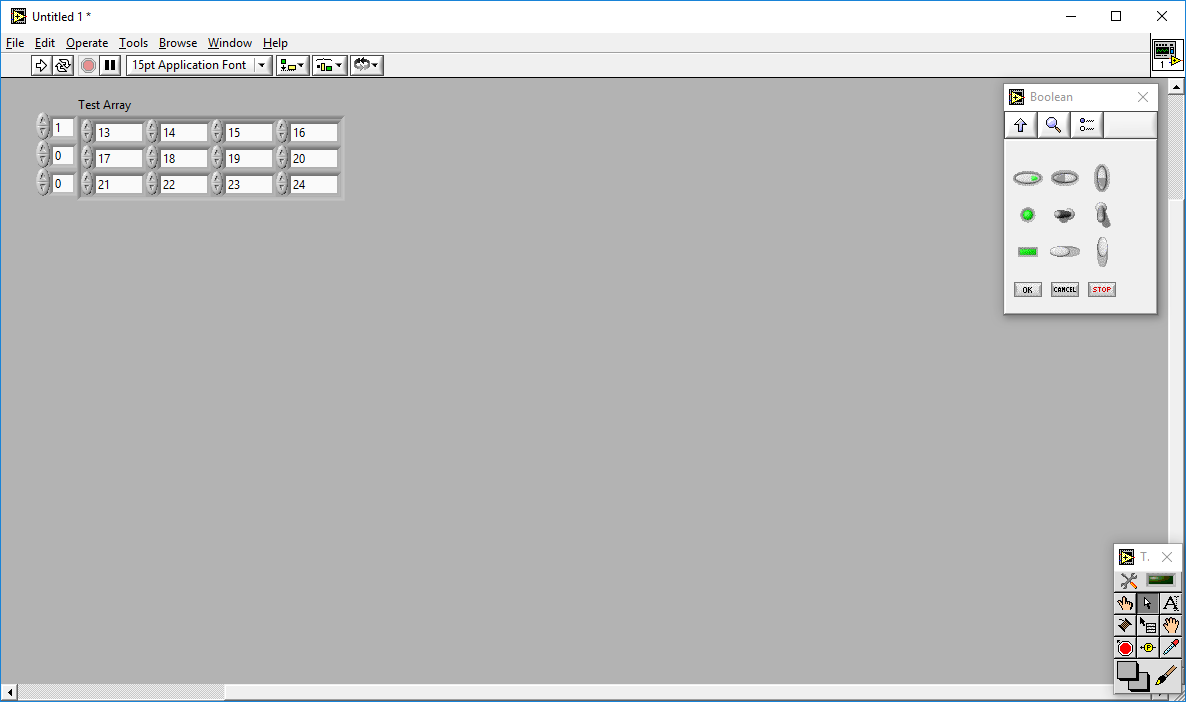


Рисунок 13 – Заполнение первого листа трёхмерного массива целочисленными элементами

Для последующего просмотра размерностей получившегося массива рядом с ним необходимо расположить ещё один массив, одномерный (Рисунок 14).

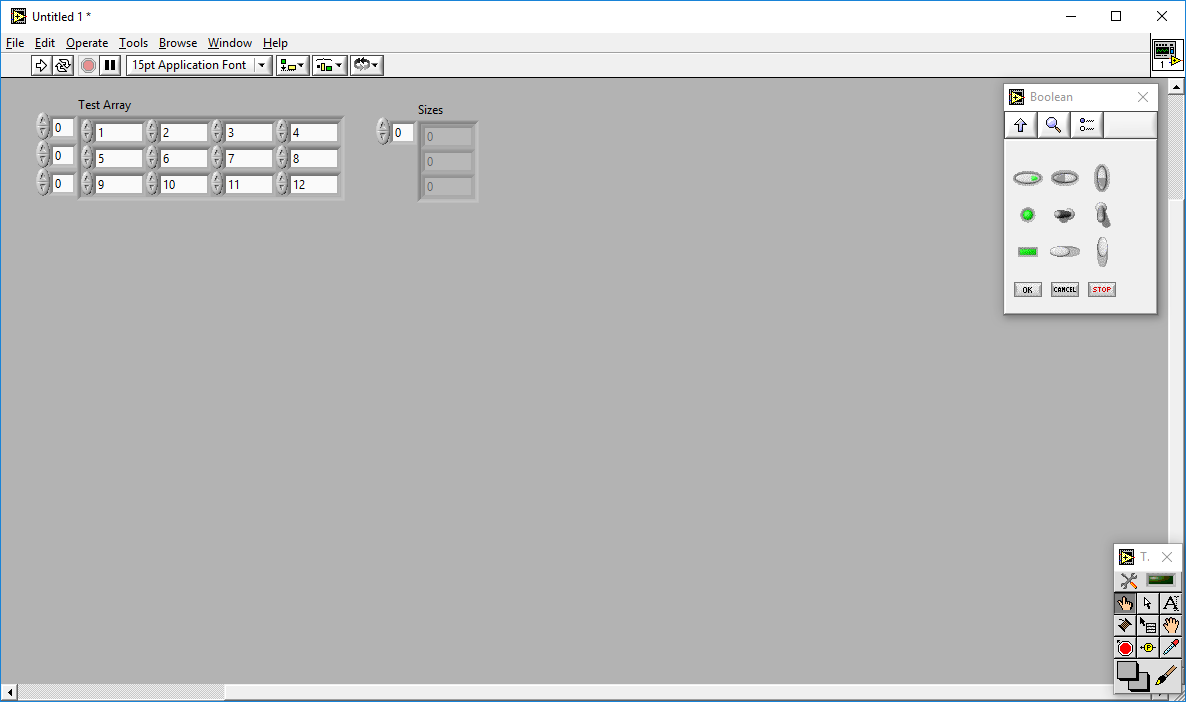


Рисунок 14 – Добавление на переднюю панель виртуального прибора одномерного массива для вывода значений размерностей трёхмерного массива

Далее на блок-диаграмме виртуального прибора необходимо связать оба этих массива через функцию (*Array Size*) получения размерностей (Рисунок 15).

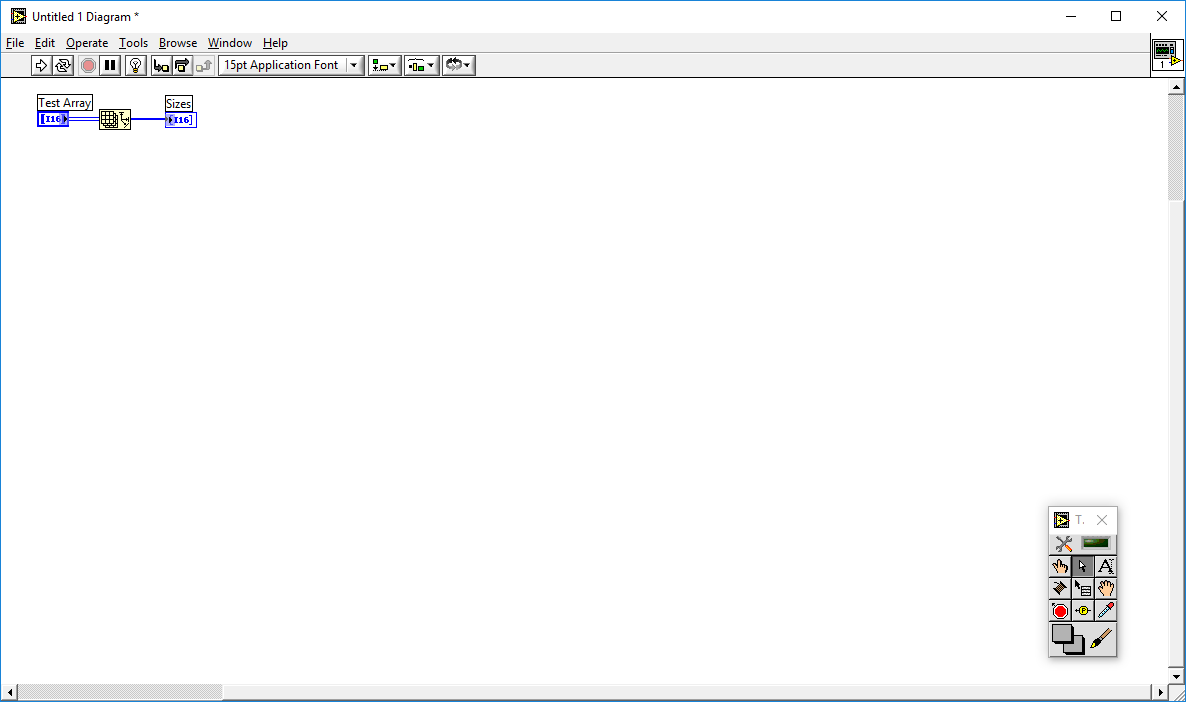


Рисунок 15 – Получение значений размерностей трёхмерного массива

После запуска получившейся программы на исполнение сверху вниз в одномерном массиве читается следующее: в исходном трёхмерном массиве 2 листа (2 страницы / 2 слоя), 3 строки и 4 столбца (Рисунок 16).

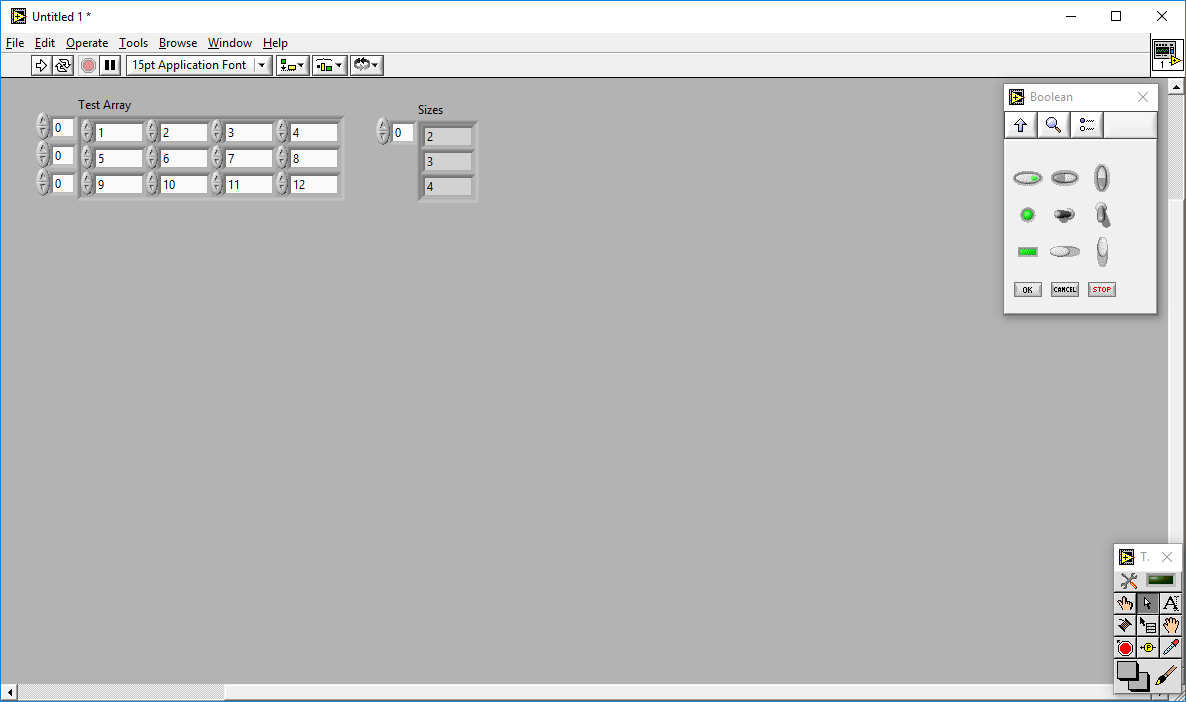


Рисунок 16 – Результат получения значений размерностей трёхмерного массива

Более углубленному пониманию и детальному усвоению навыков работы с функцией *Array Size* способствует изучение примера, связанного с выделением подматрицы из матрицы. К нему переходить рано без рассмотрения другой полезной функции для работы с массивами – *Index Array* (получение значения элемента массива по его индексу). Для получения значения [*i*, *j*]-го элемента матрицы по известным его индексам создан следующий виртуальный прибор (Рисунок 17, 18).

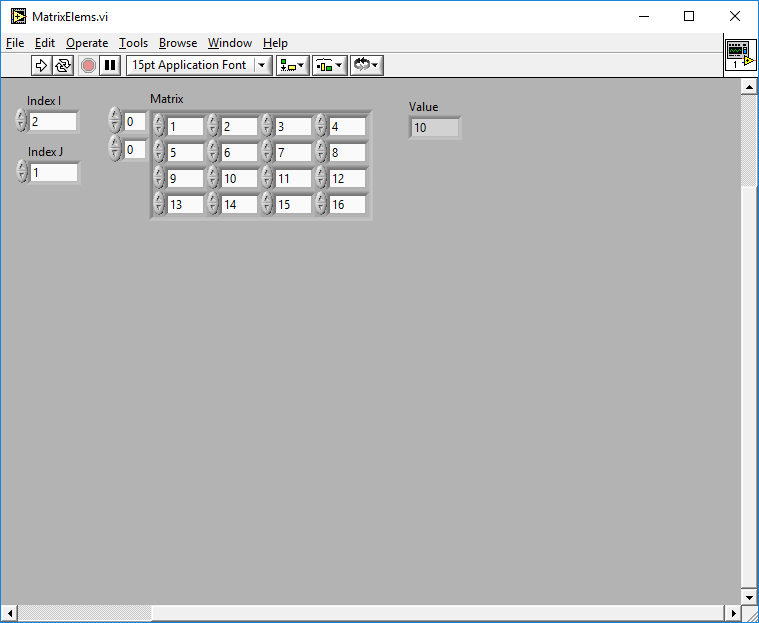


Рисунок 17 – Передняя панель ВП, выдающего в качестве ответа значения, хранящегося в двумерном массиве по указанному адресу (*i*, *j*)

Традиционно, *i* соответствует строке, *j* – столбцу. Нумерация начинается с нуля.

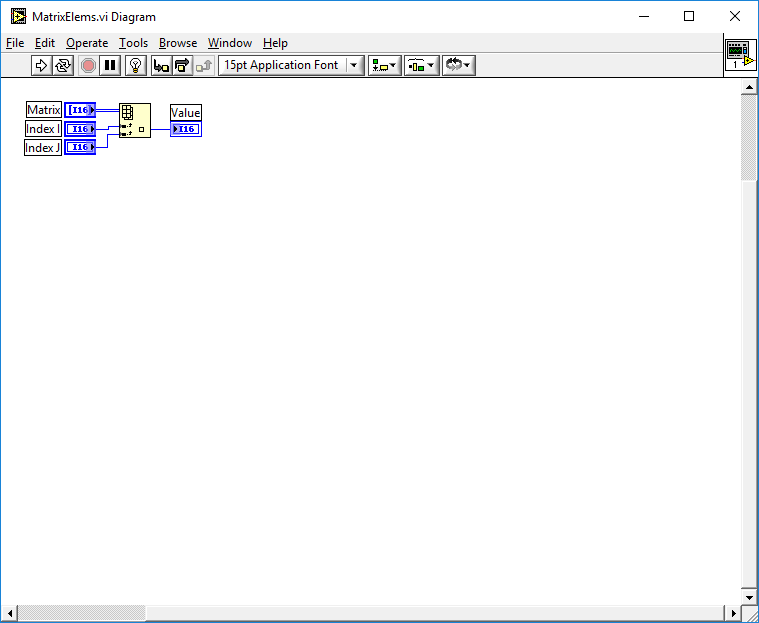


Рисунок 18 – Блок-диаграмма ВП, выдающего в качестве ответа значения, хранящегося в двумерном массиве по указанному адресу (*i*, *j*)

Как было отмечено ранее, пример задействует функцию получения значения элемента массива по указанному индексу (указанным индексам). Расположение функции для получения определённого индекса из массива в пакете прикладных программ *National Instruments LabView* представлено на Рисунке 19.

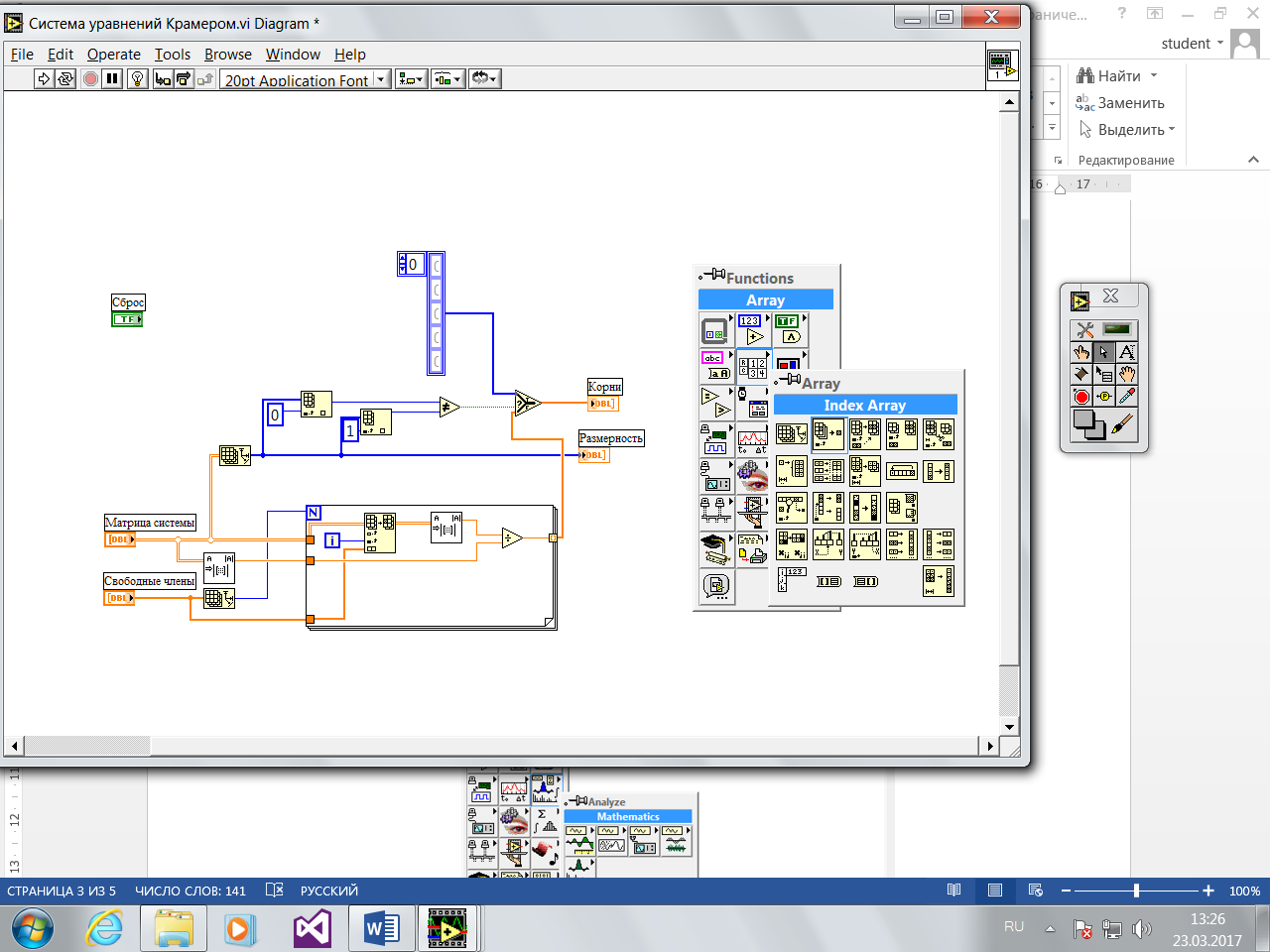


Рисунок 19 – Функция получения значения элемента массива по указанному индексу (указанным индексам)

Вот теперь информации достаточно для рассмотрения примера выделения подматрицы из исходной матрицы (*Array Subset*). Это важная операция, например, для поиска минора рассматриваемого элемента матрицы (Рисунок 20).

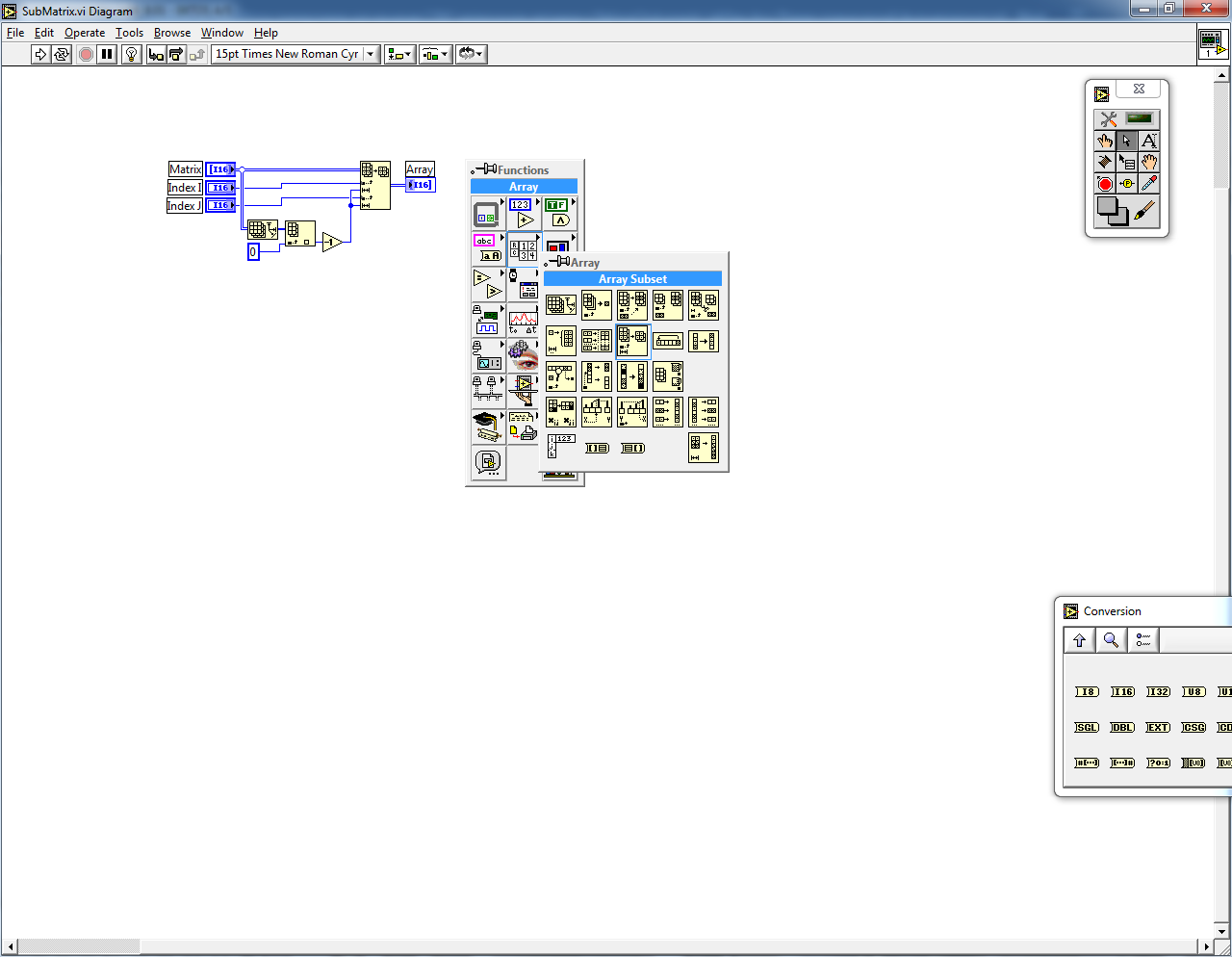


Рисунок 20 – Функция получения подматрицы из исходной матрицы

На Рисунке 21 представлен результат работы программы выделения подматрицы из исходной матрицы. Отсчёт по каждой из размерностей начинается с нуля, хотя в общем случае начало отсчёта динамически задаётся интерфейсными элементами управления «*Index I*» и «*Index J*». Длина по каждой из размерностей задаётся на единицу меньшей, чем размерность по строкам исходной матрицы. Последнее статично и не универсально, однако достаточно для демонстрации работы функции выделения подматрицы из исходной матрицы.

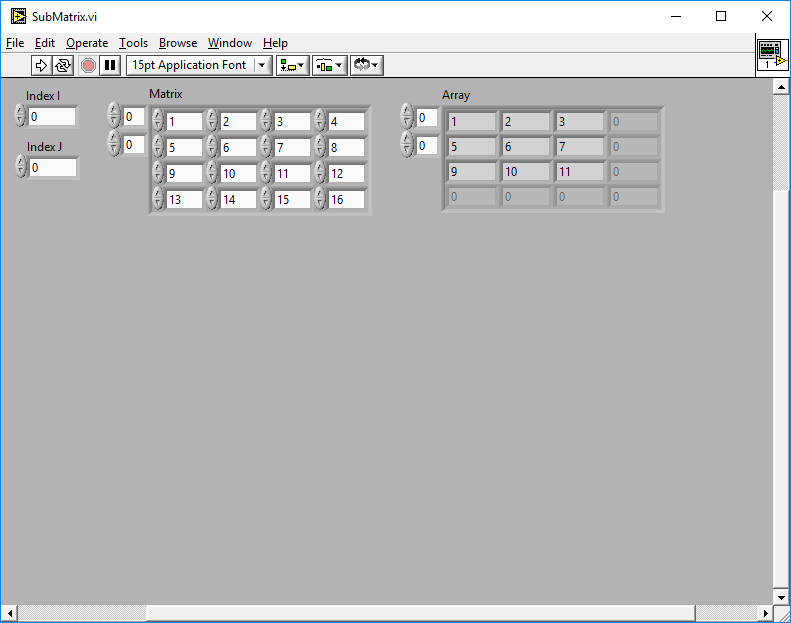


Рисунок 21 – Результат понижения размерности матрицы через выделение подматрицы

Далее на Рисунке 22 показана блок-диаграмма разработанного ВП.

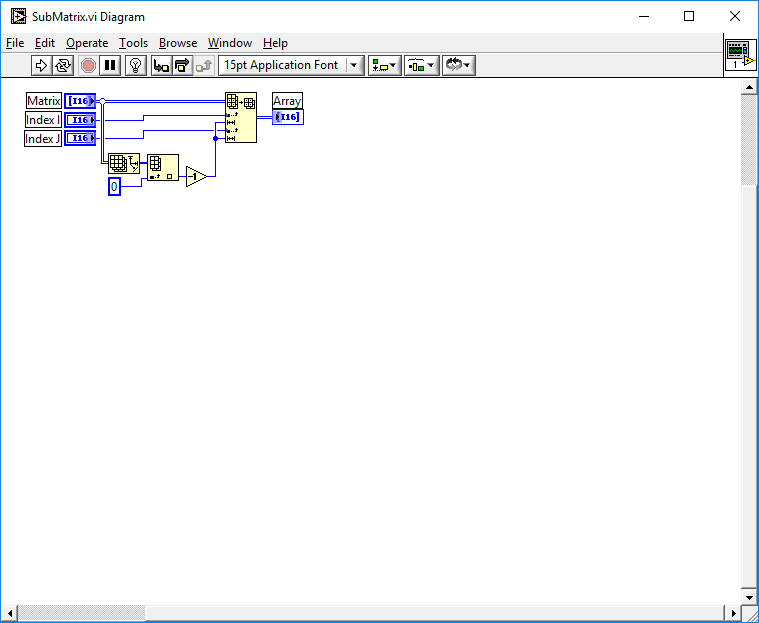


Рисунок 22 – Блок-диаграмма виртуального прибора для выделения подматрицы из исходной матрицы

В том случае, если выделение подматрицы не позволяет быстро и эффективно обособить минор указанного элемента, может быть использован иной и довольно быстрый способ исключения строки и столбца, однако он требует применения двух операций исключения строки/столбца (*Delete From Array*). На Рисунке 23 показано расположение соответствующей функции.

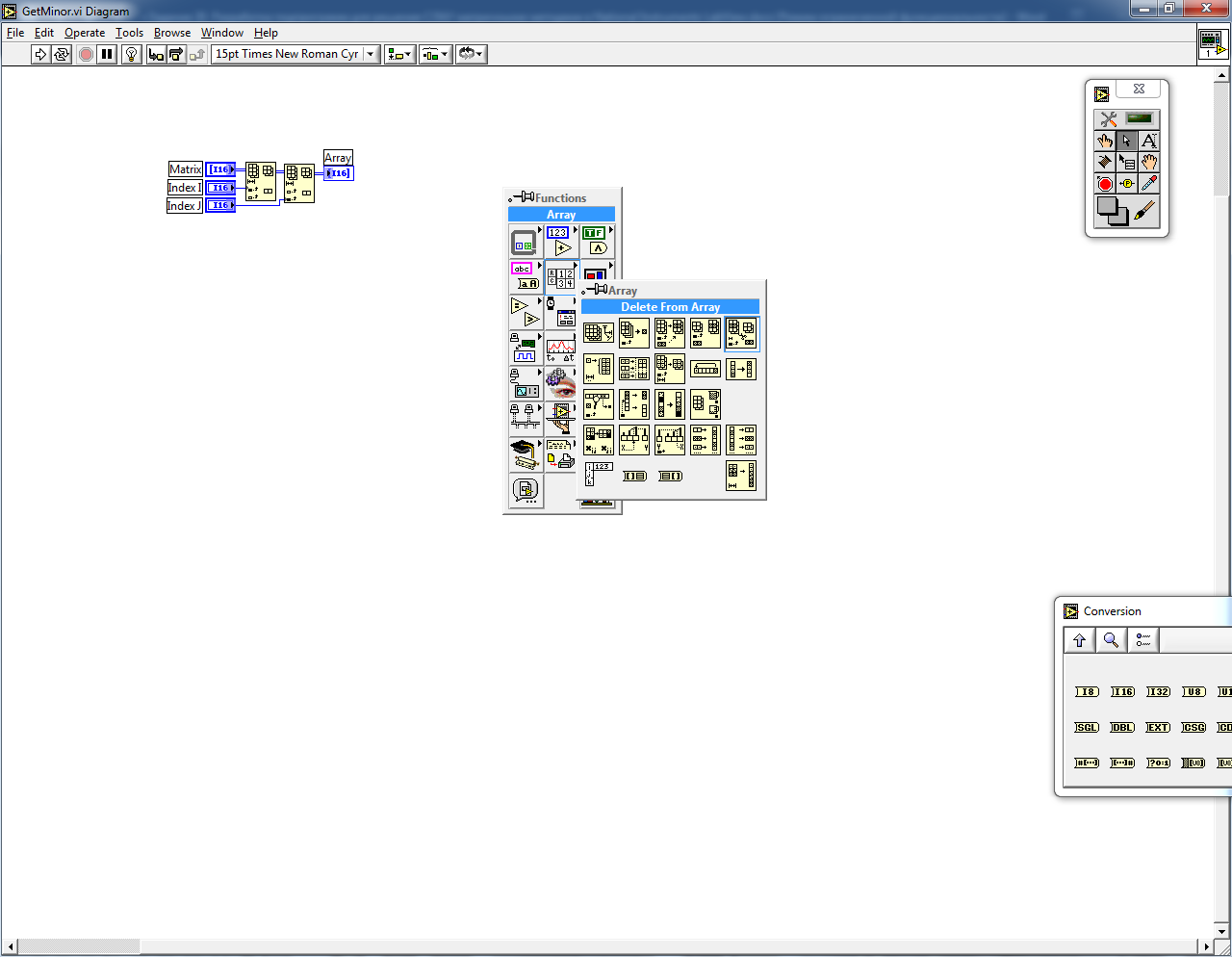


Рисунок 23 – Функция удаления строки/столбца из массива

Ещё раз следует напомнить, что начало отсчёта в *National Instruments LabView* ведётся с нуля. При учёте этого факта далее на Рисунке 24 показана работа виртуального прибора для определения минора указанного элемента матрицы. Минор элемента – это матрица на единицу меньшей размерности, откуда исключены строка и столбец, в которых находится рассматриваемый элемент.

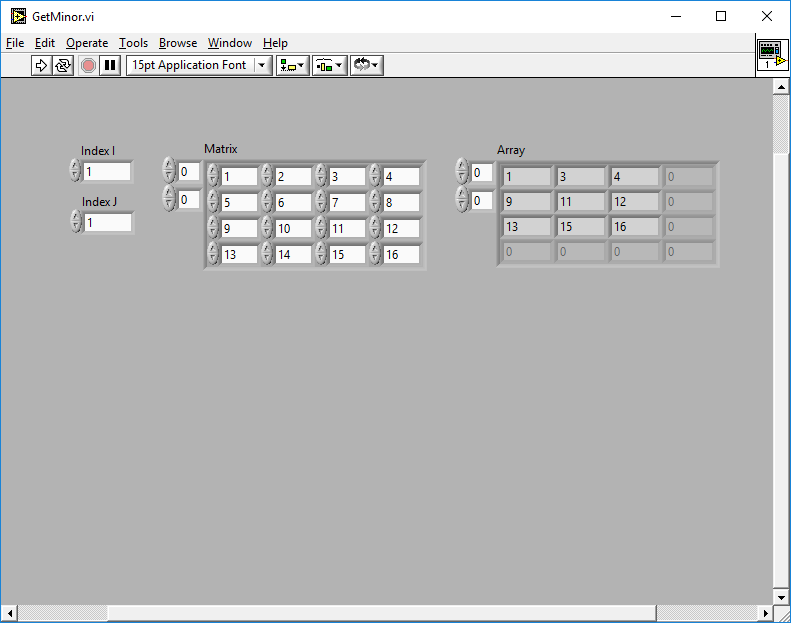


Рисунок 24 – Результат взятия минора для элемента [1,1], равного «6»

Блок-диаграмма на Рисунке 25 иллюстрирует последовательное исключение *i*-й строки и *j*-го столбца из исходной матрицы.

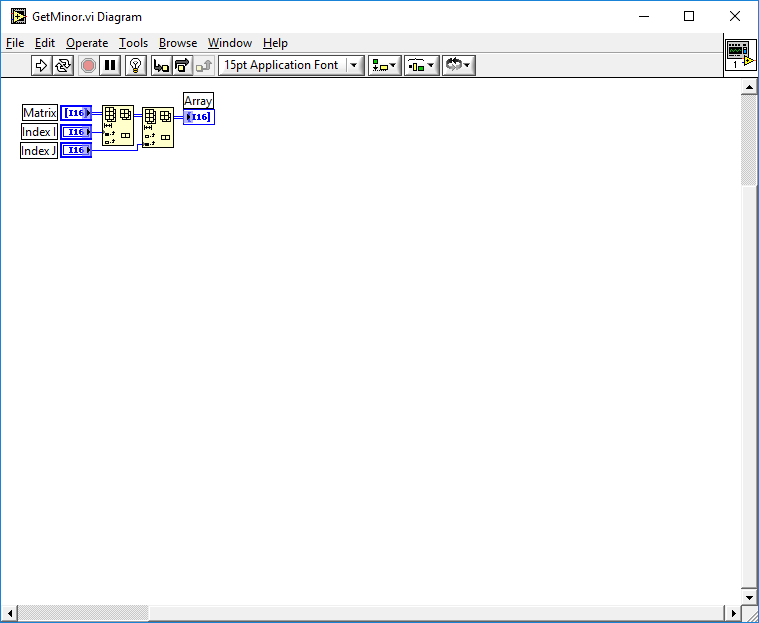


Рисунок 25 – Блок-диаграмма ВП, созданного для определения минора указанного элемента матрицы

Также для решения поставленной задачи методом Крамера потребуется использование функции подмены массива внутри матрицы. Расположение функции для подмены подматриц в исходных матрицах в пакете прикладных программ *National Instruments LabView* изображено на Рисунке 26.

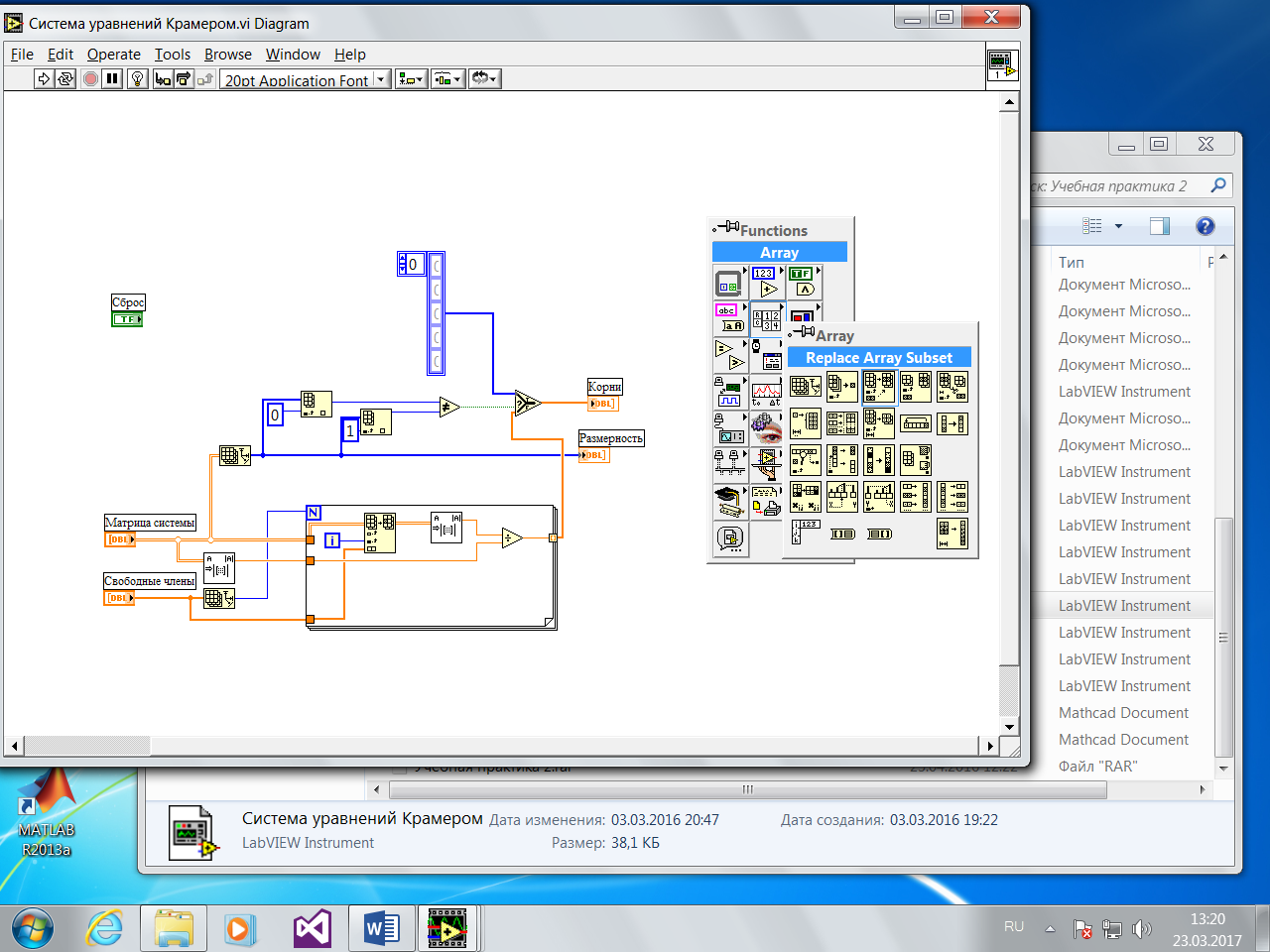


Рисунок 26 – Функция подмены подматрицы в исходной матрице

Другая часть задачи связана с использованием стандартных компонентов для поиска определителя матрицы, взятия обратной матрицы, а также для решения СЛАУ матричным методом. Расположение стандартной функции для решения СЛАУ матричным методом в пакете прикладных программ *National Instruments LabView* показано на Рисунке 27:

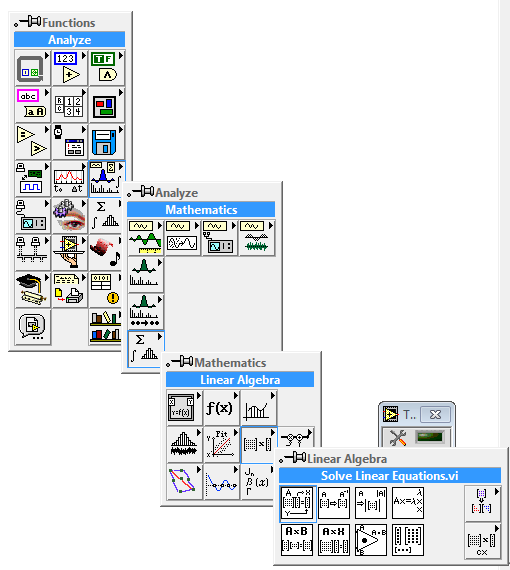


Рисунок 27 – Место расположения стандартного компонента для решения СЛАУ (*Solve Linear Equations*)

Расположение стандартного элемента для нахождения обратной матрицы указано на Рисунке 28:

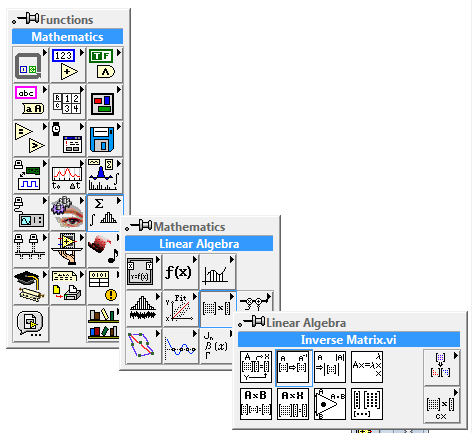


Рисунок 28 – Расположение стандартной функции для поиска обратной матрицы

**Для случаев, когда не удаётся реализовать универсальный алгоритм решения поставленной задачи:**

1. Ввести допущение, что программа не является универсальной и адаптирована под расчёт СЛАУ строго указанной на графическом пользовательском интерфейсе размерности.

2. Прибегнуть к разложению матрицы в частную формулу для поиска её определителя.

**Пример:**

Пусть дана матрица М:



Определитель матрицы М равен:



Для удобства вывода формулы откажемся на время от индексации, а поставим каждому элементу матрицы в соответствие букву латинского алфавита:



Далее для вывода формулы воспользуемся схемой поиска определителя матрицы по минорам:



Определитель рассчитывается строго при фиксированном *j*, то есть для одной любой строки матрицы. Поиск определителя минора для рассматриваемого элемента является рекурсивным обращением к указанной выше схеме. Выходом из рекурсии является достижение размерности 2х2, для которой известно расчётное соотношение вида: «Разность произведений элементов, стоящих на главной диагонали и элементов, стоящих на обратной диагонали».

Тогда для первой строки исходной матрицы рассмотрим её миноры:

1.



Результат был получен следующим образом:



Выполним подстановку переменных:



Упростим:



2.



Результат был получен следующим образом:



Выполним подстановку переменных:



Упростим:



3.



Результат был получен следующим образом:



Выполним подстановку переменных:



Упростим:



4.



Результат был получен следующим образом:



Выполним подстановку переменных:



Упростим:



Вернёмся к исходной матрице М и запишем:



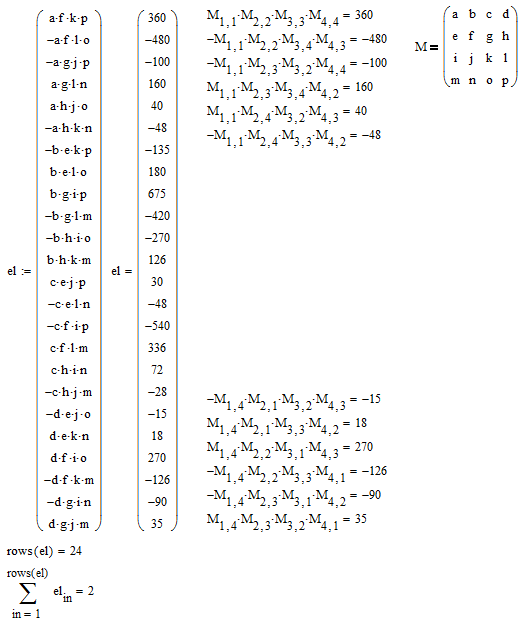
Иначе:



После подстановки выражений и упрощения получим:



Составим таблицу соответствия:



**Варианты индивидуального задания:**

Таблица 1 – Варианты индивидуального задания

|  |  |
| --- | --- |
| 1. | 2. |
| 3. | 4. |
| 5. | 6. |
| 7. | 8. |
| 9. | 10. |
| 11. | 12. |
| 13. | 14. |
| 15. | 16. |
| 17. | 18. |
| 19. | 20. |
| 21. | 22. |
| 23. | 24. |
| 25. | 26. |