



МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение
высшего образования

«МИРЭА – Российский технологический университет»
РТУ МИРЭА

Институт информационных технологий
Кафедра Вычислительной техники

Отчет по практической работе №4

по дисциплине
«Теория автоматов»

Тема практической работы:
«Делитель 4-разрядных чисел в дополнительном коде»

Выполнил студент группы ИВБО-02-19

К. Ю. Денисов

Проверил ассистент

А. С. Боронников

Москва 2021

Содержание

1	Общее строение автомата	3
2	Индивидуальное задание	4
3	Алгоритм работы автомата	5
4	Реализация Операционного автомата	5
5	Реализация управляющего автомата	7
6	Тестирование работы автомата	8
7	Вывод	10
	ПРИЛОЖЕНИЕ А	11

1 Общее строение автомата

В любом устройстве обработки цифровой информации можно выделить два основных блока – операционный автомат (ОА) и управляющий автомат (УА). Операционный автомат (ОА) служит для хранения слов информации, выполнения набора микроопераций и вычисления значений логических условий, т.е. операционный автомат является структурой, организованной для выполнения действий над информацией. Микрооперации, выполняемые ОА, задаются множеством *управляющих* сигналов $Y \{y_1, \dots, y_M\}$, с каждым из которых отождествляется определенная микрооперация.

Значения логических условий, вычисляемые в операционном автомате, отображаются множеством *осведомительных* сигналов $X = \{x_1, \dots, x_L\}$, каждый из которых отождествляется с определенным логическим условием.

Управляющий автомат (УА) генерирует последовательность управляющих сигналов, предписанную микропрограммой и соответствующую значениям логическим условий. Управляющий автомат задает порядок выполнения действий в ОА, вытекающий из алгоритма выполнения операций. Наименование операции, которую необходимо выполнить в устройстве, определяется кодом g операции, поступающим в УА извне.

В отличие от УА с жесткой логикой, закон функционирования которого обеспечивается определенным образом соединенными логическими элементами, в автоматах, построенных на основе ПЗУ, заданная микропрограмма реализуется в явной форме и хранится в памяти в виде последовательности управляющих слов. Управляющее слово определяет порядок работы устройства в течение одного такта и на-

зывается микрокомандой (МК). Она содержит информацию о микрооперациях, которые должны выполняться в данном такте, и (или) об адресе следующей микрокоманды.

2 Индивидуальное задание

В ходе данной практической работы был реализован автомат, выполняющий деление 4-разрядных чисел в дополнительном коде (алгоритм без восстановления остатка). Управляющий автомат был построен по схеме с сокращенным тактом. Рассмотрим строение управляющего автомата. См рисунок 1.

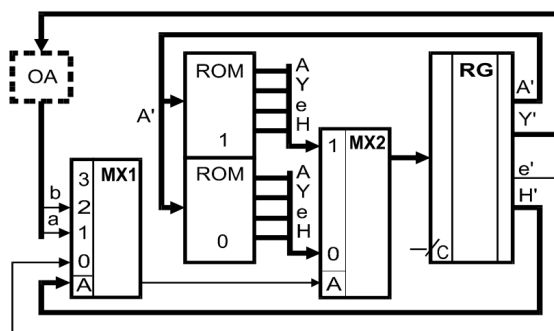


Рис. 1: УА с сокращенным тактом

В конкретной реализации на информационные входы мультиплексора подаются сигналы e – сигнал с ПЗУ, B_IS_NULL – признак нулевого делителя, CT_IS_NULL – признак окончания счета, b_high – значение старшего бита частичного остатка на текущей итерации, а на адресный вход подается двухбитовый сигнал H .

3 Алгоритм работы автомата

Опишем алгоритм работы автомата с помощью блок схемы. Используем сумматор для нахождения текущего значения частичного остатка (ЧО), счетчик для подсчета обработанных разрядов и регистры для хранения и использования разрядов делителя и делимого. Обозначим микрокоманды от m_0 до m_4 . См. рисунок 4 в Приложении А.

После построения алгоритма работы автомата следует перейти к реализации операционной части.

4 Реализация Операционного автомата

Построим операционный автомат, выполняющий деление двух 4-разрядных чисел в дополнительном коде посредством использования четырех регистров, в том числе трех сдвиговых. Приведем названия и назначения каждого из регистров. См. таблицу 1.

Идентификатор	Назначение
RG_A	Сдвиговый регистр. Хранит разряды делимого
RG_B	Хранит разряды делителя
RG_REM	Сдвиговый регистр. Хранит разряды частичного остатка
RG_RES	Сдвиговый регистр. Хранит разряды результата

Таблица 1: Регистры операционного автомата

Укажем необходимые признаки, которые впоследствии будут вырабатываться управляющим автоматом. См. таблицу 2. С целью ре-

Признак	Назначение
S	Хранит адрес следующей операции
H	Адресный вход мультиплексора
$R0$	Сигнализирует об окончании операции деления
$ERROR$	Сигнализирует об ошибке ввода – делитель равен нулю
L_RG_A	Загрузка в регистр RG_A
L_RG_B	Загрузка в регистр RG_B
L_RG_{REM}	Загрузка в регистр RG_REM
CLR	Асинхронный сброс всех элементов
$COUNT_CT$	Счет. Декремент счетчика, если $L_CT == 1$
L_CT	Загрузка счетчика
$SHIFT$	Левый сдвиг в регистрах RG_A и RG_RES

Таблица 2: Осведомительные сигналы (признаки)

ализации левого сдвига в сдвиговых регистрах разряды делимого и текущего значения частичного остатка загружаются в регистры в обратном порядке. Это позволяет отказаться от универсального сдвигового регистра, так как реализации данного алгоритма необходим только левый сдвиг, и как следствие, упрощает схему.

Для исключения возникновения ошибки при левом сдвиге разрядов частичного остатка в случае, когда два его старших разряда различны, использованы 6-разрядные регистры. См. сноску 1.

$$\begin{aligned}
 -|B| &\neq 10X...X \\
 -|B| &= 110X...X
 \end{aligned}
 \tag{1}$$

Стоит отметить, что для формирования правильного выходного

Комбинация	Коррекция
$A \geq 0, B > 0$	Коррекция не требуется
$A \geq 0, B < 0$	Перевести частное в доп. код
$A \leq 0, B > 0$	Результат верен, если остаток = 0. Иначе прибавить к отрицательному частному единицу, перевести остаток в доп. код
$A \leq 0, B \leq 0$	Изменить знак делимого, перевести остаток в доп. код

Таблица 3: Коррекция результата

результата необходимо выполнить коррекцию значений частного и остатка в зависимости от знаков операндов. Для каждой комбинации знаков делимого и делителя реализована отдельная операция коррекции. См таблицу 3.

Соединим все элементы в соответствии с алгоритмом задачи. См. рисунки 5 и 6 в приложении **Приложении А**.

5 Реализация управляющего автомата

Приступим к построению управляющего автомата, определяющего последовательность выполнения микрокоманд для деления двух 4-разрядных чисел в дополнительном коде.

Определим разрядность двух ПЗУ, участвующих в построении УА по схеме с сокращенным тактом. Адрес должен иметь 3 разряда — текущее значение параметра S . Микрокоманда представлена в виде 15 бит — 12 признаков, расположенных в следующем порядке: $S, H, R0, ERROR, CLR, L_CT, COUNT_CT, SHIFT, L_RG_REM, L_RG_B, L_RG_A, e$. Текущая команда хранится в 15 ти разрядом регистре. В первом ПЗУ, имеющем метку «0» будут храниться микро-

манды, вторая часть адреса которых содержит нулевой бит. Автомат переходит в эти состояния, когда условия, заключенные в операторе выбора, отображенном на диаграмме, не выполняются. Второй ПЗУ с меткой «1» хранит микрокоманды, в которые автомат переходит при выполнении условий, заключенных в операторе выбора. Данные микрокоманды имеют единичный бит во второй части адреса.

Микрокоманды, расположенные в ПЗУ по текущему адресу попадают на вход мультиплексора, адресным входом которого является значение признака, рассматриваемого в текущем состоянии.

Заполним память в соответствии с алгоритмом, подключим ПЗУ, мультиплексоры и регистр последовательным способом. См рисунок 7 в Приложении А.

6 Тестирование работы автомата

После реализации операционного и управляющего автомата следует приступить к объединению данных устройств и тестированию их совместной работы. Подключим признаки к входам соответствующих логических элементов и цифровых устройств с помощью туннелей. Добавим блок ввода исходных данных, используя контакты, блок вывода — регистр результата операции деления и регистр остатка, индикатор завершения операции деления.

Проведем проверку корректности выходных результатов построенного цифрового устройства. Разделим два наибольших 4-разрядных двоичных числа $1111_2 \div 1111_2$ ожидая получить частное 1_2 и остаток 0_2 . Укажем входные данные, будем подавать тактовые сигналы до тех пор, пока индикатор не сообщит нам о завершении операции, срав-

ним практические результаты с ожидаемыми. См рисунок 2. Умножение выполнено корректно. Ожидаемые и полученные результаты совпадают.

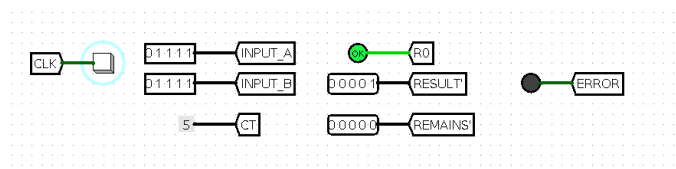


Рис. 2: Проверка работы автомата

Протестируем работу автомата на входных данных, которые гипотетически могут привести к ошибке — поделим число 101_2 на число 1111_2 . Ошибка состоит в том, что в случае использования регистров недостаточной разрядности, левый сдвиг при вычислении очередного остатка произойдет некорректно, что приведет к неверному результату. Введем данные, наблюдаем за работой автомата. См рисунок 3. Операция выполнена верно. Использование регистров большей разрядности исключило возможность ошибки.

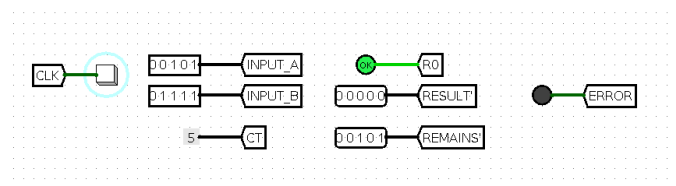


Рис. 3: Вторая проверка работы автомата

Проведем тестирование работы автомата на отрицательных числах. Рассмотрим три варианта, при которых:

1. Делимое неотрицательно, делитель отрицателен;
2. Делимое отрицательно, делитель положителен;

3. Делимое отрицательно, делитель отрицателен.

Рассмотрим конкретные примеры.

№	Комбинация		Ожидаемый	Полученный
1	$8 \div -4$	$01000_2 \div 11100_2$	$11100_2, 00000_2$	$11100_2, 00000_2$
2	$-8 \div 3$	$11000_2 \div 00011_2$	$11110_2, 11110_2$	$11110_2, 11110_2$
3	$-13 \div -3$	$10011_2 \div 11101_2$	$00100_2, 11111_2$	$00100_2, 11111_2$

7 Вывод

В ходе данной практической работы было рассмотрено строение и работа управляющего автомата, построенного по схеме с укороченным тактом. Используя полученные знания на практике, на основе данного управляющего автомата построено вычислительное устройство (операционный и управляющий автомат), реализующее операцию деления двух 4-разрядных чисел в дополнительном коде.

Работа данного устройства испытана, проверена корректность полученных результатов при работе со всеми категориями входных данных.

ПРИЛОЖЕНИЕ А

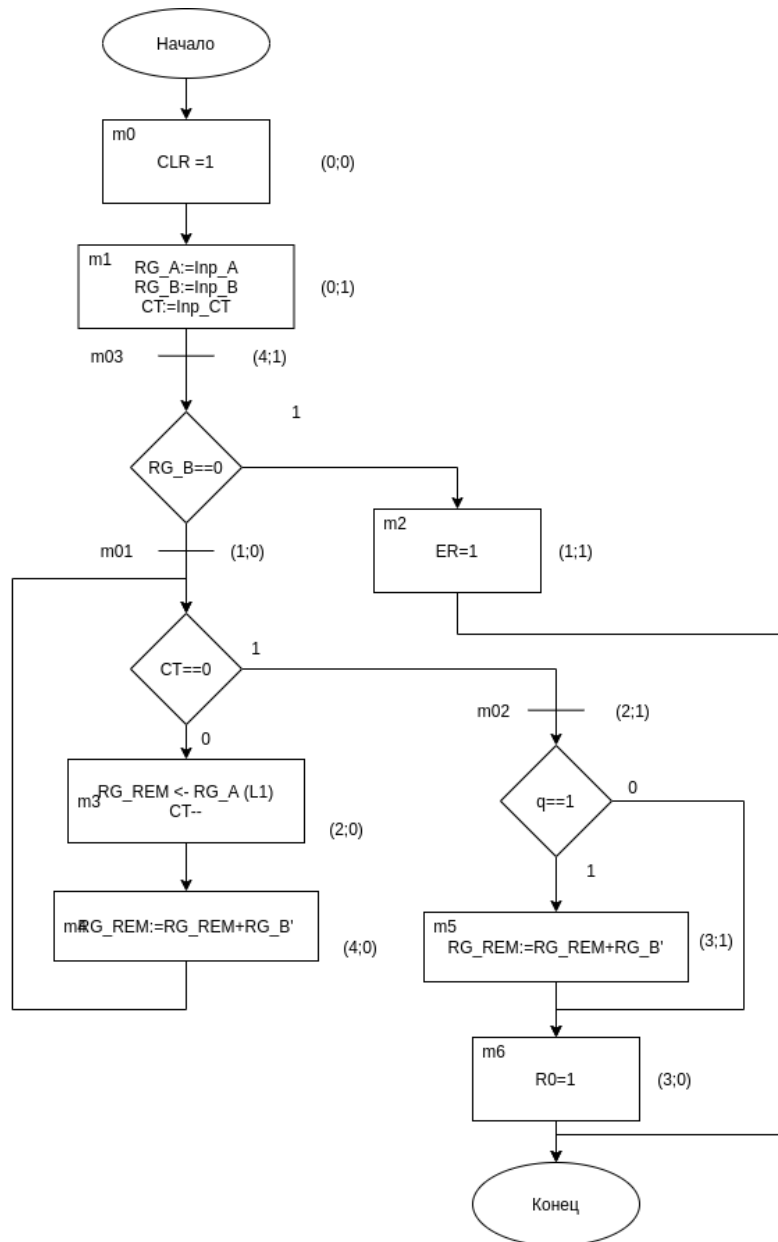


Рис. 4: Алгоритм деления двух 4-разрядных чисел

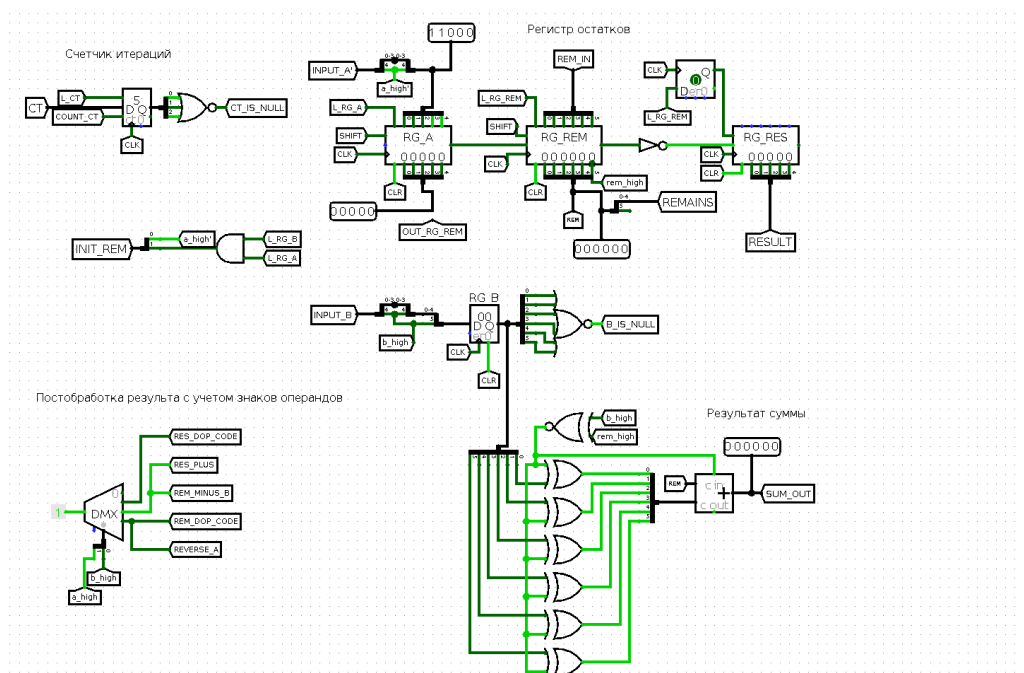


Рис. 5: Схема операционного автомата, часть 1

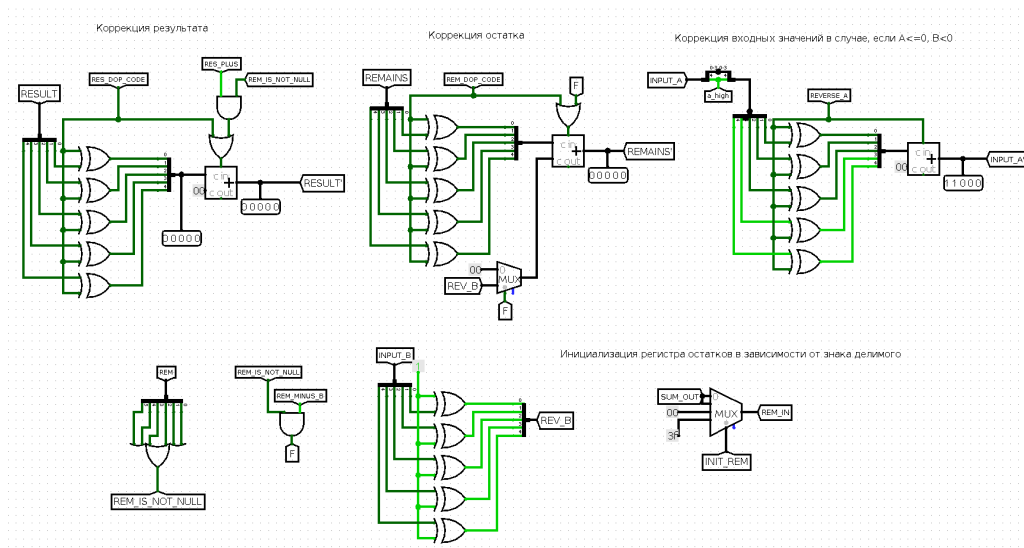


Рис. 6: Схема операционного автомата, часть 2

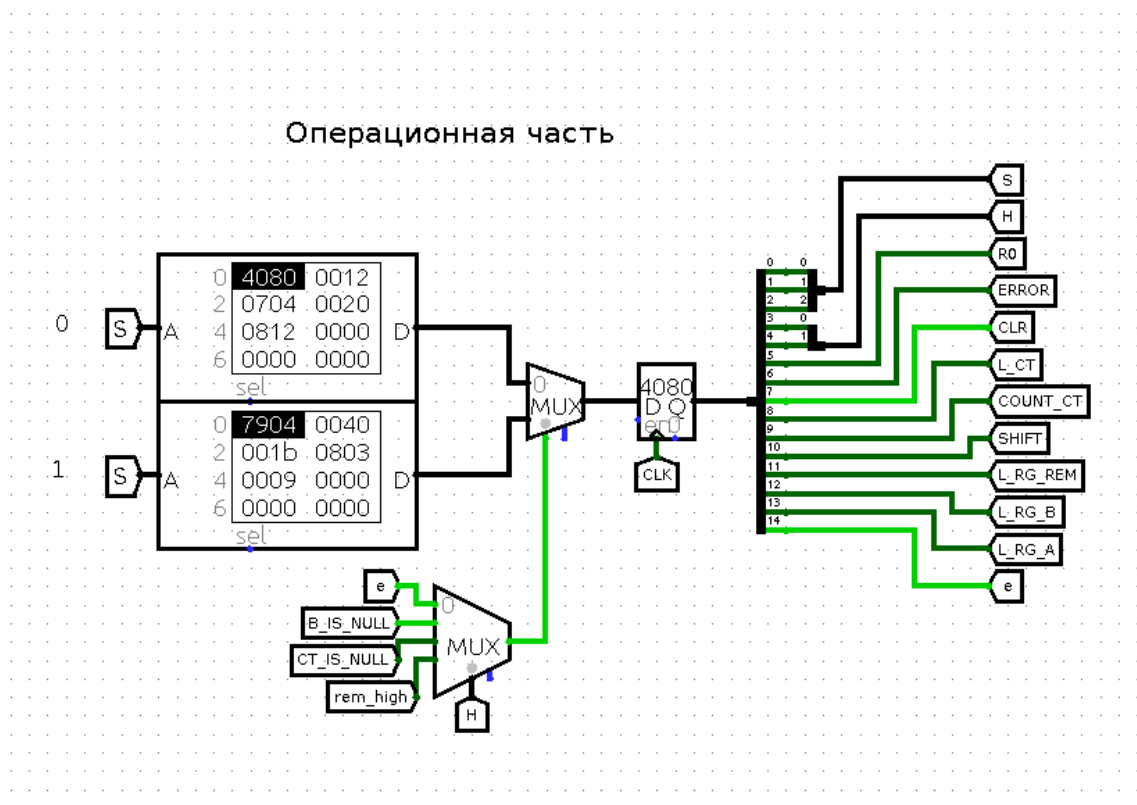


Рис. 7: Схема операционного автомата