

# РЕФЕРАТ

Отчет 38 с., 1 кн., 15 рис., 3 табл., 2 источн., 2 прил.

АРИФМЕТИКО-ЛОГИЧЕСКОЕ УСТРОЙСТВО, УПРАВЛЯЮЩЕЕ УСТРОЙСТВО, СЛОЖЕНИЕ, ВЫЧИТАНИЕ, ЧЁТНОСТЬ.

Цель работы – разработать АЛУ для операций сложения/вычитания и конъюнкция. Операндами являются 16-разрядные двоичные числа.

При разработке АЛУ использовались концепции «черного ящика», т.е. первоначальное определение общих функций устройства и системы входных и выходных сигналов. В основе дальнейшей работы с «черным ящиком» использовался принцип декомпозиции, т.е. последовательное разложение функций на подфункции до получения описания функций на элементарном уровне.

В ходе выполнения курсовой работы получены следующие результаты: разработано АЛУ на основе учебной модели ЭВМ, приведен контрольный пример в числовой форме.

**СОДЕРЖАНИЕ**

[ПЕРЕЧЕНЬ СОКРАЩЕНИЙ И ОБОЗНАЧЕНИЙ 6](#_Toc182370031)

[ВВЕДЕНИЕ 7](#_Toc182370032)

[1 Исследование предметной области 8](#_Toc182370033)

[1.1 Дополнительный код числа 8](#_Toc182370034)

[1.2 Алгоритм сложения/вычитания двоичных чисел 8](#_Toc182370035)

[1.3 Алгоритм конъюнкции 10](#_Toc182370036)

[2 Реализация устройства 12](#_Toc182370038)

[2.1 Анализ исходных данных задания на курсовую работу 12](#_Toc182370040)

[2.2 Спецификация устройства на уровне «черного ящика» 12](#_Toc182370041)

[2.3 Представление «черного ящика» в виде операционной и управляющей частей 13](#_Toc182370042)

[2.4 Разработка структуры операционной части устройства 13](#_Toc182370043)

[2.5 Разработка схемы алгоритма работы 14](#_Toc182370044)

[2.6 Составление полной спецификации устройства 16](#_Toc182370045)

[2.7 Временная диаграмма работы УУ 18](#_Toc182370047)

[2.8 Контрольный пример 20](#_Toc182370048)

[ЗАКЛЮЧЕНИЕ 22](#_Toc182370049)

[СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ 23](#_Toc182370050)

[ПРИЛОЖЕНИЕ А Листинг программы 24](#_Toc182370051)

[ПРИЛОЖЕНИЕ Б Контрольные примеры 37](#_Toc182370052)

# ПЕРЕЧЕНЬ СОКРАЩЕНИЙ И ОБОЗНАЧЕНИЙ

В настоящей курсовой работе применяют следующие обозначения и сокращения:

АЛУ – Арифметико-логическое устройство

ГСА – граф-схема алгоритма

ДК – Дополнительный код числа

МО – Микрооперация

МПР – Микропрограмма

ОЧ – Операционная часть устройства

УЧ – Управляющая часть устройства

ЧЯ – Черный ящик

# ВВЕДЕНИЕ

АЛУ является одним из ключевых компонентов цифровых вычислительных систем, обеспечивающим выполнение базовых арифметических и логических операций. От эффективности работы АЛУ зависит общая производительность и функциональность всей вычислительной системы, будь то простейший микроконтроллер или высокопроизводительный процессор. В современных процессорах и микроконтроллерах АЛУ применяется для выполнения множества операций с данными.

Цель работы – разработать АЛУ для операций сложения/вычитания и чётность чисел единиц с использованием автомата Мили и отслеживанием флагов состояния: флаг переполнения (OV) и флаг четности количества единиц (P). Операндами являются 16-разрядные двоичные числа.

Объектом исследования является устройство управления. Предметом исследования – алгоритм работы и структура устройства.

Курсовая работа включает: исследование структуры АЛУ, анализ основных требований, создание алгоритма работы, разработку спецификации устройства на уровне «черного ящика», схемы алгоритма работы устройства и его микропрограммы, составление полной спецификации устройства, построение временной диаграммы работы устройства управления. Помимо этого, в работе приведен листинг разработанной программы для учебной модели ЭВМ и результаты расчета контрольного примера.

1. Исследование предметной области

# Дополнительный код числа

**Дополнительный код** — это способ представления целых чисел в двоичной системе, который используется для удобства выполнения арифметических операций над отрицательными числами. Система ДК помогает избежать сложностей, связанных с отдельным знаком числа, и позволяет использовать единый алгоритм для сложения и вычитания чисел независимо от их знака [1].

Связь между числом и его изображением в ДК определяется соотношениями в формуле (1).

(1)

Таким образом, и ДК положительного числа равен самому числу (как обратный и прямой). ДК отрицательного числа дополняет исходное число до основания системы счисления.

ДК отрицательного числа образуется в соответствии с формулой (2).

(2)

Таким образом, для преобразования отрицательного двоичного числа в ДК следует преобразовать его сначала в обратный код (установив знаковый разряд в 1 и проинвертировав все остальные разряды числа) и добавить единицу к младшему разряду обратного кода.

# Алгоритм сложения/вычитания двоичных чисел

Так как для арифметических операций используется ДК – алгоритмы сложения и вычитания будут практически идентичны. Вычитание будет отличаться лишь тем, что перед выполнением перевода операнд в ДК, необходимо инвертировать знаковый бит вычитаемого.

Алгоритм сложения/вычитания двоичных чисел можно описать следующим образом:

1. Перевод операндов в ДК:

- Если число положительное – его представление в ДК не меняется.

- Если число отрицательное – его ДК получается путем инвертирования всех битов и прибавления 1 к результату.

1. Если выбрана операция сложения – выполняется побитовое сложение операндов, если выбрана операция вычитания – у второй операнды знак изменяется на противоположный:

- Начать с младшего бита и двигаться к старшему.

- На каждом шаге складывать соответствующие биты и добавлять перенос с предыдущего шага, если он есть.

- Если результат сложения битов больше 1, то оставить в текущем разряде остаток от деления на 2, а перенос передать в следующий разряд.

1. Определение флага переполнения (OV):

- Переполнение возникает, если оба старших бита операндов равны, но старший бит результата отличается.

1. Определение флага чётности числа единиц (P):

- Если в результате операции число единиц чётное, установить флаг P.

5) Обратный перевод:

- Полученный результат перевести из дополнительного кода в прямой.

Таким образом, были выделены все необходимые шаги для сложения/вычитания двоичных чисел.

Пример:

Пусть даны два числа:

В результате суммирования дополнительных кодов получится:

Для того чтобы произвести вычитание – инвертируем знаковый бит операнды B.

Тогда в результате суммирования А и -В, получим:

При переполнении – старший разряд уходит и остается положительное значение С.

Таким образом полученный результат соответствует вычисленному математически.

# Алгоритм конъюнкции

Конъюнкция (логическое умножение, операция «И», AND) является бинарной операцией, возвращающей 1, только когда оба операнда равны 1. При работе с двоичными числами из нескольких разрядов, эта операция осуществляется поразрядно. Поразрядные операции как правило осуществляются с операндами равной длины. В случае разной длины операндов, тот, который имеет меньшую длину, дополняется нулями в старших степенях. Например, при необходимости осуществить поразрядное логическое умножение чисел c = 110010 и d = 1010, число d следует представить как d = 001010.

В записи операция логического умножения может обозначаться различными способами, как показано в формуле:

Рассмотрим операцию поразрядного логического умножения на примере двух двоичных чисел a и b:

Таким образом, a ^ b = 101101 ^ 100110 = 100100.

**2 Разработка устройства**

## **2.1 Анализ исходных данных задания на курсовую работу**

Необходимо разработать устройство для выполнения операций алгебраического сложения и вычитания, а также конъюнкции. Исходные данные поступают в формате 16-разрядных двоичных чисел. Исходя из этого очевидно, что входы (A, B) и выход (C) должны иметь 16 разрядов.

**2.2 Спецификация устройства на уровне «черного ящика»**

В общем случае устройство должно иметь два 16 разрядных входа для операндов A и B, один 16 разрядный выход C для результата, 2 одноразрядных разрядных выхода для флагов OV (признак арифметического переполнения) и P (признак чётности количества единиц), один 2 разрядный вход U для сигнала управления.



Рисунок 1 – Схема устройства на уровне «черного ящика»

## **2.3 Представление «черного ящика» в виде управляющего устройства и операционного устройства**

Упрощенное представление разрабатываемого устройства в виде управляемой части и операционной части представлено на рисунке 2.

Рисунок 2 – Разбиение схемы устройств на УЧ и ОЧ

## **2.4 Разработка структуры операционного устройства**

Операционная часть устройства имеет два регистра A и B для размещения в них операндов и один регистр C для сохранения результата. Упрощенная структура ОЧ устройства представлена на рисунке 3.



Рисунок 3 – Структурная схема операционного устройства

1. 1. Разработка схемы алгоритма работы

Схема алгоритма умножения на уровне микроопераций изображена на рисунке 4.

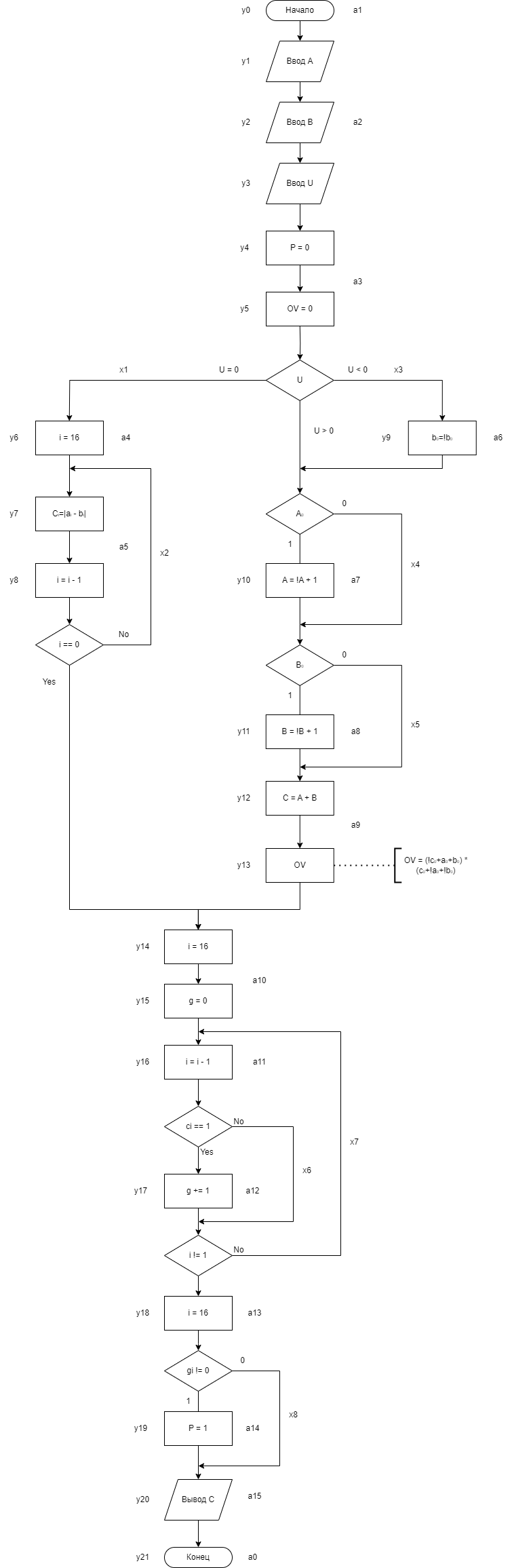


Рисунок 4 – Схема алгоритма на уровне операций

На основе графической схемы алгоритма умножения возможно построить граф переходов автомата Мили. Вершины графа соответствуют состояниям автомата, дуги – переходам из состояния am в состояние as. У вершин графа записываются микрокоманды, соответствующие состояниям, в начале дуги – логические условия, определяющие переход из состояния am в состояние as. Граф переходов представлен на рисунке 5.

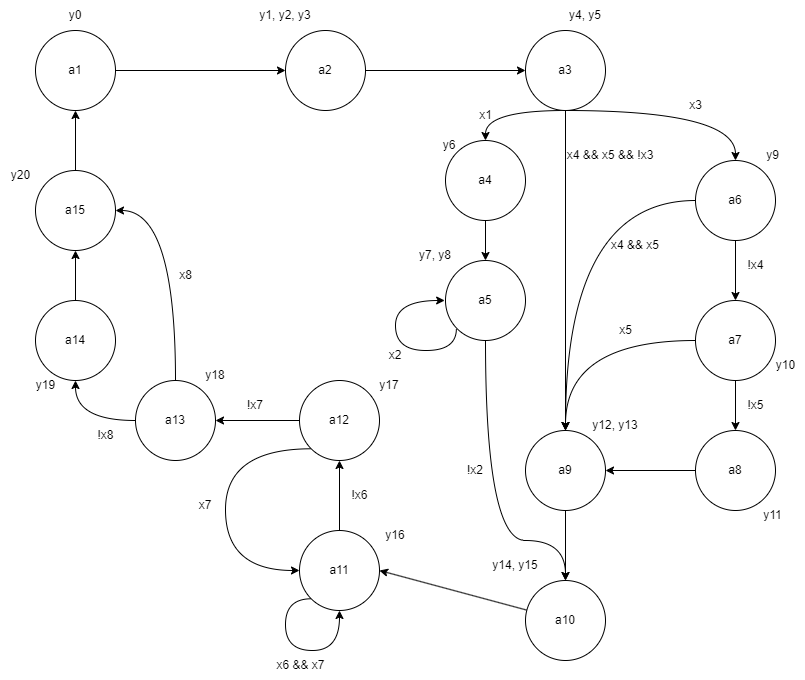


Рисунок 5 – Граф переходов автомата Мили.

## **2.6 Составление полной спецификации устройства**

В таблице 1 приведено описание всех линий и сигналов, полученных в процессе разработки алгоритма работы АЛУ.

Таблица 1 – Линии и сигналы

|  |  |  |
| --- | --- | --- |
| Имя сигнала/шины и разрядность | Тип (In/Out) | Назначение сигнала |
| y0 | I для ОЧ | Сигнал о начале программы |
| y1 | I для ОЧ | Сигнал о записи числа A в регистр |
| y2 | I для ОЧ | Сигнал о записи числа B в регистр |
| y3 | I для ОЧ | Сигнал о получении управляющего сигнала U |
| y4 | I для ОЧ | Сигнал об установке флага P = 0 |
| y5 | I для ОЧ | Сигнал об установке флага OV = 0 |
| y6 | I для ОЧ | Сигнал о записи 16 в счетчик i |
| y7 | I для ОЧ | Сигнал о записи в i-ый бит числа С модуля результата вычитания из i-го бита числа А, i-го бита числа В |
| y8 | I для ОЧ | Сигнал об уменьшении счетчика на 1 |
| y9 | I для ОЧ | Сигнал о записи в i-ый бит числа С его инвертированного значения |
| y10 | I для ОЧ | Сигнал о записи в число A результата его инверсии и прибавления 1 |
| y11 | I для ОЧ | Сигнал о записи в число B результата его инверсии и прибавления 1 |
| y12 | I для ОЧ | Сигнал о записи в число C суммы чисел А и B |
| y13 | I для ОЧ | Сигнал об установке флага OV = 1 |
| y14 | I для ОЧ | Сигнал о записи 16 в счетчик i |
| y15 | I для ОЧ | Сигнал о записи 0 в счётчик g |
| y16 | I для ОЧ | Сигнал об уменьшении счетчика на 1 |
| y17 | I для ОЧ | Сигнал об увеличении счётчика g на 1 |
| y18 | I для ОЧ | Сигнал о записи 16 в счетчик i |
| y19 | I для ОЧ | Сигнал об установке флага P = 1 |
| y20 | I для ОЧ | Сигнал о выводе числа C в регистр |
| y21 | I для ОЧ | Сигнал о завершении программы |
| x1 | O для ОЧ | Сигнал о проверке числа U на равенство 0 |
| x2 | O для ОЧ | Сигнал о проверке значения счетчика на равенство 0 |
| x3 | O для ОЧ | Сигнал о проверке числа U на равенство 1 |
| x4 | O для ОЧ | Сигнал о проверке 0-го бита числа A на равенство 0 |
| x5 | O для ОЧ | Сигнал о проверке 0-го бита числа B на равенство 0 |
| x6 | O для ОЧ | Сигнал о проверке i-го бита числа C на неравенство 0 |
| x7 | O для ОЧ | Сигнал о проверке значения счетчика на не равенство 1 |
| x8 | O для ОЧ | Сигнал о проверке i-го бита числа g на равенство 0 |

Схема структуры устройства на микропрограммном уровне представлена на рисунке 8.

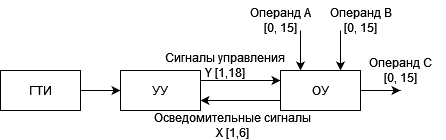


Рисунок 8 – Структура устройства на микропрограммном уровне

Таким образом, была составлена полная спецификация устройства, а также построена схема структуры устройства на микропрограммном уровне.

## **2.7 Временная диаграмма работы УУ**

На таблице 2 приведена временная диаграмма работы автомата при выполнении одной итерации контрольного примера конъюнкции.

Таблица 2 – Временная диаграмма конъюнкции.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ГТИ |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| y0 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| y1 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| y2 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| y3 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| y4 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| y5 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| y6 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| y7 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| y8 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| y9 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| y10 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| y11 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| y12 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| y13 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| y14 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| y15 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| y16 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| y17 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| y18 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| y19 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| y20 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| y21 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| x1 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| x2 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| x3 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| x4 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| x5 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| x6 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| x7 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| x8 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

На таблице 3 приведена временная диаграмма работы автомата при выполнении одной итерации контрольного примера сложения/вычитания.

Таблица 3 – Временная диаграмма сложения/вычитания.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ГТИ |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| y0 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| y1 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| y2 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| y3 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| y4 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| y5 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| y6 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| y7 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| y8 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| y9 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| y10 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| y11 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| y12 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| y13 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| y14 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| y15 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| y16 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| y17 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| y18 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| y19 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| y20 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| y21 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| x1 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| x2 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| x3 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| x4 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| x5 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| x6 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| x7 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| x8 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

* 1. Контрольный пример

На рисунке 8 изображен интерфейс учебной модели ЭВМ с разработанной программой АЛУ. В примере выполнена операция «конъюнкция» для чисел 25678 и 12345. Результатом является значение --11144. Переполнения нет, результат ненулевой.



Рисунок 8 – Интерфейс учебной модели ЭВМ

Активные функциональные элементы выделены цветом:

- красный – установка управляющего сигнала U (0 – конъюнкция, 1 – сложение, -1 – вычитание);

- коричневый – флаг OV (0 – переполнения нет, 1 – переполнение есть);

- оранжевый – флаг P (0 – результат ненулевой, 1 – результат нулевой);

- зелёный – поля ввода операнд A и B в десятичной форме, где первый символ является знаковым (0 – положительное число, 1 – отрицательное число);

- желтый – двоичное представление введенных операнд (16-битовая форма, где старший бит – знаковый), такой форме значения поступают на вход АЛУ;

- серый – разбиение поступающих значений на отдельные биты, перевод в ДК при необходимости;

- фиолетовый – вычисление результата в ячейках по битам;

- синий – получение результата в виде 16-битовой формы, где старший бит знаковый и десятичной формы того же значения.

- розовый – представление результат в десятичной форме.

Контрольные примеры работы разработанного АЛУ представлены в приложении Б.

# ЗАКЛЮЧЕНИЕ

В процессе выполнения курсовой работы была разработана арифметико-логическая единица (АЛУ), способная осуществлять определённый набор операций с использованием автомата Мили. Проведённый анализ требований к устройству, а также создание временных диаграмм позволили глубже понять принципы работы АЛУ. Разработка структуры и алгоритмов, а также спецификаций на уровне «черного ящика» помогли создать эффективное и корректное устройство, отвечающее поставленным задачам.

Результаты контрольного примера подтвердили верность функционирования созданной АЛУ и её соответствие исходным данным и требованиям. Исследование углубило знания в области цифровых вычислительных систем и проектирования управляющих автоматов.

# СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ

1. Жмакин, А. П. Архитектура ЭВМ: 2-е изд., перераб. и доп.: учеб. пособие. — СПб.: БХВ-Петербург, 2010. — 352 с.
2. Баранов С. И. Синтез микропрограммных автоматов. — Л.: Энергия, 1974. – 216 с.

## **ПРИЛОЖЕНИЕ А**

## **Листинг программы**

RD #390

WR R0

RD #403

WR R1

CALL BIN

RD #391

WR R0

RD #413

WR R1

CALL BIN

RD #403

WR R0

RD #435

WR R1

CALL Intermediate

RD #413

WR R0

RD #455

WR R1

CALL Intermediate

RD #0

WR R9

IN

JZ ZeroInput

JNS PositiveInput

JS NegativeInput

ZeroInput:

RD #460

WR R0

RD #420

WR R1

RD #440

WR R2

CALL NQ

JMP OutputPhase

PositiveInput:

RD #420

WR R0

CALL DK

RD #440

WR R0

CALL DK

RD #475

WR R0

RD #435

WR R1

RD #455

WR R2

CALL SUM

RD #460

WR R0

CALL DK

JMP OutputPhase

NegativeInput:

RD #440

WR R0

RD #1

SUB @R0

WR @R0

JMP PositiveInput

OutputPhase:

RD 475

JNZ OutputPhase\_1

RD #1

WR R9

OutputPhase\_1:

RD #460

WR R0

RD #480

WR R1

CALL OUTPUT

RD #475

WR R0

RD #490

WR R1

CALL DEC

OUT

HLT

Intermediate:

RD #4

WR R2

I1:

RD @R0

WR R3

RD #4

WR R4

RD #1

WR R5

I2: RD R3

DIV R5

I21:

SUB #100

JNS I21

ADD #100

I22:

SUB #10

JNS I22

ADD #10

WR @R1

RD R1

SUB #1

WR R1

RD R5

MUL #10

WR R5

RD R4

SUB #1

WR R4

JNZ I2

RD R0

SUB #1

WR R0

RD R2

SUB #1

WR R2

JNZ I1

RET

DK:

RD #15

WR R1

RD @R0

WR R2

JZ DK\_OUT

DK1:

RD R1

SUB #1

WR R1

RD R0

ADD #1

WR R0

RD #1

SUB @R0

WR @R0

RD R1

JNZ DK1

RD #16

WR R1

DK2:

RD @R0

JNZ DK21

RD #1

WR @R0

RD #0

WR R2

JMP DK22

DK21:

RD #0

WR @R0

DK22:

RD R0

SUB #1

WR R0

RD R1

SUB #1

WR R1

JZ DK\_OUT

RD R2

JZ DK\_OUT

JMP DK2

DK\_OUT:

RD R2

JZ DK\_OUT1

RD #1

WR R8

DK\_OUT1:

RET

SUM:

RD #16

WR R3

RD #0

WR R6

SUM1:

RD @R1

ADD @R2

ADD R6

WR R4

DIV #2

MUL #2

WR R5

RD R4

SUB R5

WR @R0

RD R5

DIV #2

WR R6

RD R0

SUB #1

WR R0

RD R1

SUB #1

WR R1

RD R2

SUB #1

WR R2

RD R3

SUB #1

WR R3

RD R3

JZ SUM\_OUT

RD #1

WR R8

JMP SUM1

SUM\_OUT:

RD R1

ADD #1

WR R1

RD R2

ADD #1

WR R2

RD R0

ADD #1

WR R0

RD @R1

ADD @R2

WR R7

SUB #1

JZ SUM\_OUT1

ADD @R0

JS SUM\_OUT1

SUB #2

JZ SUM\_OUT1

SUM\_OUT1:

RET

NQ:

RD #16

WR R3

NQ1:

RD @R1

SUB @R2

JZ ONE

RD #0

JMP NQ11

ONE:

RD @R1

NQ11:

WR @R0

RD R0

ADD #1

WR R0

RD R1

ADD #1

WR R1

RD R2

ADD #1

WR R2

RD R3

SUB #1

WR R3

JZ NQ\_OUT

JMP NQ1

NQ\_OUT:

RET

OUTPUT:

RD #4

WR R2

OUTPUT\_1:

RD #4

WR R3

RD #0

WR @R1

OUTPUT\_2:

RD @R1

MUL #10

ADD @R0

WR @R1

JZ OUTPUT\_3

OUTPUT\_3:

RD R0

ADD #1

WR R0

RD R3

SUB #1

WR R3

JNZ OUTPUT\_2

RD R1

ADD #1

WR R1

RD R2

SUB #1

WR R2

JNZ OUTPUT\_1

RET

BIN:

RD #4

WR R2

RD @R0

WR R7

JNS BIN\_1

RD #1

WR R4

RD #0

SUB #1

MUL R7

WR R7

BIN\_1:

RD #4

WR R3

RD #1

WR R5

RD #0

WR @R1

BIN\_2:

RD R7

DIV #2

MUL #2

WR R6

RD R7

SUB R6

MUL R5

ADD @R1

WR @R1

RD R7

DIV #2

WR R7

RD R5

MUL #10

WR R5

RD R3

SUB #1

WR R3

JNZ BIN\_2

RD R1

SUB #1

WR R1

RD R2

SUB #1

WR R2

JNZ BIN\_1

RD R4

JZ BIN\_OUT

RD R5

DIV #10

WR R5

RD R1

ADD #1

WR R1

RD @R1

ADD R5

WR @R1

BIN\_OUT:

RD R7

JZ BIN\_OUT\_1

RD #1

WR R8

BIN\_OUT\_1:

RET

DEC:

RD #15

WR R2

RD #0

WR @R1

RD #1

WR R3

DEC\_1:

RD @R0

MUL R3

ADD @R1

WR @R1

JZ DEC\_2

DEC\_2:

RD R3

MUL #2

WR R3

RD R0

SUB #1

WR R0

RD R2

SUB #1

WR R2

JNZ DEC\_1

RD @R0

JZ DEC\_3

RD #0

SUB #1

MUL @R1

WR @R1

DEC\_3:

RET

# ПРИЛОЖЕНИЕ Б Контрольные примеры

1. Сложение положительных чисел изображено на рисунке А.1.

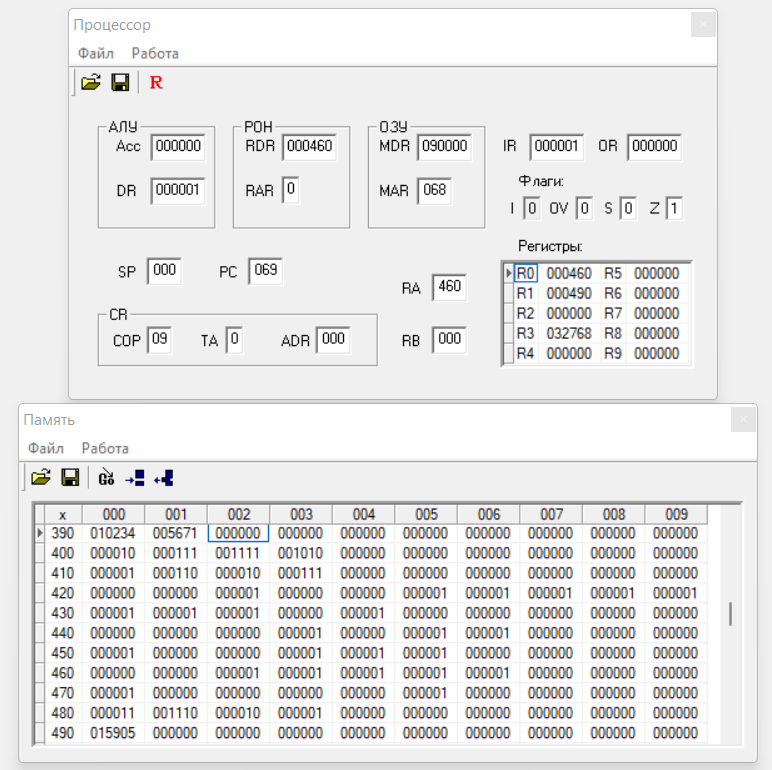


Рисунок А.1 – Сложение положительных чисел

1. Сложение чисел с разными знаками изображено на рисунке А.2.

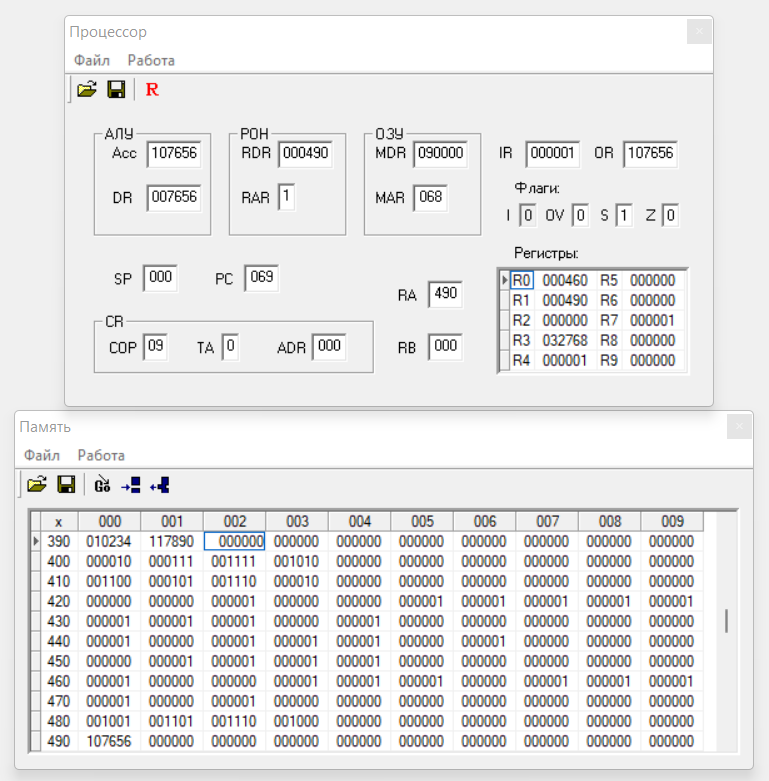


Рисунок А.2 – Сложение чисел с разными знаками

1. Вычитание положительного числа изображено на рисунке А.3.



Рисунок А.3 – Вычитание положительного числа

1. Вычитание отрицательного числа изображено на рисунке А.4.

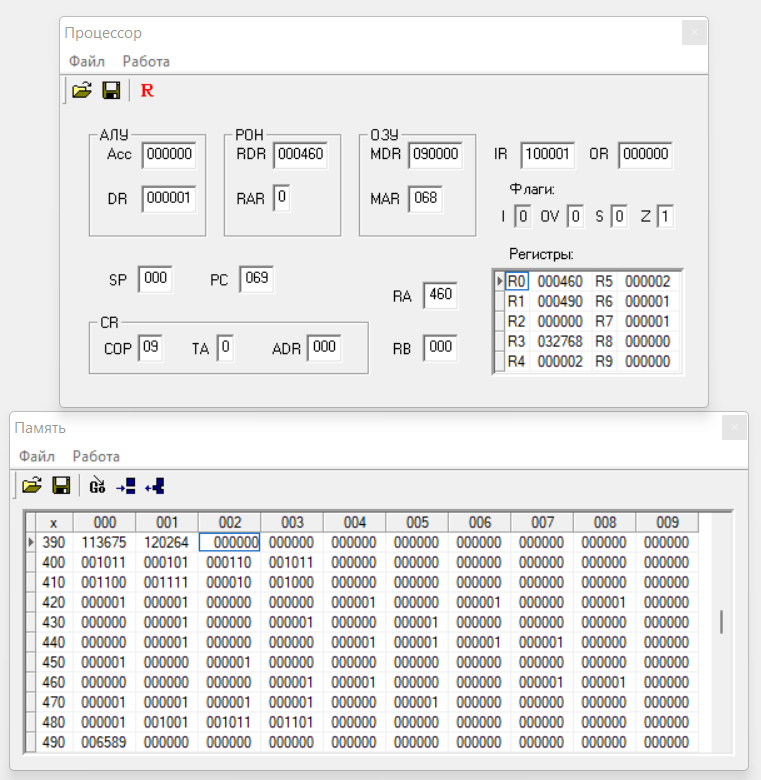


Рисунок А.4 – Вычитание отрицательного числа

1. Конъюнкция изображена на рисунке А.5.

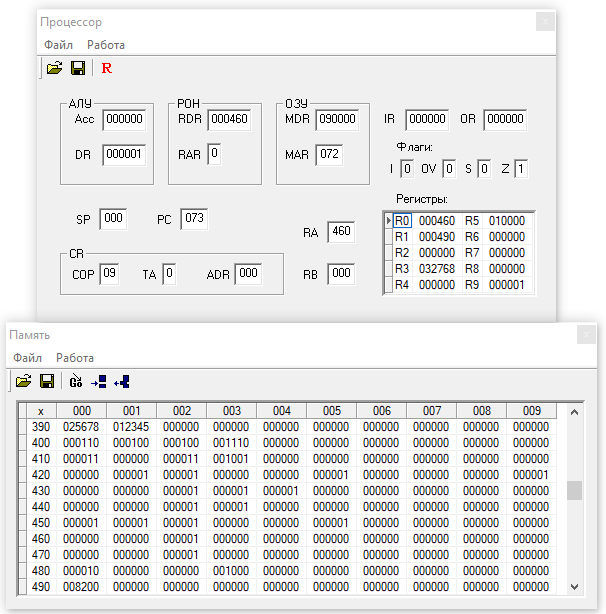


Рисунок А.5 –Конъюнкция

1. Проверка флага переполнения изображена на рисунке А.6.



Рисунок А.6 – Проверка флага переполнения

1. Проверка флага чётности числа единиц в результате изображена на рисунке А.7.

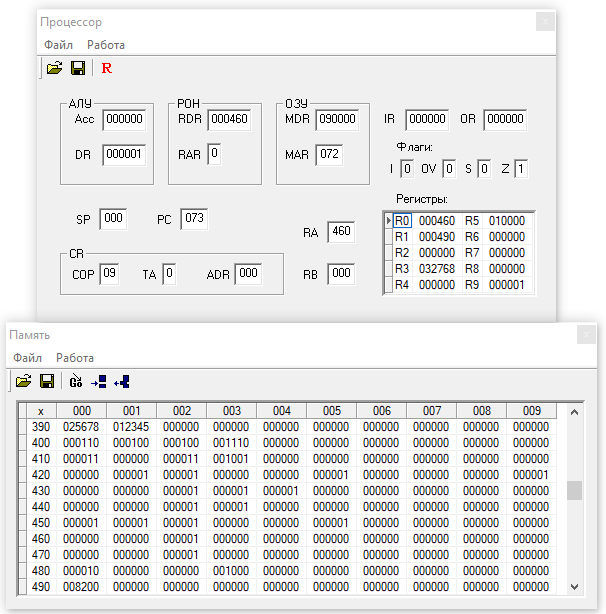


Рисунок А.7 – Проверка флага чётности числа единиц в результате