

МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ

Учреждение образования
«Витебский государственный технологический университет»

**АВТОМАТИЗАЦИЯ ТЕХНОЛОГИЧЕСКОГО
ОБОРУДОВАНИЯ ДЛЯ ТРЕХМЕРНЫХ ТЕХНОЛОГИЙ**

Методические указания по выполнению лабораторных работ
для студентов специальности
1-36 07 02 «Производство изделий на основе трехмерных технологий»

Витебск
2023

УДК 621.7

Составители:

А. Л. Климентьев, А. В. Котович

Рекомендовано к изданию редакционно-издательским
советом УО «ВГТУ», протокол № 10 от 29.06.2023.

Автоматизация технологического оборудования для трехмерных технологий : методические указания по выполнению лабораторных работ / сост. : А. Л. Климентьев, А. В. Котович. — Витебск : УО «ВГТУ», 2023. — 43 с.

Методические указания являются руководством по выполнению лабораторных работ по учебной дисциплине «Автоматизация технологического оборудования для трехмерных технологий». Изложены содержание, методика и порядок выполнения работ. Также приведен краткий справочник по основным командам управляющих программ и описан пример создания управляющей программы. Предназначены для студентов специальности 1-36 07 02 «Производство изделий на основе трехмерных технологий».

Издание в электронном виде расположено в репозитории библиотеки УО «ВГТУ».

УДК 621.7

© УО «ВГТУ», 2023

СОДЕРЖАНИЕ

ЛАБОРАТОРНАЯ РАБОТА 1. РАЗРАБОТКА СИСТЕМЫ УПРАВЛЕНИЯ НА БАЗЕ ЛОГИЧЕСКОГО АВТОМАТА	4
1.1 Элементы теории.....	4
1.2 Содержание работы	10
1.3 Варианты заданий.....	11
ЛАБОРАТОРНАЯ РАБОТА 2. РАЗРАБОТКА СИСТЕМЫ УПРАВЛЕНИЯ НА БАЗЕ ЛОГИЧЕСКОГО АВТОМАТА В БАЗИСЕ РЕЛЕЙНО- КОНТАКТНЫХ ЭЛЕМЕНТОВ	13
2.1 Элементы теории.....	13
2.2 Содержание работы	19
2.3 Варианты заданий.....	19
ЛАБОРАТОРНАЯ РАБОТА 3. ПРОГРАММИРОВАНИЕ ТЕХНОЛОГИЧЕСКОГО ОБОРУДОВАНИЯ С ПРОГРАММНЫМ УПРАВЛЕНИЕМ.....	20
3.1 Элементы теории.....	20
3.2 Порядок разработки управляющей программы.....	24
3.3 Содержание работы	30
СПИСОК РЕКОМЕНДУЕМЫХ ИСТОЧНИКОВ.....	31
ПРИЛОЖЕНИЕ А. Краткий справочник по основным командам	32

Лабораторная работа 1

РАЗРАБОТКА СИСТЕМЫ УПРАВЛЕНИЯ НА БАЗЕ ЛОГИЧЕСКОГО АВТОМАТА

Цель работы: изучение методов синтеза комбинационных логических схем путем минимизации переключательных функций и получение навыков структурного синтеза логических автоматов без памяти.

1.1 Элементы теории

Комбинационные логические схемы (КЛС) (автоматы без памяти) представляют собой частный случай цифровых автоматов, также являются составной частью структурной схемы цифровых автоматов с памятью. Синтез КЛС производится на основе законов алгебры-логики (и следствий из них) с помощью различных методов минимизации.

Основные положения и законы алгебры-логики (булевой алгебры).

Логические переменные (ЛП) — переменные, принимающие только два значения — 0 или 1. Другие названия логических переменных — двоичные или булевы переменные.

Логические переменные можно разделить на простые и сложные.

Простые ЛП — независимые переменные.

Сложные ЛП — зависимые переменные или функции ЛП, принимающие также значения 0 или 1, называются двоичными или переключательными функциями (ПФ). Для ПФ справедлив принцип суперпозиции.

Обычно ПФ является функцией конечного числа переменных аргументов, обозначаемых x_i , $i = \overline{1, n}$, а сама ПФ обозначается как $y = f(x_i, i = \overline{1, n})$.

Конечность числа переменных (n) и конечность принимаемых ими значений ($k = 2$) определяет конечность значений ПФ (m), которая определяется количеством сочетаний любого из 2-х значений каждой из n переменных. Каждое сочетание (комбинация) называется набором, а количество наборов определяется как $m = 2^n$.

Значение, принимаемое ПФ на конкретном наборе, называется *значением истинности*, а таблица, содержащая значения истинности ПФ на всех наборах, — *таблицей истинности* (ТИ).

ТИ является исходной формой для аналитической записи ПФ. Аналитическая запись ПФ предполагает (аналогично обычной математике) набор элементарных операций, называемых логическими связями, которые определяются как ПФ 2-х переменных, а затем на основе принципа суперпозиции позволяют построить сколь угодно сложную ПФ.

Для двух переменных возможны $2^2 = 4$ набора, на них можно определить $2^4 = 16$ различных логических связей (ЛС), т. е. ЛС, которые на основе принципа суперпозиции позволяют построить ПФ любой сложности, составляют функционально-полный набор ЛС (ФПН), определяемый совокупностью свойств, входящих в него ЛС.

Наиболее широкое распространение получил основной ФПН (ОФПН), включающий три ЛС из всего набора:

- 1) f_1 — отрицание (операция НЕ, логическая инверсия);
- 2) f_2 — конъюнкция (операция И, логическое умножение);
- 3) f_3 — дизъюнкция (операция ИЛИ, логическое сложение).

Таблица истинности операций ОФПН и их аналитическая запись.

x_1	x_2	f_1	f_2	f_3	f_4	f_5
0	0	1	0	0	1	1
0	1	1	0	1	1	0
1	0	0	0	1	1	0
1	1	0	1	1	0	0

$$f_1 = f(x_1) = \overline{x_1};$$

$$f_2 = f(x_1, x_2) = x_1 x_2 = x_1 \& x_2 = x_1 \wedge x_2;$$

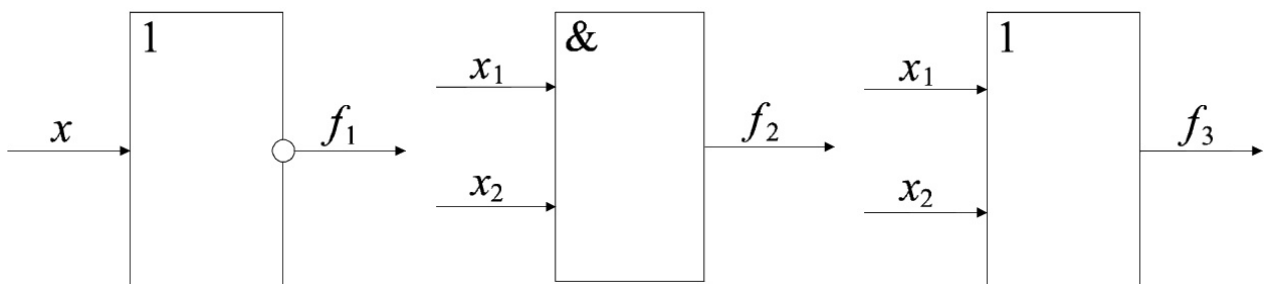
$$f_3 = f(x_1, x_2) = x_1 + x_2 = x_1 \vee x_2.$$

ОФП — набор избыточный, т. к. функционально-полными являются наборы ЛС (f_1, f_2) и (f_1, f_3) . ФПН представляют также функции f_4 и f_5 .

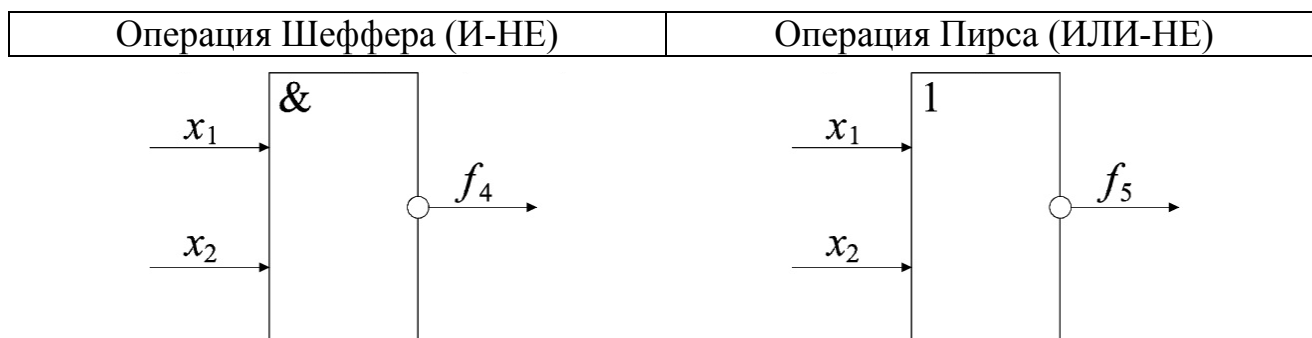
- f_4 — операция Шеффера, операция И-НЕ, $f_4 = \overline{x_1 x_2} = \overline{x_1} \& \overline{x_2} = x_1 \wedge x_2 = f_2$.
- f_5 — операция Пирса, операция ИЛИ-НЕ, $f_5 = \overline{x_1 + x_2} = \overline{x_1} \vee \overline{x_2} = \overline{f_3}$.

Свойства операций ОФПН и их обозначения на функциональных схемах.

Отрицание	Конъюнкция	Дизъюнкция
$\overline{\overline{x}} = x$ $\overline{\overline{\overline{x}}} = \overline{x}$ Если $x_1 = x_2$, то $\overline{x_1} = \overline{x_2}$	$xx \dots x = x$ $1 \cdot x = x$ $0 \cdot x = 0$ $x \cdot \overline{x} = 0$	$x + x + \dots + x = x$ $1 + x = 1$ $0 + x = x$ $x + \overline{x} = 1$



Обозначение на функциональных схемах операций Шеффера и Пирса.



Законы алгебры логики

	Переместительный	Сочетательный
И	$x_1 x_2 = x_2 x_1$	$x_1 x_2 x_3 = x_1 (x_2 x_3)$
ИЛИ	$x_1 + x_2 = x_2 + x_1$	$x_1 + x_2 + x_3 = x_1 + (x_2 + x_3)$
	Распределительный	Инверсия
И	$x_1 (x_2 + x_3) = x_1 x_2 + x_1 x_3$	$\overline{x_1 x_2} = \overline{x_1} + \overline{x_2}$
ИЛИ	$x_1 (x_2 x_3) = (x_1 + x_2)(x_1 + x_3)$	$\overline{x_1 + x_2} = \overline{x_1} \cdot \overline{x_2}$

Следствия:

а) правило старшинства логических операций:

- отрицание;
- конъюнкция;
- дизъюнкция;

б) правило склеивания $x_1 x_2 + x_1 \overline{x_2} = x_1$; $x_1 x_2 x_3 = x_1 x_2 \overline{x_3} = x_1 x_2$;

в) правило поглощения $x_1 + x_1 x_2 = x_1$.

Конъюнкция (дизъюнкция) называются *элементарными*, если выполняются над конечным числом одиночных аргументов или их отрицаний. Например: $x_1 \overline{x_2} x_3 x_4$, $x_1 + \overline{x_2} + \overline{x_3} + x_4$.

Количество аргументов в элементарной конъюнкции (дизъюнкции) — ранг r .

Две элементарные конъюнкции (дизъюнкции) с одинаковым рангом называются *соседними*, если отличаются знаком отрицания только одного из аргументов. Например: $x_1 \cdot \overline{x_2} \cdot \overline{x_3} \cdot x_4$, $x_1 \cdot x_2 \cdot \overline{x_3} \cdot x_4$; $(x_1 + x_2 + x_3 + x_4, x_1 + x_2 + x_3 + \overline{x_4})$, к ним применимо правило склеивания (они склеиваются).

Если $r = n$, где n — количество аргументов в ПФ $f = f(x_i, i = \overline{1, n})$, то элементарная конъюнкция (дизъюнкция) называется *конституантой* единицы (нуля).

Очевидно, что конституанта сохраняет единицу (нуль) только на одном наборе значений, входящих в нее аргументов из таблицы истинности. На этом правиле основаны способы (формы) аналитической записи переключательных функций по таблице истинности.

Совершенная дизъюнктивная нормальная форма (СДНФ)

Для наборов, дающих единицу переключательной функции, записываются конstituанты единицы, в которых переменные, имеющие на наборе нулевое значение, берутся с отрицанием (например, для набора $x_1 = 0, x_2 = 1, x_3 = 0$ имеем конstituанту $\overline{x_1} \cdot x_2 \cdot \overline{x_3}$) и объединяются знаком дизъюнкции.

Совершенная конъюнктивная нормальная форма (СКНФ)

Симметрично конstituанты нуля, составленные для наборов, дающих нулевое значение ПФ, объединяются знаком конъюнкции. С отрицанием в конstituантах нуля берутся переменные, имеющие на наборе единичное значение (например, для набора $x_1 = 0, x_2 = 1, x_3 = 0$ имеем конstituанту $x_1 + \overline{x_2} + x_3$).

Пример.

СДНФ для рассматриваемых функций

$$f_2 = x_1 x_2;$$

$$f_3 = \overline{x_1} x_2 + x_1 \overline{x_2} + x_1 x_2;$$

$$f_4 = \overline{x_1} \overline{x_2} + x_1 \overline{x_2} + \overline{x_1} x_2;$$

$$f_5 = \overline{x_1} x_2;$$

для них же СКНФ

$$f_2 = (x_1 + x_2)(x_1 + \overline{x_2})(\overline{x_1} + x_2);$$

$$f_3 = x_1 + x_2;$$

$$f_4 = \overline{x_1} + \overline{x_2};$$

$$f_5 = (x_1 + \overline{x_2})(\overline{x_1} + x_2)(\overline{x_1} + \overline{x_2}).$$

В результате преобразований СДНФ (СКНФ) на основании законов алгебры логики ранг некоторых конъюнкций (дизъюнкций), входящих в формулу может быть понижен. Такие формы называются, соответственно, дизъюнктивной (ДНФ) и конъюнктивной (КНФ) нормальной формой.

Минимизация переключаяющих функций

Одним из критериев минимизации комбинационных логических схем является критерий минимума аргументов, записанных в форме, иначе критерий минимума входов (логической переменной соответствует вход логической схемы), а также минимума элементарных конъюнкций (дизъюнкций) в форме.

Минимизация производится путем:

1) многократного попарного склеивания конъюнкций (дизъюнкций), в результате чего получается Д(К)НФ, называемая *сокращенной*, а ее составляющие, попарно не склеиваемые, — простыми *импликантами*;

2) последующего исключения лишних импликант, удаление которых не влияет на истинность ПФ, в результате чего получается «тупиковая форма», дальнейшая минимизация которой в рамках нормальных форм невозможна.

Дальнейшая минимизация возможна путем применения законов инверсии и распределительного, нарушающих ПФ. Этот путь не формализован и основан на интуиции и опыте разработчика (интуитивно-эвристический путь).

Одним из методов, включающим в себя два первых этапа является табличный метод (метод диаграмм Вейча–Карно). Он использует специальные таблицы — диаграммы Вейча–Карно, заполненные в соответствии с таблицей истинности, преобразуемые по специальным правилам.

Таблица прямоугольной формы состоит из клеток, количество которых определяется количеством переменных в ПФ и равно 2^n . Для четных n каждая сторона прямоугольника образуется из $2^{n/2}$ клеток, а для нечетных n — одна сторона из $2^{(n-1)/2}$; а другая из $2^{(n+1)/2}$ клеток.

Каждой малой клетке соответствует свой набор значений переменных из таблицы истинности (количество наборов $m = 2^n$) и располагаются наборы в клетках диаграммы так, чтобы в соседних клетках находились соседние наборы. Крайние на верхнем и нижнем, а также на правом и левом полях клетки также считаются соседними.

Диаграммы Вейча – Карно

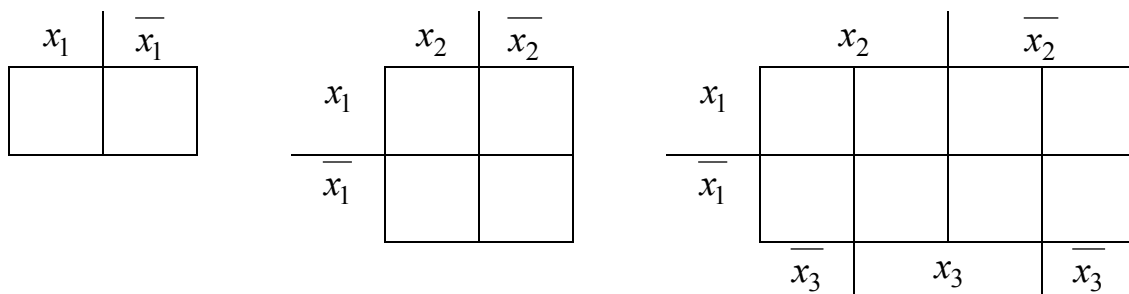
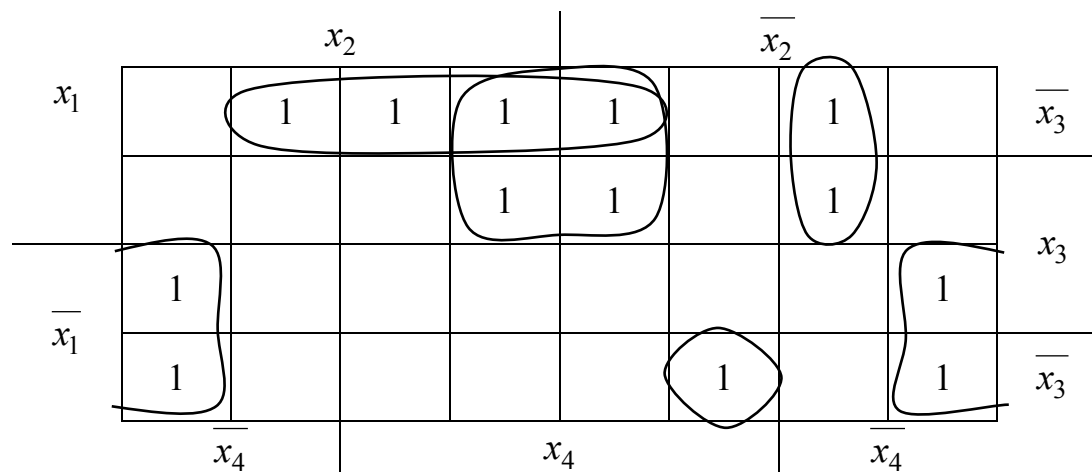


Диаграмма Вейча – Карно заполняется значениями таблицы истинности, соответствующими наборам, стоящими в клетках. Причем, если минимизируется СДНФ, то записываются только 1, а нулевые клетки остаются пустыми. Для СКНФ записываются только нули ПФ.

Некоторые единицы (нули) в таблице стоят в соседних клетках. Любое 2^s число соседних единиц (нулей) может быть обведено контуром.

Например,



Количество контуров определяет количество простых импликант в полученной из диаграммы «тупиковой форме».

2^s соседних клеток, составляющих контур, соответствуют одной импликанте, ранг которой меньше ранга n константы на s единиц. Это означает что чем больше клеток в контуре, тем проще член тупиковой формы, полученных из него; чем меньше контуров, тем меньше импликант содержит «тупиковая форма».

Если диаграмма для n переменных содержит q контуров, в каждом из которых 2^{s_j} , $j = \overline{1, q}$ клеток, то тупиковая форма позволит создать схему, состоящую из логических элементов с суммарным количеством входов:

$$C = \sum_{j=1}^q (n - s_j) + q.$$

Величина C является ценой схемы.

Таким образом, не выписывая «тупиковой формы», можно сравнить по цене диаграммы, составленные, соответственно, для СДНФ и СКНФ и выбрать минимальную из них.

Выписыванием «тупиковой формы» заканчивается процесс формальной минимизации.

Структурный синтез автоматов без памяти

Структурный синтез автоматов без памяти состоит из 4-х этапов:

1. Составление таблицы истинности синтезируемого автомата на основе анализа его назначения и (словесного) описания принципа работы. Таблица истинности заполняется путем выяснения отклика ПФ на каждый набор переменных.

2. Получение математической формулы для логической функции, возможно оптимизированной по какому-либо критерию, например, по минимуму цены схемы.

- записать аналитическое выражение в СДНФ и СКНФ булевой функции;
 - минимизировать выражение ПФ в СДНФ (методом диаграмм Вейча – Карно), определить цену схемы;
 - то же для СКНФ.
4. Для МДНФ построить релейно-контактную схему, реализующую ПФ в СДНФ и СКНФ.
5. Привести выражение МДНФ, реализующую ПФ в СДНФ и СКНФ, к базисам Шеффера и Пирса (соответственно).
6. Построить логическую схему в базисе Шеффера и логическую схему в базисе Пирса.

1.3 Варианты заданий

Таблица 1.1 — Варианты заданий

Состояния входов			Состояния выхода у (по вариантам)									
x_1	x_2	x_3	1	2	3	4	5	6	7	8	9	10
0	0	0	1	1	1	1	1	0	1	0	0	1
0	0	1	1	0	1	0	0	1	0	0	1	0
0	1	0	1	0	0	0	1	1	0	1	1	1
0	1	1	0	0	1	1	0	1	1	1	0	0
1	0	0	1	1	0	1	0	1	0	0	0	0
1	0	1	0	1	0	0	1	0	1	0	0	0
1	1	0	0	0	0	1	0	0	0	1	1	1
1	1	1	0	0	1	0	0	0	1	0	0	1

Состояния входов			Состояния выхода у (по вариантам)									
x_1	x_2	x_3	11	12	13	14	15	16	17	18	19	20
0	0	0	0	1	1	0	0	1	0	1	0	0
0	0	1	1	0	0	1	1	0	0	1	0	1
0	1	0	1	0	1	0	1	0	0	0	1	0
0	1	1	1	1	0	0	0	1	1	0	1	1
1	0	0	0	1	0	0	0	1	1	1	0	1
1	0	1	0	0	1	1	1	0	0	0	0	1
1	1	0	0	1	1	1	0	0	1	1	1	0
1	1	1	1	1	0	1	1	1	1	0	1	0

Окончание таблицы. 1.1

Состояния входов			Состояния выхода y (по вариантам)									
x_1	x_2	x_3	21	22	23	24	25	26	27	28	29	30
0	0	0	0	0	1	1	0	1	0	0	1	0
0	0	1	1	1	0	0	1	0	1	0	1	0
0	1	0	1	1	1	0	0	1	0	0	1	0
0	1	1	1	0	0	1	0	0	1	0	1	1
1	0	0	0	1	0	1	1	1	0	1	0	0
1	0	1	0	0	1	0	0	0	1	1	0	1
1	1	0	1	0	0	0	1	1	0	1	0	1
1	1	1	0	1	1	1	1	0	1	1	0	1

Состояния входов				Состояния выхода (по вариантам)									
x_1	x_2	x_3	x_4	y_1	y_2	y_3	y_4	y_5	y_6	y_7	y_8	y_9	y_{10}
0	0	0	0	0	0	1	0	1	0	0	1	1	—
0	0	0	1	0	0	0	0	0	—	1	1	—	1
0	0	1	0	0	1	1	1	1	1	0	1	1	0
0	0	1	1	—	0	0	1	—	0	—	1	0	0
0	1	0	0	1	0	—	0	0	1	0	0	0	—
0	1	0	1	0	—	0	—	1	0	0	—	—	1
0	1	1	0	1	0	0	0	1	0	0	0	0	0
0	1	1	1	0	0	0	1	1	1	0	—	0	1
1	0	0	0	1	1	—	0	0	0	1	1	0	0
1	0	0	1	—	0	1	—	—	1	—	0	1	1
1	0	1	0	—	1	1	1	—	1	0	1	—	0
1	0	1	1	1	—	—	—	1	—	1	—	0	—
1	1	0	0	0	1	—	—	—	—	—	0	—	—
1	1	0	1	—	—	0	1	0	1	0	0	1	1
1	1	1	0	1	0	1	0	—	0	1	—	—	—
1	1	1	1	—	0	1	1	1	1	—	1	1	0

Лабораторная работа 2

РАЗРАБОТКА СИСТЕМЫ УПРАВЛЕНИЯ НА БАЗЕ ЛОГИЧЕСКОГО АВТОМАТА В БАЗИСЕ РЕЛЕЙНО-КОНТАКТНЫХ ЭЛЕМЕНТОВ

Цель работы: получение навыков структурного синтеза логических автоматов без памяти и реализации системы управления на базе релейно-контактных элементов.

2.1 Элементы теории

Для разработки и моделирования работы схемы управления можно использовать программного обеспечения для виртуального моделирования и симуляции FluidSIM (FESTO). FluidSIM представляет собой комплексное программное обеспечение для создания, моделирования, обучения и изучения электропневматических, электрогидравлических, цифровых и электронных схем. FluidSIM сочетает в себе интуитивно понятный редактор принципиальных схем с подробным описанием компонентов, их фотографиями и соответствующей анимацией.

Для запуска FluidSIM можно воспользоваться ярлыком — Все программы / Festo Didactic / FluidSIM Pneumatics V 4.2 Student Version.

В результате откроется основное окно редактора FluidSIM (рис. 2.1).

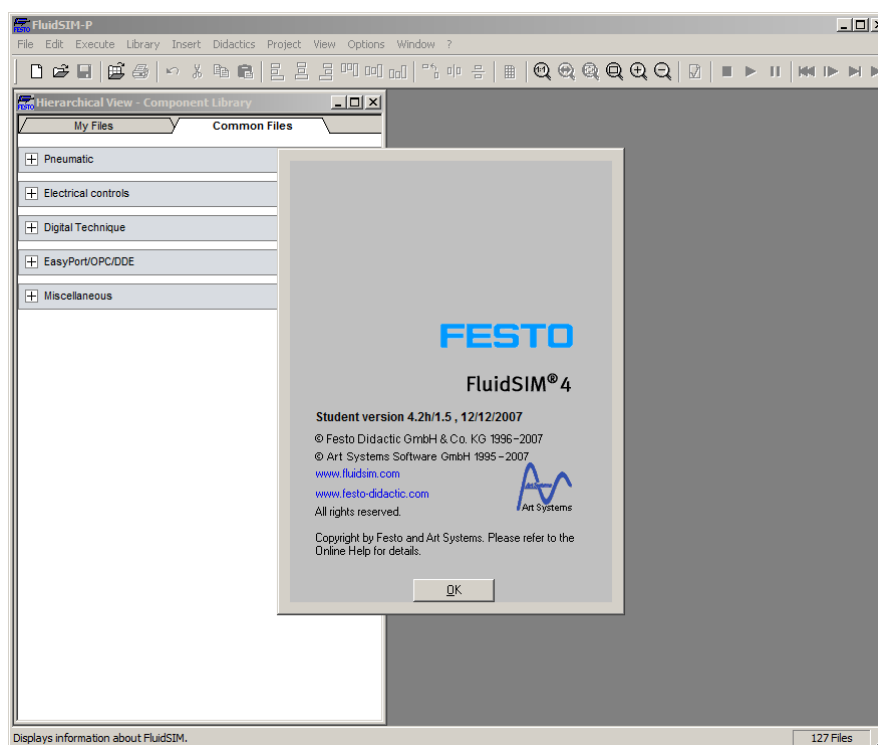


Рисунок 2.1 — Окно редактора FluidSIM (с окном «О программе»)

Основное окно редактора FluidSIM имеет интерфейс, который в том числе содержит строку меню и панель инструментов (рис. 2.2).

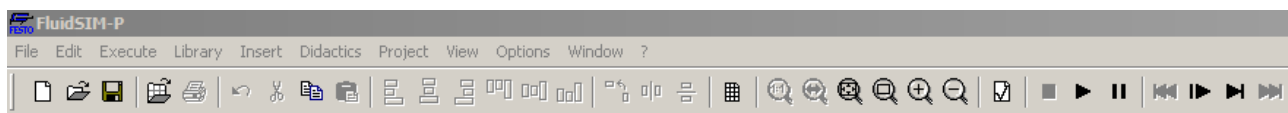


Рисунок 2.2 — Строка меню и панель инструментов редактора FluidSIM

Для создания схемы необходимо создать новый файл (рабочее пространство). При этом для удобства можно воспользоваться привязкой элементов по сетке (рис. 2.3), отображение которой можно включить в ленте инструментов.

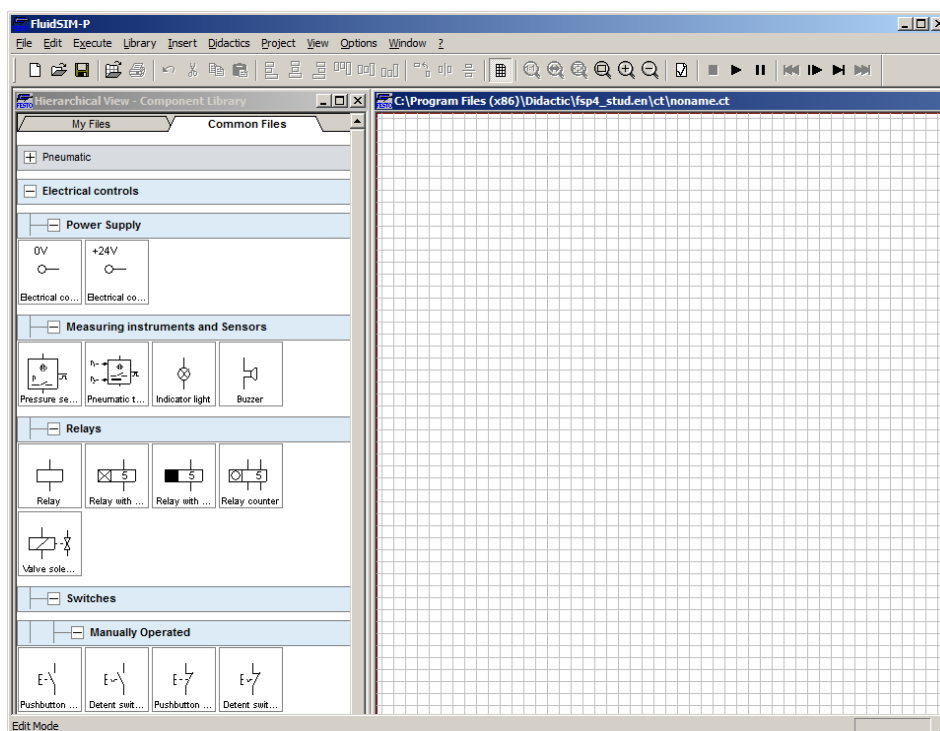


Рисунок 2.3 — Окно редактора FluidSIM (с отображением сетки для привязки элементов в рабочем пространстве)

Для разработки схемы управления необходимо использовать элементы из библиотеки компонентов (рис. 2.4).

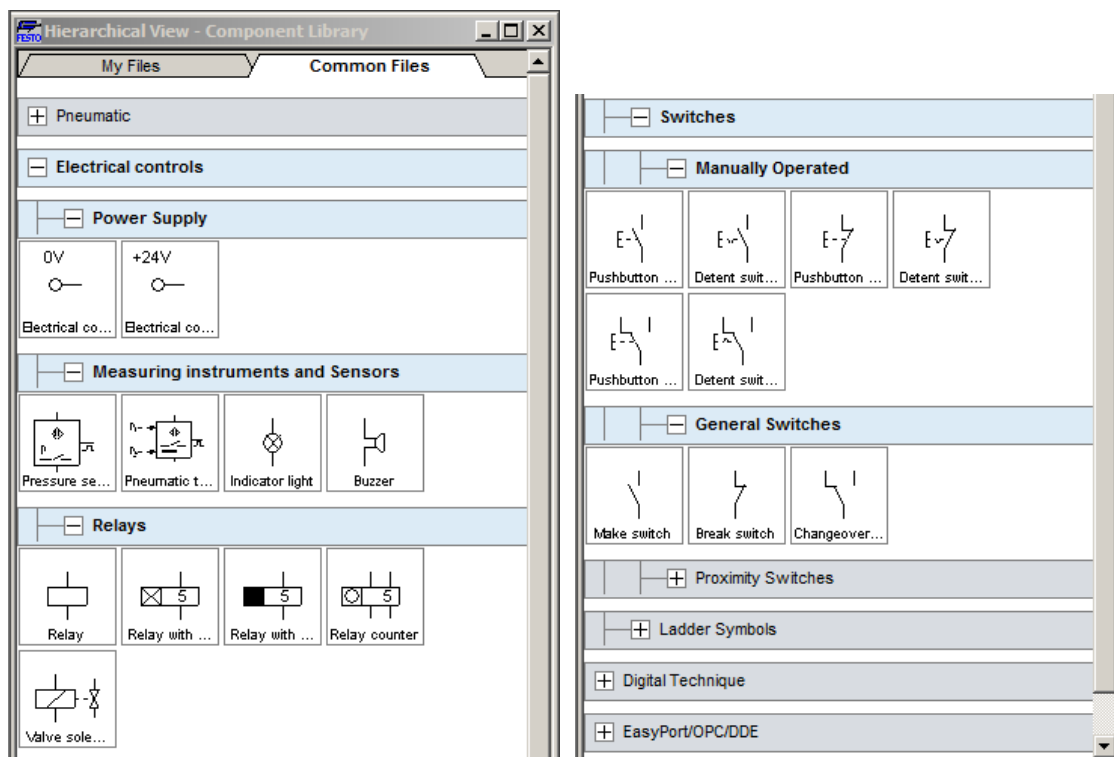
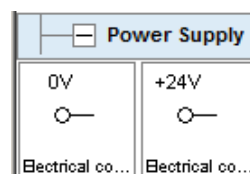


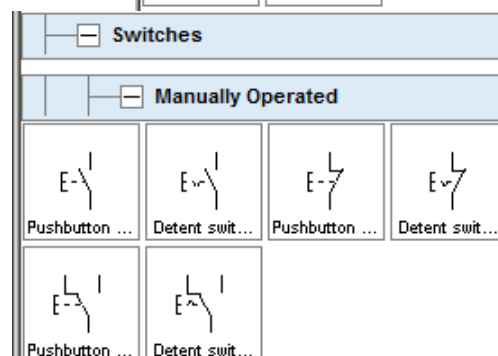
Рисунок 2.4 — Фрагменты окна библиотеки компонентов редактора FluidSIM (иерархическое представление компонентов)

Для разрабатываемой схемы управления необходимо использовать соответствующие компоненты.

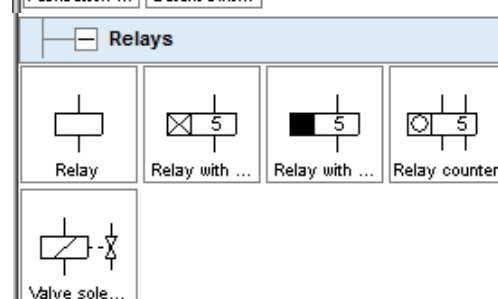
Источник питания /
Power Supply



Переключатели с ручным управлением /
Switches Manually Operated

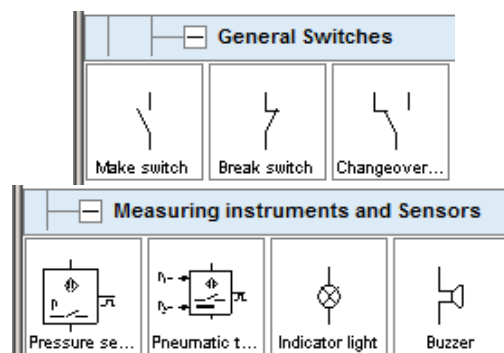


Реле /
Relay



Общие переключатели / General Switches

Измерительные инструменты и Сенсоры / Measuring instruments and Sensors



Для рассматриваемого примера по логической функции $f = x_1 + x_2 x_3 + x_2 x_3$ схема управления на базе релейно-контактных элементов должна содержать (рис. 2.5):

- 1 нормально-замкнутую кнопку с фиксацией положения (Вкл.);
- 1 нормально-разомкнутую кнопку с фиксацией положения (СТОП);
- 3 кнопки без фиксации положения с соответствующими 3 реле (X1–X3);
- 1 реле для исполнительной цепи (Y);
- компоненты исполнительной цепи в составе 1 общего переключателя (Y), 1 светового индикатора (Indicator light) и зуммера (Buzzer);
- некоторое количество общих переключателей для реализации собственно цепи управления в виде соответствующего количества нормально-замкнутых и нормально-разомкнутых общих переключателей.

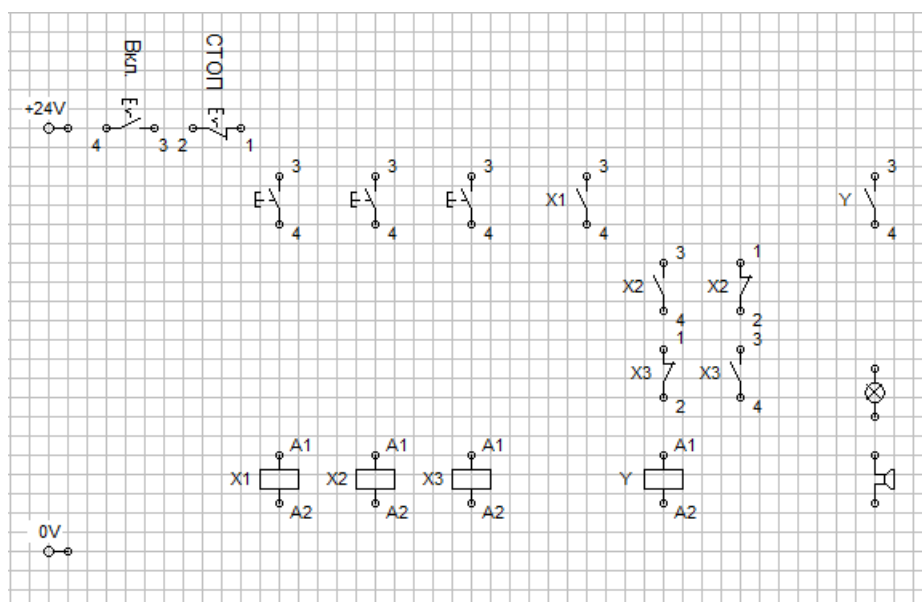


Рисунок 2.5 — Компоненты схемы управления в окне редактора FluidSIM
(с выравниванием компонентов по сетке)

Необходимо отметить, что для привязки реле и основных переключателей необходимо для реле и соответствующих переключателей добавить соответствующую метку (Label), для чего, в свою очередь, необходимо

двойным кликом ЛКМ на компоненте вызвать соответствующее окно (рис. 2.6).

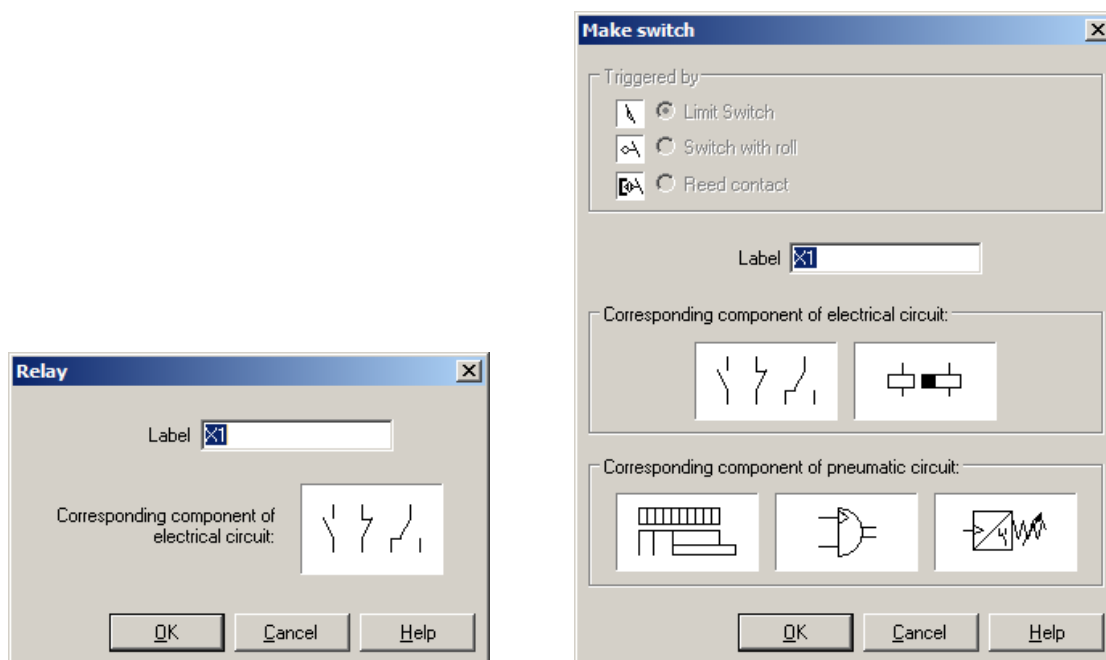


Рисунок 2.5 — Окна для ввода метки (Label) для реле X1 (Relay) и одного из основных переключателей (Switch)

Для соединения размещенных в рабочем пространстве компонентов необходимо нажав и удерживая ЛКП курсором соединить соответствующий контакт (на схеме имеют нумерацию 1–4 или обозначение A1, A2) выбранного компонента вначале с источником питания, а затем с полученными линиями соединений (рис. 2.6).

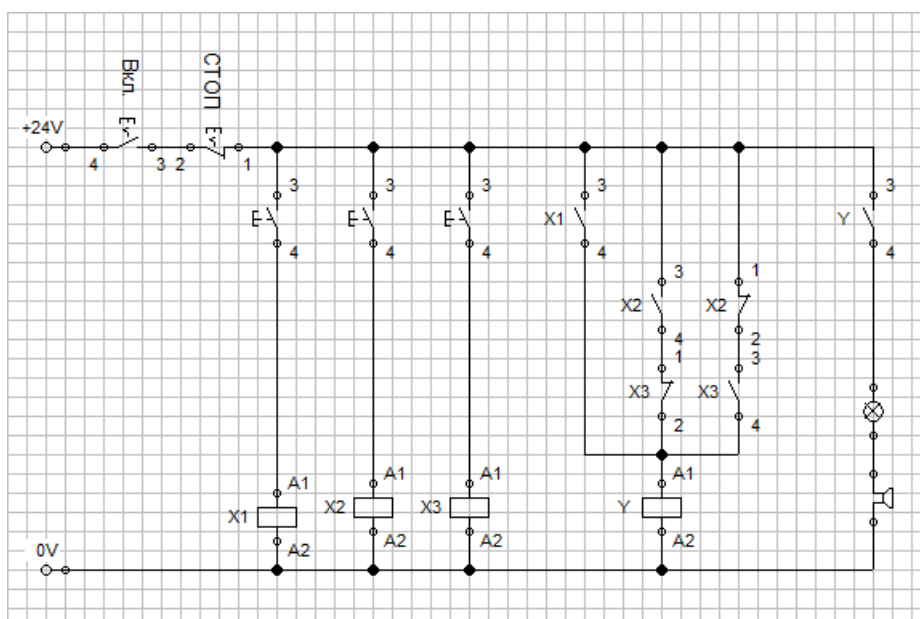


Рисунок 2.6 — Схема управления, полученная в результате соединения компонентов в рабочем пространстве

Окончательно схема управления состоит из источника питания, имитатора вводного выключателя «Вкл.», имитатора кнопки аварийного выключения «СТОП», цепей имитации входов (а), цепи управления (б) и исполнительной цепи (в), представленных на рисунке 2.7.

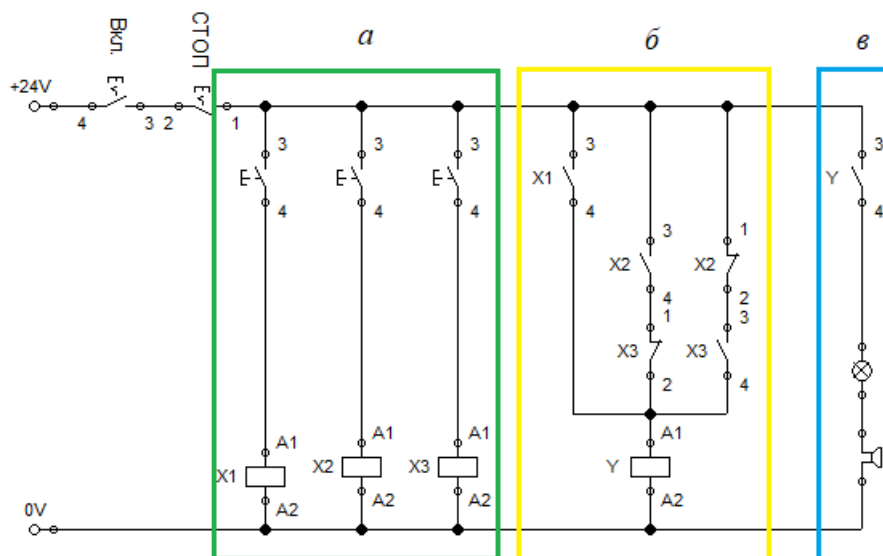
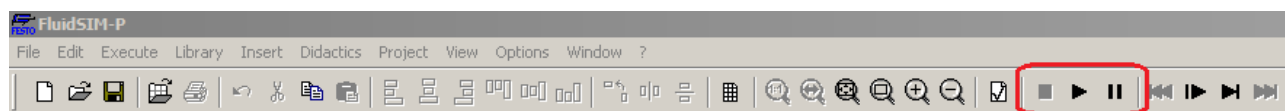


Рисунок 2.7 — Схема управления для логической функции $f = x_1 + x_2 \overline{x_3} + \overline{x_2} x_3$

Программное обеспечение FluidSIM позволяет выполнить симуляцию работы схемы для чего необходимо воспользоваться соответствующими инструментами/командами в меню или панели инструментов.



Обязательным условием выполнения работы является выполнение валидации разработанной схемы управления. Для этого необходимо в режиме симуляции выполнить перебор всех условий по таблице состояний в соответствии с вариантом заданий и проверить соответствие состояния исполнительной цепи (включено/выключено). Для имитации работы входов использовать соответствующие кнопки в цепях реле X1, X2, X3. Состояние исполнительной цепи контролируется включением светового индикатора и зуммера, рисунок 2.7.

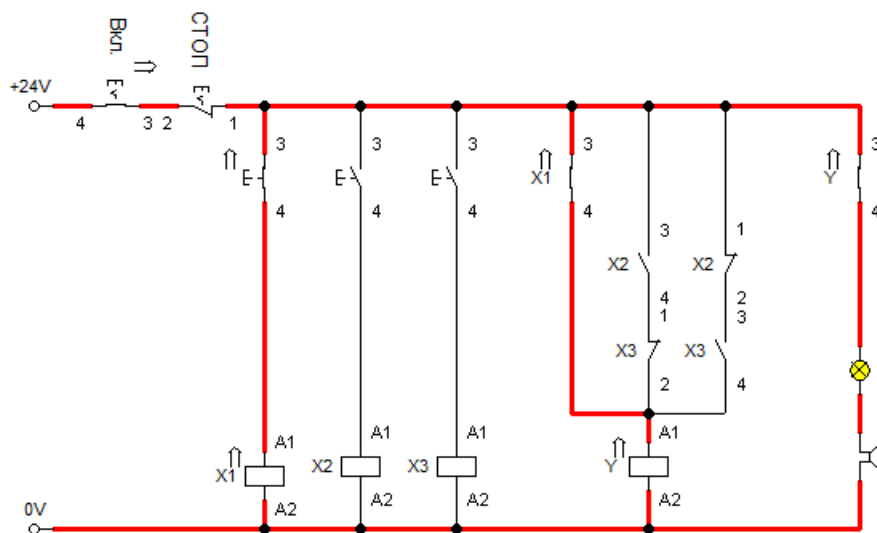


Рисунок 2.7 — Состояние схемы управления для логической функции $f = x_1 + x_2\overline{x_3} + \overline{x_2}x_3$ при проверке условия 1–0–0

2.2 Содержание работы

В процессе работы необходимо решить поставленную задачу в соответствии с индивидуальным заданием и оформить отчет с выводами.

Порядок выполнения работы

1. Изучить теоретическую часть.
2. Для МДНФ с использованием среды FluidSIM построить релейно-контактную схему, реализующую ПФ в СДНФ и СКНФ.
3. Выполнить валидацию полученных схем управления в базисе релейно-контактных элементов на соответствие заданной таблице истинности.
4. Выбрать одну из полученных схем и реализовать её (путем сборки) на стенде FESTO с использованием релейно-контактных элементов, входящих в комплектацию стенда.

2.3 Варианты заданий

Вариантами заданий являются минимизированные переключательные функции, полученные согласно индивидуальным вариантам заданий работы 1.

Лабораторная работа 3

ПРОГРАММИРОВАНИЕ ТЕХНОЛОГИЧЕСКОГО ОБОРУДОВАНИЯ С ПРОГРАММНЫМ УПРАВЛЕНИЕМ

Цель работы: освоение процесса разработки управляющих программ для технологического оборудования с программным управлением.

3.1 Элементы теории

G-код — условное именование языка программирования устройств с числовым программным управлением (ЧПУ). Был создан компанией Electronic Industries Alliance в начале 1960-х. Окончательная доработка была одобрена в феврале 1980 года как стандарт RS274D. Комитет ISO утвердил G-код как стандарт ISO 6983–1:2009, Госкомитет по стандартам СССР — как ГОСТ 20999–83 [1]. В советской технической литературе G-код обозначается как код ИСО 7-бит (ISO 7-bit), это вызвано тем, что G-код кодировали на 8-дорожечную перфоленту в коде ISO 7-bit (разработан для представления информации УЧПУ в виде машинного кода так же, как и коды AEG и PC8C), восьмая дорожка использовалась для контроля чётности. [2]

Производители систем УЧПУ (CNC), как правило, используют ПО управления станком, для которого написана (оператором) программа обработки в качестве осмысленных команд управления, используется G-код в качестве базового подмножества языка программирования, расширяя его по своему усмотрению [2].

G-Code — это также стандартный язык, используемый многими моделями 3D-принтеров для управления процессом печати. Файлы GCODE могут быть открыты с помощью различных программ 3D-печати, например, Simplify3D, GCode Viewer, а также с помощью текстового редактора, поскольку их содержимое представляет собой обычный текст. [2]

Основные требования к структуре [2]

Программа, написанная с использованием G-кода, имеет жёсткую структуру. Все команды управления объединяются в кадры — группы, состоящие из одной или более команд. Кадр завершается символом перевода строки (CR/LF) и может необязательно иметь явно указанный номер, начинающийся с буквы N, за исключением первого кадра программы и комментариев. Этот номер является, по сути, меткой кадра и необязательно должен нарастать в программе или представлять собой последовательные целые числа, важно, чтобы номер не повторялся в пределах программы, например, допустимо:

```
...  
N200 G0  
n100 x0  
x5y4  
...
```

В большинстве современных интерпретаторов кода допустимо использовать в коде программы строчные и прописные буквы, как в примере.

Пробелы в строке кадра игнорируются, поэтому допустимо слитное написание команд кадра.

Первый (а в некоторых случаях ещё и последний) кадр содержит только один необязательный знак `<%>`. Завершается программа командами M02 или M30.

Комментарии к программе размещаются в круглых скобках. Комментарий может располагаться как в отдельной строке, так и в любом месте кадра среди команд. Недопустимо оформлять в качестве комментария несколько строк, охваченных парой круглых скобок.

Элементарные команды в каждом кадре выполняются одновременно, поэтому порядок команд в кадре строго не оговаривается, но традиционно предполагается, что первыми указываются подготовительные команды (например, выбор плоскости круговой интерполяции, скоростей перемещений по осям и др.), затем задание координат перемещения, затем выбора режимов обработки и технологические команды.

Максимальное число элементарных команд и заданий координат в одном кадре зависит от конкретного интерпретатора языка управления станками, но для большинства популярных интерпретаторов (стоек управления) не превышает 6.

Координаты задаются указанием оси с последующим числовым значением координаты. Целая и дробная части числа координаты разделяются десятичной точкой. Допустимо опускание незначащих нулей, либо их добавление. Также в подавляющем количестве интерпретаторов допустимо не добавлять десятичную точку к целым числам. Например, Y0.5 и Y.5, Y77, Y77. и Y077.0.

Существуют так называемые модальные и немодальные команды. Модальные команды изменяют некоторый параметр/настройку, и эта настройка действует на все далее исполняемые кадры программы до их смены очередной модальной командой либо её отмены. К модальным командам, например, относятся скорости перемещения инструмента, управления скоростью шпинделя, подачи смазочно-охлаждающей жидкости (СОЖ) и др. Немодальные команды действуют только внутри их содержащего кадра. К немодальным командам относятся, например, команды разгона и торможения шпинделя.

Интерпретатор кода (стойка управления) станком запоминает значение введенных параметров и настроек до их смены очередной модальной командой или отмены ранее введенной модальной команды, поэтому необязательно указание в каждом кадре, например, скорости перемещения инструмента.

Описание и вызов подпрограмм [2]

Язык допускает многократное исполнение однократно записанной последовательности команд и перемещений инструмента, вызываемой из разных частей программы, например, вырезания в листовой заготовке многих отверстий с одинаковым сложным контуром, расположенных в разных местах будущей детали. При этом в теле подпрограммы описывается траектория движения инструмента для вырезания одного отверстия, а в программе производится многократный вызов подпрограммы из разных мест. В теле подпрограммы перемещения инструмента задаются в относительных координатах — координатах, описывающих траекторию инструмента при обработке отверстия, переход к относительной системе координат (иногда такую систему координат называют <инкрементной>) производится командой G91 в начале тела подпрограммы, а возврат к абсолютной системе координат командой G90 — в конце тела подпрограммы. В инкрементной системе команды перемещения инструмента интерпретируются как приращения, например,

G90 X5

(назначение абсолютной системы координат, после исполнения этого кадра машинная координата по оси X станет равной 5)

G91 X10

(назначение инкрементной системы координат, после исполнения этого кадра машинная координата по оси X станет равной 15)

X-15

(после исполнения этого кадра машинная координата по оси X станет равной 0, так как продолжает действовать инкрементная система координат, заданная модальной командой G91).

Тело подпрограммы обязательно должно быть описано до команды конца программы — M30, но допустимо расположение подпрограммы после команды M02 — конца программы и иметь имя, начинающееся с буквы O с цифрами номера подпрограммы, например, O112. В конце тела подпрограммы помещается команда возврата в основную программу — M99.

В программе вызов подпрограммы производится командой M98 с указанием обязательного параметра имени подпрограммы P. Недопустимо совпадение имён подпрограмм в пределах одной программы. Пример вызова подпрограммы O112: M98 P112. Допустимо при вызове подпрограммы указание числа вызовов подпрограммы добавлением необязательного параметра L, например, двукратный вызов подпрограммы 112: M98 P112 L2, что, например, может быть полезно при описании выполнения второго прохода чистовой обработки после первого прохода черновой обработки. При опущенном параметре L подпрограмма вызывается однократно.

Управляющее математическое обеспечение некоторых станков или

некоторые интерпретаторы G-кода допускают вызов подпрограмм по номеру строки (кадра) в программе, для этого используется команда M97 с параметром P, указывающим на номер (метку) строки, например, M97 P321 L4 — четырёхкратный вызов подпрограммы, начинающейся с меткой N321. Оформленная таким образом подпрограмма, как обычно должна заканчиваться командой M99 — возвратом в вызвавшую программу.

Допустимо вложение подпрограмм, то есть из подпрограммы возможен вызов другой подпрограммы. Максимально допустимое число уровней вложения зависит от реализации конкретного интерпретатора G-кода.

Пример программы вырезания 2 прямоугольных отверстий 10 x 20 мм, увеличенных на диаметр торцевой фрезы, с координатами левых нижних углов отверстий x=57, y=62 и x=104, y=76 в листовой заготовке толщиной 5 мм с вызовом подпрограммы описывающей вырезание одного отверстия

```
...
(Фрагмент программы)
G00 X57 Y62 (позиционирование по X, Y на 1-е отверстие)
M98 P112 (вырезание 1-го отверстия)
G00 X104 Y76 (позиционирование по X, Y на 2-е отверстие)
M98 P112 (вырезание 2-го отверстия)
...
M02 (Конец программы)
...
(Тело подпрограммы)
O112 (Метка подпрограммы, номер 112)
G00 Z1 (Подвод инструмента на высоту 1 мм над поверхностью
заготовки со скоростью холостого перемещения)
G01 F40 Z-5.5 (Врезание инструмента на глубину -5,5 мм в заготовку
со скоростью 40 мм/мин)
G91 (Переход в относительную систему координат, в этой системе
вначале X=0, Y=0)
G01 F20 X10 (Вырезание 1-й стороны прямоугольника со скоростью 20
мм/мин)
Y20 (Вырезание 2-й стороны прямоугольника со скоростью 20 мм/мин)
X-10 (Вырезание 3-й стороны прямоугольника со скоростью 20 мм/мин.
Так как включена инкрементальная система координат, то возврат
инструмента в исходную точку до вызова подпрограммы указывается в
виде приращения координаты, здесь -10.)
Y-20 (Вырезание 4-й стороны прямоугольника со скоростью 20 мм/мин)
G90 (Переход в абсолютную систему координат, восстановление
текущих координат до перехода в относительную систему)
G00 Z5 (Подъём инструмента на высоту 5 мм над поверхностью
заготовки со скоростью холостого перемещения)
M99 (Возврат в вызывавшую программу или подпрограмму)
...
M30 (Конец интерпретируемого кода программы. После исполнения этой
команды указатель номера кадра устанавливается на 1-ю строку
программы и исполнение программы останавливается)
```

3.2 Порядок разработки управляющей программы

В качестве объекта программно-управляемого технологического оборудования рассматриваем 3D-принтер Creality Ender 3, в качестве объекта производства — логотип университета.

Для разработки управляющей программы для 3D-принтера можно использовать специализированные программы-слайсеры, например, свободно распространяемый UltiMaker Cura (рис. 4.1–4.2).

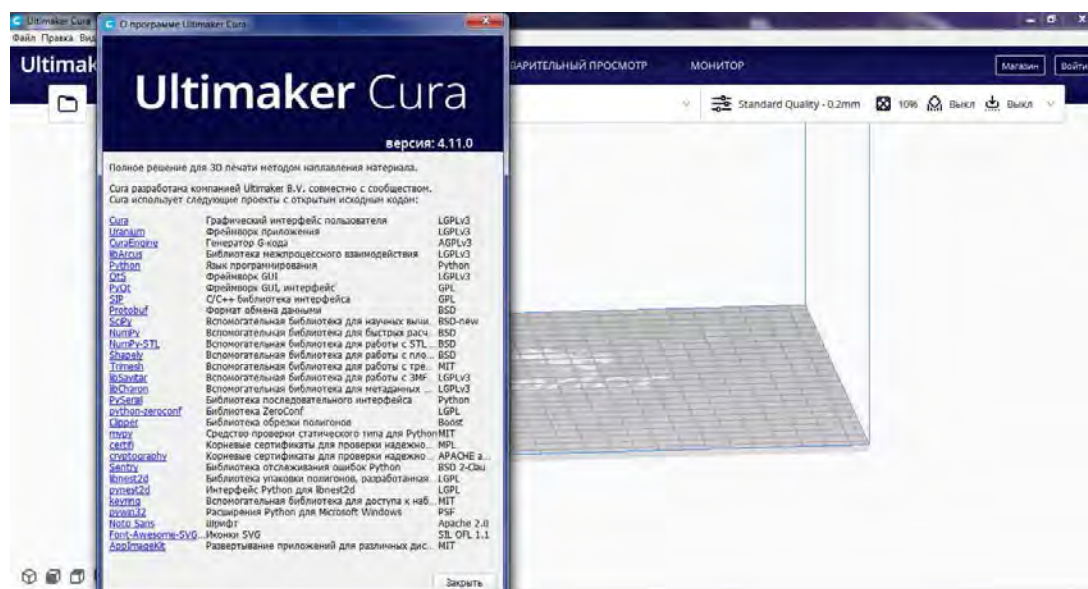
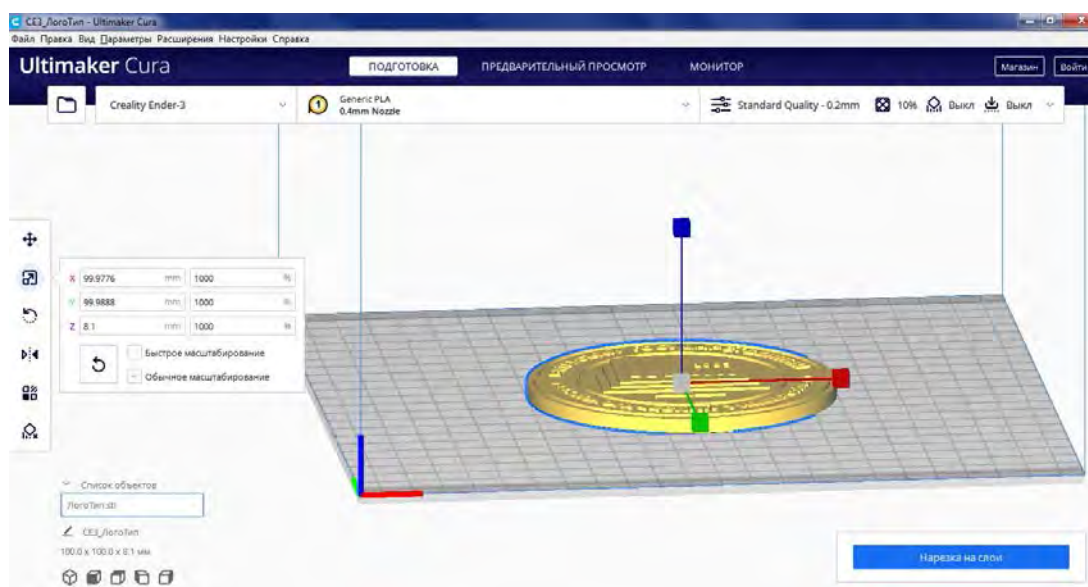


Рисунок 3.1 — Окно слайсера UltiMaker Cura (с окном «О программе»)

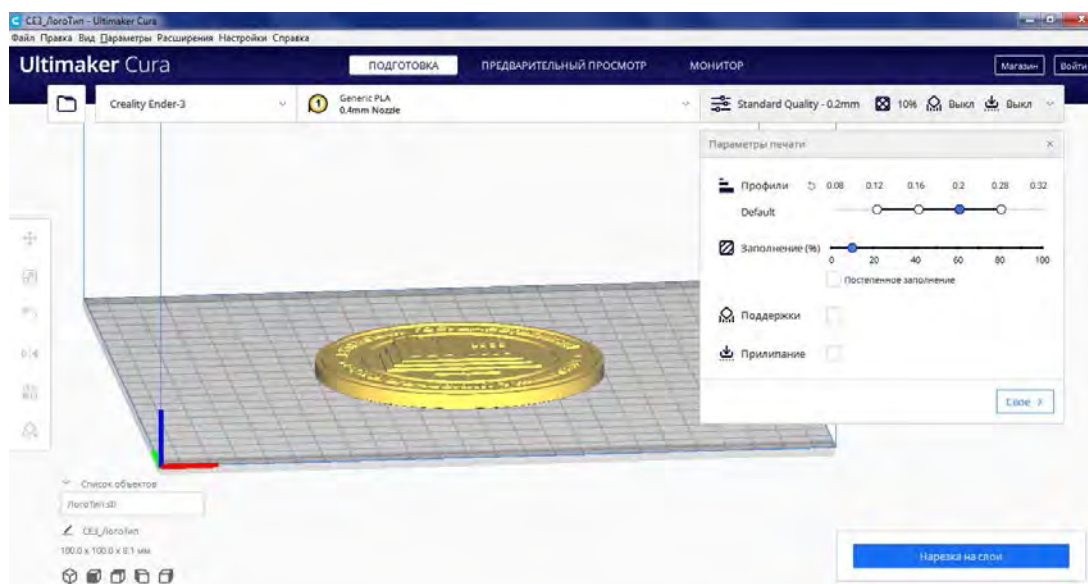
На первом этапе загружается 3D-модель объекта в форма *.stl в рабочее пространство для печати. Для этого необходимо нажать кнопку «Открыть» и в открывшемся окне выбрать соответствующий файл модели.

Модель в пространстве печати загружается в соответствии с той ориентацией и размерами, которые были использованы при экспорте модели в формат *.stl. При необходимости положение, ориентацию и масштаб модели в рабочем пространстве можно изменить.

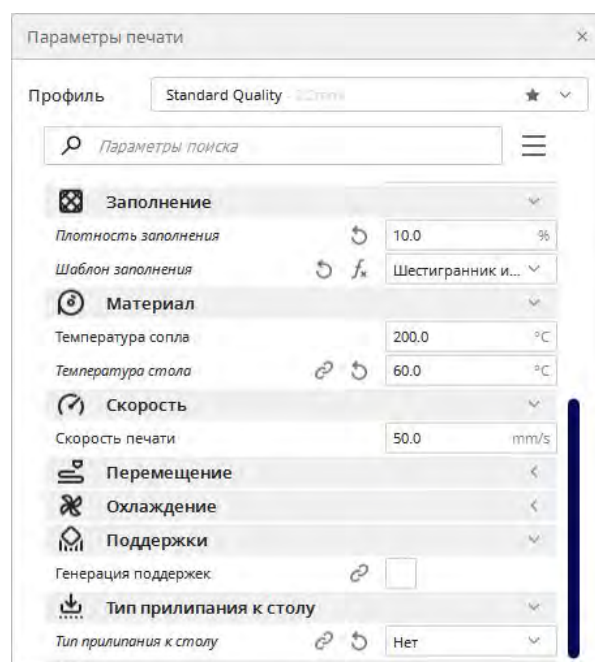
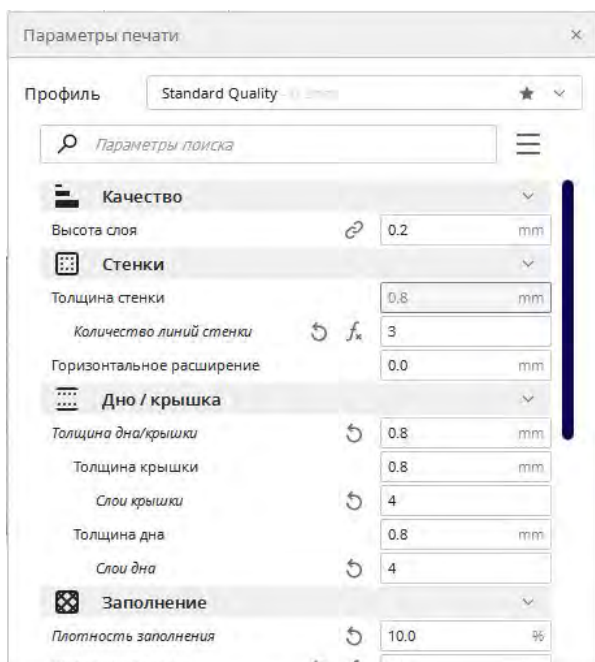


ВНИМАНИЕ! При экспорте модели в формат *.stl значения линейных размеров модели по умолчанию установлены в сантиметрах, при открытии же модели в слайсере значения линейных размеров установлены в миллиметрах.

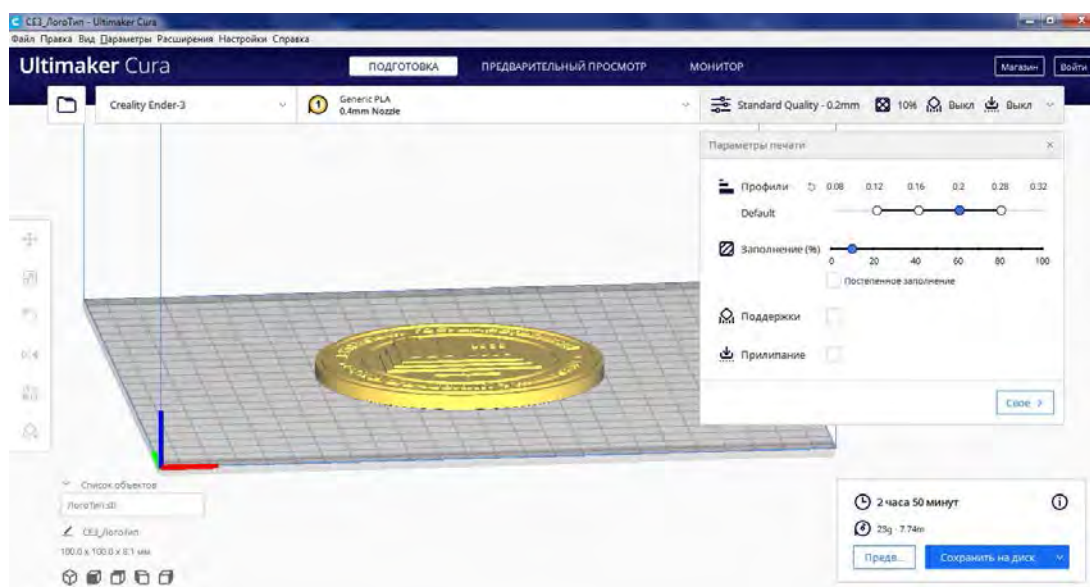
Далее устанавливаются значения необходимых параметров печати. В частности, для примера печати модели логотипа университета устанавливаются следующие параметры: тип пластика — PLA, толщина слоя — 0,2 мм, заполнение — 10 %.



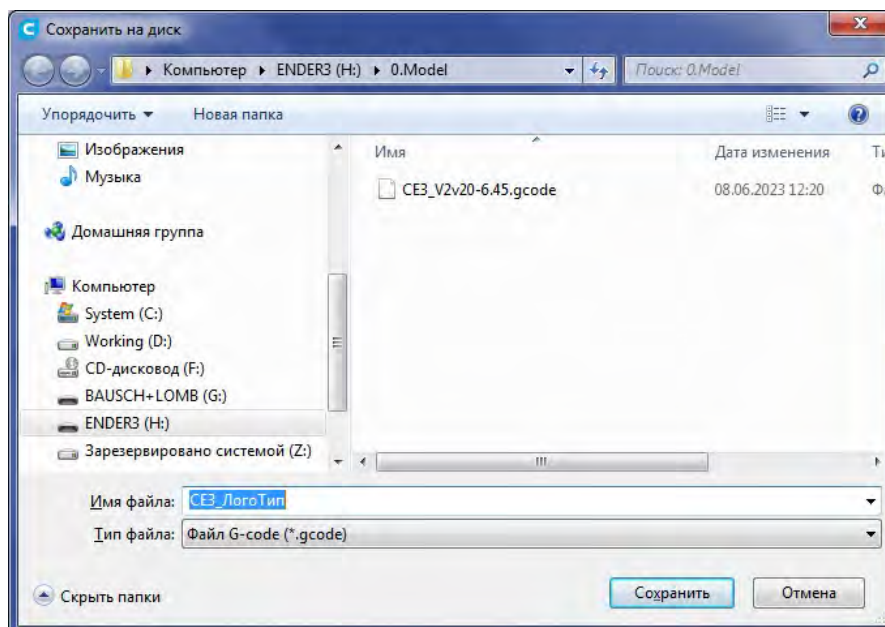
В слайсере UltiMaker Cura параметры печати могут быть установлены в трех режимах — выбор предустановленных профилей качества печати, выбор значений основных параметров печати (в примере выше) или установка значений параметров печати в расширенном режиме.



На следующем этапе осуществляется подготовка управляющей программы, для чего необходимо выбрать команду «Нарезка на слои» и дождаться завершения подготовки программы. По завершению выполнения команды отобразится ориентировочное значение времени печати и количества необходимого филамента (в примере, время печати — 2 часа 50 минут, количество филамента — 23 г, 7,74 м).

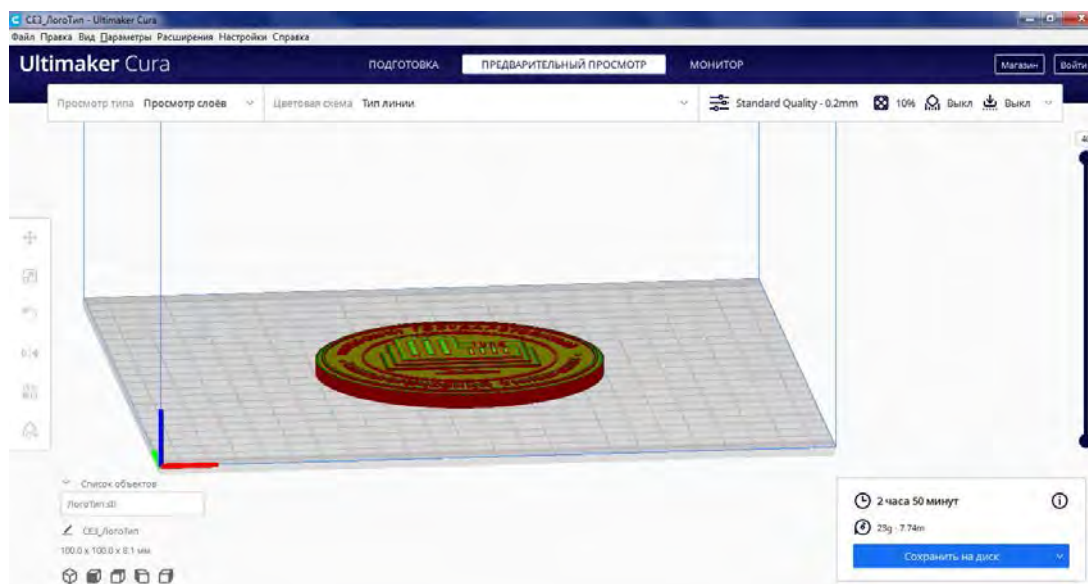


Полученную программу необходимо сохранить на носитель для дальнейшего переноса управляющей программы на лабораторную установку. Для сохранения необходимо выбрать команду «Сохранить на диск», в открывшемся окне выбрать носитель и указать имя файла.

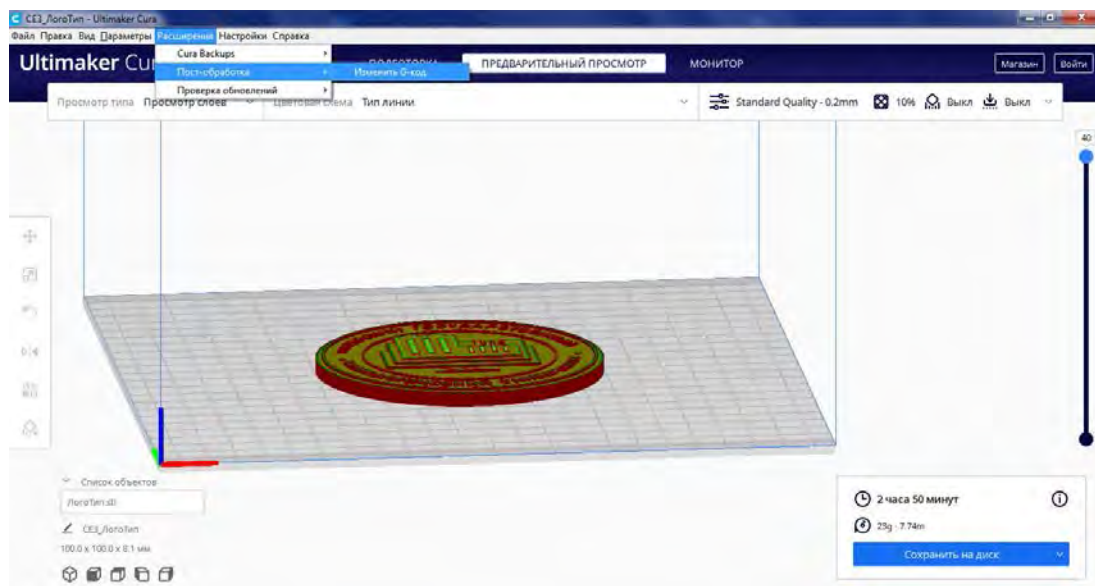


ВНИМАНИЕ! В текущей версии прошивки 3D-принтера Creality Ender 3 не поддерживается отображение кириллических шрифтов на экране принтера.

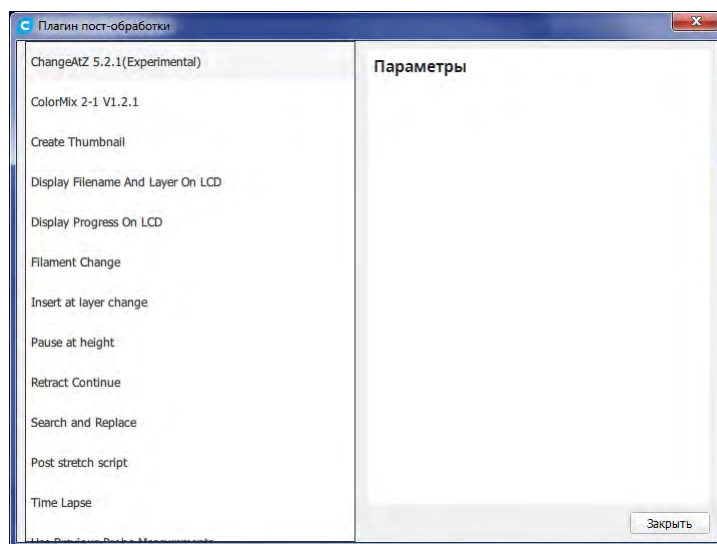
При необходимости в слайсере можно осуществить предварительный просмотр печати с разбивкой модели по слоям, для чего можно переключиться в режим «Предварительный просмотр» и использовать ползунок в правой части окна слайсера.



В слайсере при необходимости можно осуществлять модификацию кода управляющей программы. Для это в основном меню программы необходимо выбрать «Расширения», далее «Пост-обработка», далее «Изменить G-код».

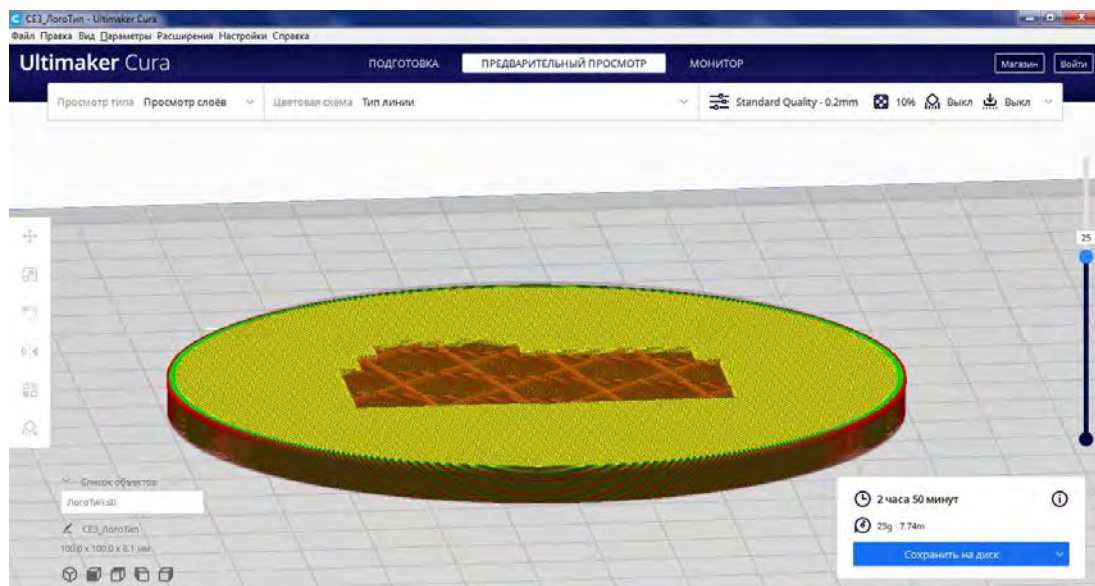


В открывшемся окне отображаются доступные плагины для модификации кода управляющей программы.

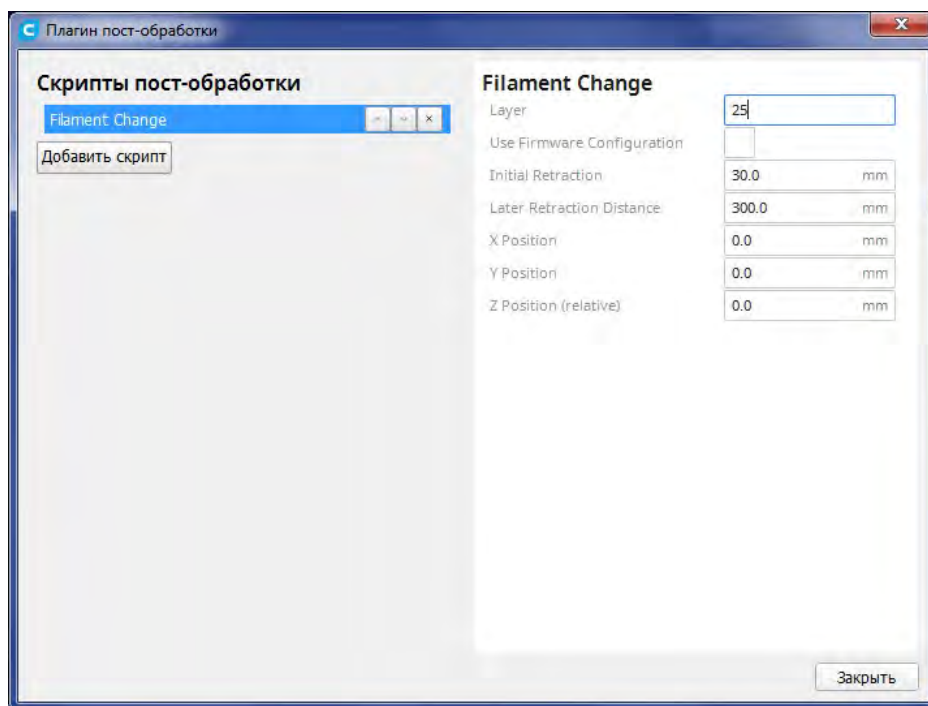


В качестве примера модификации кода управляющей программы для рассматриваемого объекта — логотипа университета используем плагин смены филамента в процессе печати.

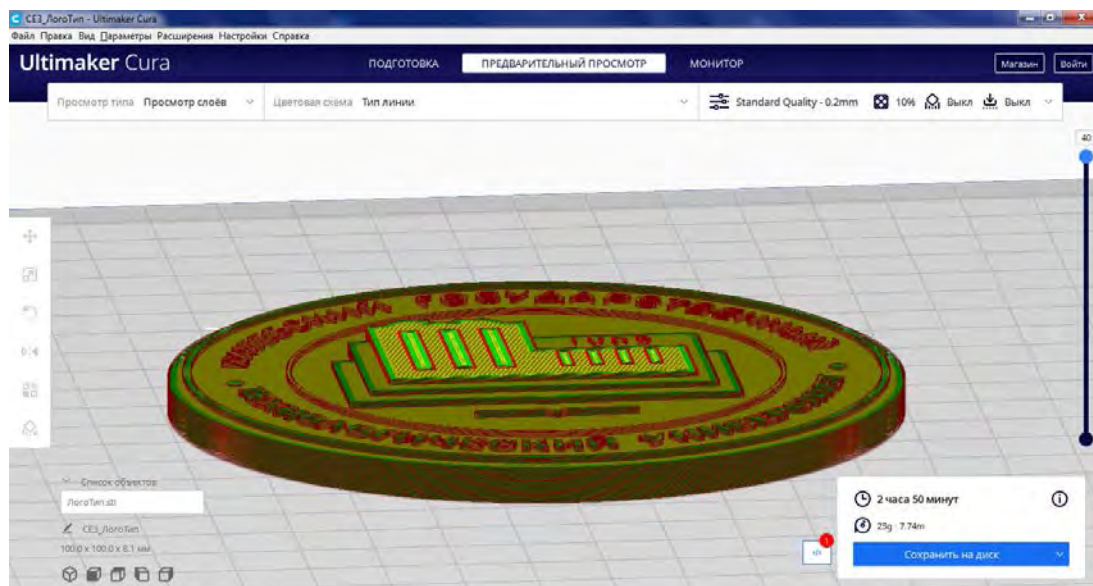
На предварительном этапе необходимо в режиме предварительного просмотра определить слой, после которого будет осуществлена остановка печати и смена филамента. В рассматриваемом примере выберем слой завершающий основание модели (в примере, слой 25).



Далее выбираем скрипт пост-обработки Filament Change и в отрывшемся окне параметров скрипта устанавливаем необходимые значения (в примере, установлено требуемое значение номера слоя (Layer) 25).



После чего необходимо по ранее описанной методике осуществить разбивку модели по слоям и сохранить управляющую программу на диск (носитель).



3.3 Содержание работы

В процессе работы необходимо решить поставленную задачу в соответствии с индивидуальным заданием и оформить отчет с выводами.

Порядок выполнения работы

1. Изучить теоретическую часть.
2. По заданному растровому изображению разработать изображение в векторном формате.
3. На основе полученного векторного изображения разработать 3D-модель изделия, результаты сохранить в нативном формате системы проектирования и экспортировать в формат *.stl.
4. На основе полученной 3D-модели в формате *.stl разработать соответствующую управляющую программу, результаты сохранить в формате g-code.
5. Выполнить анализ полученной управляющей программы в формате g-code.
6. Загрузить полученную управляющую программу в лабораторную установку и осуществить обработку изделия.

СПИСОК РЕКОМЕНДУЕМЫХ ИСТОЧНИКОВ

1. Устройства числового программного управления для металлообрабатывающего оборудования. Кодирование информации управляющих программ [Текст] : ГОСТ 20999–83. — 1983. — Введ. 1984–07–01. — Москва : Государственный комитет СССР по стандартам, 1983. — 28 с.
2. G-code / Wikipedia [Электронный ресурс]. — Режим доступа : <https://ru.wikipedia.org/wiki/G-code>.
3. G-CODE по-русски для 3D печати (Мини-справочник) / SteamGun; 3D-принтеры сегодня! (3dtoday.ru) [Электронный ресурс]. — Режим доступа : <https://3dtoday.ru/blogs/steamgun/g-code-po-russki-dlya-3d-pechati-mini-spravochnik>.
4. Автоматизация технологического оборудования для трехмерных технологий / Виртуальная образовательная среда ВГТУ [Электронный ресурс]. — Режим доступа : <https://sdo.vstu.by/course/view.php?id=1970>.

ПРИЛОЖЕНИЕ А

Краткий справочник по основным командам

Основные (подготовительные) команды (G, General)

- перемещение рабочих органов оборудования с заданной скоростью (линейное и круговое);
- выполнение типовых последовательностей (таких, как обработка отверстий и резьба);
- управление параметрами инструмента, системами координат, и рабочих плоскостей.

Код	Описание	Пример
G00	Ускоренное перемещение инструмента (холостой ход). При холостом перемещении НЕ ОБЯЗАТЕЛЬНО производится линейная интерполяция перемещения аналогично команде G01. В некоторых интерпретаторах при выполнении команды перемещения по нескольким осям одновременно, перемещение по осям обрабатывается с максимальной скоростью, поэтому линейное перемещение от исходной точки в конечную точку не обеспечивается, поэтому нельзя производить ходы обработки детали при действии этой модальной команды.	G0 X0 Y0 Z100.
G01	Линейная интерполяция, модальная команда. Инструмент (рабочий орган) перемещается по отрезку прямой линии от исходной точки с координатами до исполнения команды в точку с заданными в команде координатами, скорость перемещения задаётся здесь же или ранее модальной командой F.	G01 X0. Y0. Z100. F200.
G02	Круговая интерполяция по часовой стрелке, модальная команда. Инструмент перемещается по дуге окружности по часовой стрелке от исходной точки с координатами до исполнения команды в точку с заданными в команде координатами, скорость перемещения задаётся в этой команде параметром F, радиус дуги задаётся параметром R, либо указанием координат центра дуги параметрами I — (смещение центра по оси X относительно начальной координаты X), J — (смещение центра по оси Y относительно начальной координаты Y), K — (смещение центра по оси Z относительно начальной координаты Z) относительно начальных координат инструмента. Для указания плоскости, в которой производится круговая интерполяция, предварительно должна быть указана плоскость круговой интерполяции (в этом же или в другом предварительном кадре) модальной командой G17 (плоскость XY), или G18 (плоскость XZ), или G19 (плоскость YZ). Скорость перемещения задана модальной командой F.	G02 G17 X15. Y15. R5. F200. или G02 G17 X20. Y15. I-50. J-60.
G03	Круговая интерполяция против часовой стрелки. Параметры и действие аналогичны команде G02.	G03 X15. Y15. R5. F200.
G04	Задержка выполнения программы, способ задания величины задержки зависит от реализации системы управления, P обычно задает паузу в миллисекундах, X — в секундах. В некоторых интерпретаторах P задает паузу в секундах и параметр X в этой команде не используется. Также в некоторых интерпретаторах	G04 P500 или G04 X.5

Код	Описание	Пример
G10	возможно задание задержки параметром U. Переключение абсолютной системы координат. В примере начало координат станет в точке 10, 10, 10 старых координат.	G10 X10. Y10. Z10.
G15	Переход в полярную (цилиндрическую) систему координат. В этой системе параметр X задаёт радиус, а Y угол в градусах. Если включена абсолютная система координат (G90), то начало полярных координат будет в точке текущих координат 0; 0, если включена инкрементная система координат, то начало координат будет в точке, достигнутой при отработке предыдущего кадра.	G15 X15. Y22.5
G16	Отмена полярной системы координат.	G16 X15. Y22.5
G17	Выбор рабочей плоскости X-Y.	G17
G18	Выбор рабочей плоскости Z-X.	G18
G19	Выбор рабочей плоскости Y-Z.	G19
G20	Режим работы в дюймовой системе.	G90 G20
G21	Режим работы в метрической системе.	G90 G21
G22	Активировать установленный предел перемещений (Инструмент не выйдет за их предел).	G22 G01 X15. Y25.
G28	Вернуться на референтную точку.	G28 G91 Z0 Y0
G30	Поднятие по оси Z на точку смены инструмента.	G30 G91 Z0
G40	Отмена компенсации радиуса инструмента.	G1 G40 X0. Y0. F200.
G41	Компенсировать радиус инструмента слева от траектории.	G41 X15. Y15. D1 F100.
G42	Компенсировать радиус инструмента справа от траектории.	G42 X15. Y15. D1 F100.
G43	Компенсировать длину инструмента в положительную сторону. В основном применяется при смене инструмента.	G43 X15. Y15. Z100. H1 S1000 M3
G44	Компенсировать длину инструмента в отрицательную сторону. Действие аналогично G43.	G44 X15. Y15. Z4. H1 S1000 M3
G49	Отмена компенсации длины инструмента.	G49 Z100.
G50	Сброс всех масштабирующих коэффициентов в 1,0.	G50
G51	Назначение масштабов. В примере — уменьшение масштаба по оси X в 10 раз. После этой модальной команды все указанные в командах перемещения и координаты по оси X будут умножаться на масштабирующий коэффициент 0,1 и результат интерпретироваться как требуемое перемещение. Если задать масштабирующий коэффициент по некоторой оси (или по любым осям) равным -1, то последующее движение будет зеркальным по этой оси (или осям, где масштабирующий коэффициент -1).	G51 X.1 или G51 X-1
G53	Переход в систему координат станка.	G53 G0 X0. Y0. Z0.
G54–	Переключиться на заданную оператором систему координат.	G54 G0 X0. Y0. Z100.
G59		
G61–	Переключение режимов Точный Стоп/Постоянная скорость.	–
G64		
G68	Поворот координат на нужный угол.	G68 X0 Y0 R45.
G70	Цикл продольного чистового точения.	G70 P10 Q15.
G71	Цикл многопроходного продольного чернового точения.	G71 P10 Q15. D.5 U.2 W.5
G80	Отмена циклов сверления, растачивания, нарезания резьбы метчиком и т. д.	G80
G81	Цикл сверления.	G81 X0 Y0. Z-10. R3. F100.
G82	Цикл сверления с задержкой.	G82 X0. Y0. Z-10. R3. P100

Код	Описание	Пример
G83	Цикл прерывистого сверления (с периодическим полным выводом сверла). Параметр Z указывает полную глубину сверления от поверхности (Z=0), R — высота вывода инструмента над поверхностью для вывода стружки и также конечное положение после завершения сверления, Q — величина заглубления одного из нескольких заглублений при сверлении, F — скорость подачи (необязательна, при отсутствии этого параметра скорость определяется ранее заданной скоростью в команде G1.	F100. G83 Z-20 R1 Q2 F20
G84	Цикл нарезания резьбы.	G95 G84 M29 X0. Y0. Z-10. R3 F1.411
G90	Задание абсолютных координат опорных точек траектории.	G90 G1 X0.5. Y0.5. F10.
G91	Задание координат инкрементально относительно координат последней введенной опорной точки, перемещение инструмента в этой системе координат задаётся в виде приращений.	G91 G1 X4. Y5. F100.
G94	F (подача) — в формате мм/мин.	G94 G80 Z100. F75.
G95	F (подача) — в формате мм/об.	G95 G84 X0. Y0. Z-10. R3 F1.411
G99	После каждого цикла не отходить на <проходную точку>	G99 G91 X10. K4.

Дополнительные (технологические) команды (M, Miscellaneous)

- сменить инструмент;
- включить/выключить шпиндель;
- включить/выключить охлаждение;
- работа с подпрограммами;
- и пр.

Код	Описание	Пример
M00	Приостановить работу станка до нажатия кнопки <старт> на пульте управления, так называемая <безусловная технологическая остановка>	G0 X0 Y0 Z100 M0
M01	Приостановить работу станка до нажатия кнопки <старт>, если включён режим подтверждения остановки. Если этот режим отключён, то команда игнорируется. Используется для начальной проверки (отладки) кода.	G0 X0 Y0 Z100 M1
M02	Конец программы, без сброса модальных функций. Указатель номера кадра не изменяется.	M02
M03	Начать вращение шпинделя по часовой стрелке	M3 S2000
M04	Начать вращение шпинделя против часовой стрелки	M4 S2000
M05	становить вращение шпинделя	M5
M06	Сменить инструмент	T15 M6
M07	Включить дополнительное охлаждение	M3 S2000 M7
M08	Включить основное охлаждение. Иногда использование более одного M-кода в одной строке (как в примере) недопустимо, для этого используются M13 и M14	M3 S2000 M8
M09	Выключить охлаждение	G0 X0 Y0 Z100 M5 M9
M13	включить одновременно охлаждение и вращение	S2000 M13

Код	Описание	Пример
M14	шпинделя по часовой стрелке Включить одновременно охлаждение и вращение	S2000 M14
M17	шпинделя против часовой стрелки Возврат из подпрограммы или из макроса (действие аналогично M99)	M17
M48	Разрешить переопределять скорость подачи	
M49	Запретить переопределение скорости подачи	
M25	Замена инструмента вручную	M25
M97	Запуск подпрограммы, находящейся в той же программе (где P — номер кадра, в примере переход осуществится к строке с меткой N25), <i>реализована не во всех интерпретаторах</i> , предположительно — только на станках HAAS	M97 P25
M98	Запуск подпрограммы, находящейся отдельно от основной программы (где P — номер подпрограммы, в примере переход осуществится к программе O1015)	M98 P1015
M99	Конец подпрограммы и переход в вызвавшую программу	M99
M30	Конец программы, со сбросом модальных функций и изменением указателя номера кадра на начало программы	M30

Параметры команд

Код	Описание	Пример
X	Перемещение инструмента в заданную точку с заданной координатой по оси X при работе в абсолютной системе координат (см. G90) или задание смещений относительно точки, достигнутой в предыдущем кадре при работе в инкрементной системе координат (см. G91)	G0 X100 Y0 Z0
Y	Аналогично предыдущему по оси Y	G0 X0 Y100 Z0
Z	Аналогично предыдущему по оси Z	G0 X0 Y0 Z100
P	При использовании в команде вызова подпрограммы (M98) — указание номера вызываемой подпрограммы с именем, заданным после буквы O, например, P301 вызовет подпрограмму с меткой O301. При использовании в команде задержки (G04) указывает время задержки в миллисекундах.	G04 P500; M98 P301
O	Метка подпрограммы с указанным номером	O301
F	Линейная скорость перемещения инструмента. Для фрезерных станков это дюймы в минуту (IPM) или миллиметры в минуту (мм/мин), Для токарных станков это дюймы за оборот (IPR) или миллиметры за оборот (мм/об). Выбор единиц измерения, дюймы или миллиметры выполняется командами G20 и G21.	G1 G91 X10 F100
S	Частота вращения шпинделя в оборотах в минуту.	S3000 M3
T	Указание номера инструмента в команде смены инструмента. Обычно указывается перед командой M6.	T1 M6
R	Расстояние отвода инструмента в повторяющихся циклах обработки, например, прерывистого сверления глубоких отверстий (G81-G89) или радиус дуги при круговых интерполяциях перемещения инструмента (G02, G03).	G81 Z-20 R2 или G2 G91 X12.5 R12.5
D	Параметр коррекции радиуса выбранного инструмента.	G1 G41 D1 X10. F150.
L	Число вызовов подпрограммы, число вызовов макроса, или количество циклов в повторяющихся операциях X_Y_R_ — параметры, передаваемые в макрос.	M98 L82 P10 или G65 L82 P10 X_Y_R_

Код	Описание	Пример
I	Указание смещения по оси X координаты центра дуги при круговой интерполяции перемещения инструмента (см G02, G03). Координаты центра дуги по осям указываются в виде смещения относительно начальной точки (достигнутой в предыдущем кадре). Плоскость интерполяции (плоскость, которая параллельна заданной координатной плоскости указывается командами G17, G18, G19.	G03 X10 Y10 I10 J0 F10
J	Аналогично предыдущему для оси Y.	G03 X10 Y10 I0 J10 F10
K	Аналогично предыдущему для оси Z.	G03 X10 Y10 I0 K0 F10

Основные (подготовительные) команды (G, General) 3D-принтеров

Код	Описание	Пример
G0/G1	G0 (G1) Xnnn Ynnn Znnn Ennn Fnnn — перемещение. G0 — быстрое холостое перемещение. G1 — линейное рабочее перемещение. Xnnn, Ynnn, Znnn — координаты. Ennn — количество выдавливаемого материала в мм (при отрицательных значениях — ретракт). Fnnn — скорость перемещения в мм/мин (это значение скорости будет использоваться до следующего изменения).	G0 X12 G0 F1500 G1 X90.6 Y13.8 E22.4
G4	G4 Pnnn (или Snnn) — ожидание. Pnnn — время ожидания, в миллисекундах. Snnn — время ожидания в секундах.	G4 S2 G4 P2000
G10	G10 — откат пластика (Ретракт). Откат филамента в соответствии с настройками M207.	—
G11	G11 — подача пластика. Подача / восстанавливает положение пластика в соответствии с настройками M208.	—
G20	G20 — Установка единиц измерения в дюймах.	—
G21	G21 — Установка единиц измерения в миллиметрах.	—
G28	G28 — перемещение в начало («домой») до срабатывания концевых выключателей. G28 — перемещение «домой» по всем осям. G28 X Z — перемещение домой только по осям X и Z.	—
G29	G29 — Создание сетки кривизны стола (MESH_BED_LEVELING). Команда позволяет создать компенсационную (по высоте Z) сетку и использовать ее в дальнейшем при печати. Сетку можно использовать многократно, даже после выключения принтера. После использования команды G28 сетка, созданная командой G29 «слетает». Необходимо сохранять сетку сразу после ее создания! Для вызова сетки из памяти	—

Код	Описание	Пример
	использовать команду M420. Перед использованием G29 обязательно использовать G28, иначе сетка будет неверной. Для использования сохраненной в EEPROM сетки при печати необходимо использовать команду M420 S1 (см. M420).	
G90	G90 — установка абсолютных координат. Все координаты являются абсолютными относительно начала координат станка.	—
G91	G91 — установка относительных координат. Все координаты с этого момента становятся относительными по отношению к последней позиции. Марлин переводит все оси в относительные координаты, в том числе экструдер.	—
G92	G92 Xnnn Ynnn Znnn Ennn — установить позицию. Эта команда может быть использована без каких-либо дополнительных параметров. G92 — сбросит все координаты осей на ноль. Xnnn — новая координата X. Ynnn — новая координата Y. Znnn — новая координата Z. Ennn — новая позиция экструдера.	G92 X10 E90

Дополнительные (технологические) команды (M, Miscellaneous) 3D-принтеров

Код	Описание	Пример
M17	M17 — включить/подать питание на все шаговые двигатели.	—
M18	M18 — убрать ток с двигателей. Аналог команды M84	—
M20	M20 — список файлов на SD карте.	—
M21	M21 — инициализация SD карты.	—
M22	M22 — освобождение SD карты.	—
M23	M23 — выбор файла на SD карте.	M23 filename.gco
M24	M24 — начало/продолжение печати с SD карты.	—
M25	M25 — пауза печати с SD карты.	—
M28	M28 — начать запись на SD карту.	M28 filename.gco
M29	M29 — остановить запись на SD карту. Файл, открытый командой M28, закрывается и все последующие команды исполняются в обычном режиме.	M29 filename.gco
M30	M30 — удалить файл с SD карты.	M30 filename.gco
M32	M32 — выбрать файл и начать печать с SD карты. Используется для печати с SD карты и работает так же как M23 и M24.	M32 filename.gco
M80	M80 — включить блок питания ATX.	—

Код	Описание	Пример
	Переводит блок питания АТХ из спящего режима в рабочий режим.	
M81	M81 — выключить блок питания АТХ.	—
M82	M82 — установить экструдер в абсолютный режим.	—
	Позволяет экструдеру производить экструзию в абсолютных единицах.	
M83	M83 — установить экструдер в относительный режим.	—
	Позволяет экструдеру производить экструзию в относительных единицах.	
M84	M84 Snnn X, Y, Z, E — перевести моторы в режим ожидания. Snnn — время в секундах. Если тайм-аут задан с помощью Snnn, то команда просто устанавливает тайм-аут неактивности шагового двигателя. Если моторы (X, Y, Z или E) не указаны, команда немедленно отключает все. Если указана одна или несколько осей, команда немедленно отключает указанные.	M84 S10
M92	M92 Xnnn Ynnn Znnn Ennn — установить количество шагов по осям на единицу. Xnnn, Ynnn, Znnn — шаги на единицу по осям. Ennn — шаги на единицу для экструдера. Позволяет устанавливать количество шагов на единицу (обычно мм) для двигателей. Данные значения заменяются на значения из прошивки при включении питания, если не записать их в EEPROM см. M500.	M92 X87.489 Y87.489 Z87.489 M92 E420
M104	M104 Snnn — установить температуру экструдера и НЕ ждать. Snnn — заданная температура. Устанавливает температуру активного экструдера (°C) и сразу же возвращает управление (то есть НЕ ЖДЕТ достижения экструдером заданной температуры). Еще см. M109	M104 S190
M105	M105 — получить температуру экструдера. Получает температуру активного экструдера и горячего стола (°C). Температура передается на подключенный компьютер (ответ, переданный на компьютер может выглядеть так: ok T:201 B:117).	—
M106	M106 Snnn — включить вентилятор обдува детали. Snnn — Скорость вращения вентилятора от 0 до 255 (значение 127 — 50 % скорости).	—
M107	M107 — выключить вентилятор.	—
M108	M108 — отменить нагрев. Обрывает ожидание достижения заданной	—

Код	Описание	Пример
	командами M109 и M190 температуры, продолжает печать.	
M109	M109 Snnn — установить температуру экструдера и ждать.	—
	Устанавливает температуру (°C) и ожидает ее достижения. Еще см. M104.	
M110	M110 Nnnn — установить номер текущей строки.	M110 N123
	Nnnn — номер строки.	
M112	M112 — экстренная остановка.	—
M114	M114 — получение текущих позиций.	—
M115	M115 — получить версию прошивки.	—
M119	M119 — получить статус концевиков.	—
M140	M140 — установить температуру стола и НЕ ждать.	M140 S65
M190	M190 — Установить температуру стола и ждать.	—
	Устанавливает температуру (°C) и ОЖИДАЕТ ее достижения. См. M140.	
M200	M200 Dnnn Tnnn — установить РЕАЛЬНЫЙ диаметр прутка филамента.	M200 D1.65
	Dnnn — диаметр в мм.	
	Tnnn — номер экструдера (для одноэкструдерных принтеров можно не указывать).	
	Используется для вычислений реального выдавливаемого объема.	
	Для установки номинальных параметров см. M404.	
M201	M201 Xnnn Ynnn Znnn Ennn — установка максимальных ускорений (мм/с ²).	M201 X1000 Y1000 Z100 E2000
	Xnnn, Ynnn, Znnn — ускорения для осей.	
	Ennn — ускорения для экструдера.	
	Можно использовать только один/два из параметров.	
	Для сохранения параметров в EEPROM использовать M500.	
M202	M202 — Установка максимального ускорения для простого (холостого) перемещения (мм/с ²).	M202 X1000 Y1000
	<i>Не используется в Марлин!</i>	
M203	M203 Xnnn Ynnn Znnn Ennn — установка максимальной скорости (мм/с).	M203 X6000 Y6000 Z300 E8000
	Xnnn, Ynnn, Znnn — макс. скорость для осей.	
	Ennn — макс. скорость для экструдера.	
	Можно использовать только один/два из параметров.	
	Для сохранения параметров в EEPROM использовать M500.	
M204	M204 Pnnn Rnnn Tnnn — установка ускорений (мм/с ²).	M204 P800 T3000 R9000

Код	Описание	Пример
	Pnnn — ускорения при печати. Rnnn — ускорение ретракта. Tnnn — ускорения при холостых перемещениях. Можно использовать только один/два из параметров. Для сохранения параметров в EEPROM использовать M500.	
M205	M205 Xnnn, Znnn, Ennn — установка максимальных рывков(jerk) (мм/сек). Xnnn — рывок по осям X и Y (по этим осям рывки одинаковые). Znnn — рывок по оси Z. Ennn — рывок для экструдера. Можно использовать только один/два из параметров. Для сохранения параметров в EEPROM использовать M500.	M205 X30 Z5
M206	M206 Xnnn, Ynnn, Znnn — установка смещений относительно концевиков (ноля). Подобие команды G92, но эти смещения можно записать в EEPROM см. M500.	M206 X10.0 Y10.0 Z-0.4
M207	M207 Snnn Fnnn Znnn — установка параметров ретракта (втягивание прутка). Snnn — положительное значение ретракта (мм). Fnnn — скорость подачи (мм/с). Znnn — лифт (подъем) головы по оси Z (мм) при ретракте (помогает не задеть модель). Используется впоследствии для команд G10 и G11. Для сохранения параметров в EEPROM использовать M500.	M207 S4.0 F2400 Z0.075
M208	M208 Snnn Fnnn — параметры восстановления подачи прутка после ретракта. Snnn — положительное значение подачи (мм). Fnnn — скорость подачи (мм/с). Для сохранения параметров в EEPROM использовать M500.	—
M209	M209 Snnn — вкл/выкл автоматического ретракта. Snnn — значение 1 — вкл, 0 — выкл. Используется, если слайсер не поддерживает команды G10 и G11. Каждая команда «выдавливания» будет классифицироваться как ретракт, в зависимости от значения (положительное или отрицательное).	—
M218	M218 Tnnn Xnnn Ynnn — установка смещения головы. Tnnn — номер головы.	M218 T0 X50 Y10.5

Код	Описание	Пример
M220	Xnnn, Ynnn — координаты по X, Y. M220 — установка (переопределение) скорости перемещения (%).	M220 S80
M221	Snnn — скорость перемещения (0..100 %). M221 — установка (переопределение) скорости выдавливания (экструзии). Snnn — переопределение скорости выдавливания (0...100 %, по умолчанию 100 %). Dnnn — номер привода экструдера (только RepRapFirmware), по умолчанию 0. Tnnn — номер привода экструдера (только прошивка Prusa), по умолчанию 0, если не задан — 1.	M221 S70 M221 S95 D1 M221 S85 T1
M301	M301 Hnnn Pnnn Innn Dnnn — записать PID параметры хотэнда (!). Hnnn — номер экструдера; H1 — первый экструдер (хотэнд). Pnnn — коэффициент proportional (Kp). Innn — коэффициент integral (Ki). Dnnn — коэффициент derivative (Kd). Для сохранения параметров в EEPROM использовать M500. Для записи PID стола смотри M304.	M301 H1 P1 I2 D3
M302	M302 Snnn — разрешить выдавливание при температуре Snnn и выше. Snnn — заданная температура.	M302 S170 M302 S0
M303	M303 Ennn Snnn Cnnn — запустить процесс PID калибровки для стола/хотэнда. Ennn — E0 — хотэнд, E1 — стол. Snnn — температура калибровки (°C). Cnnn — количество циклов калибровки (больше циклов — точнее параметры).	M303 E1 C8 S110
M304	M304 Pnnn Innn Dnnn — записать PID параметры стола (!). Pnnn — коэффициент proportional (Kp). Innn — коэффициент integral (Ki). Dnnn — коэффициент derivative (Kd). M301 — без параметров выведет текущие параметры. Для сохранения параметров в EEPROM использовать M500. Для записи PID экструдера смотри M301.	M301 H1 P1 I2 D3
M404	M404 Wnnn — установка номинальной толщины филамента 1.75 или 3. Wnnn — номинальная (теоретическая) толщина филамента (мм). M404 без параметров выведет текущее номинальное значение строкой. Это значение используется для определения процентной разницы при автоматической	M404 W1.75

Код	Описание	Пример
	настройке расхода в ответ на измеренную ширину нити и должно соответствовать значению, используемому для ширины нити в настройках слайсера. Установка реальной толщины филамента см. M200.	
M420	M420 Snnn — вкл/выкл использования сетки компенсации кривизны стола (MESH_BED_LEVELING). Snnn — S1 — вкл., S0 — выкл. M420 S1 — использовать при печати сетку компенсации кривизны стола загруженной из EEPROM. См. G29 чтобы получить текущий статус и создать сетку компенсации кривизны стола.	—
M500	M500 — сохранение данных в EEPROM.	—
M501	M501 — чтение данных из EEPROM.	—
M600	M600 — команда для автоматической смены филамента.	—

Учебное издание

АВТОМАТИЗАЦИЯ ТЕХНОЛОГИЧЕСКОГО ОБОРУДОВАНИЯ ДЛЯ ТРЕХМЕРНЫХ ТЕХНОЛОГИЙ

Методические указания
по выполнению лабораторных работ

Составители:
Климентьев Андрей Леонидович
Котович Антон Викторович

Редактор *А.В. Пухальская*
Корректор *А.В. Пухальская*
Компьютерная верстка *А.Л. Климентьев*

Подписано к печати 04.09.2023. Усл. печ. листов 2,7.
Уч.-изд. листов 3,4. Заказ № 225.

Учреждение образования «Витебский государственный технологический университет»
210038, г. Витебск, Московский пр., 72.

Отпечатано на ризографе учреждения образования

«Витебский государственный технологический университет».

Свидетельство о государственной регистрации издателя, изготовителя,
распространителя печатных изданий № 1/172 от 12 февраля 2014 г.

Свидетельство о государственной регистрации издателя, изготовителя,
распространителя печатных изданий № 3/1497 от 30 мая 2017 г.

Учебное издание

АВТОМАТИЗАЦИЯ ТЕХНОЛОГИЧЕСКОГО ОБОРУДОВАНИЯ ДЛЯ ТРЕХМЕРНЫХ ТЕХНОЛОГИЙ

Методические указания
по выполнению лабораторных работ

Составители:
Климентьев Андрей Леонидович
Котович Антон Викторович

Редактор *А.В. Пухальская*
Корректор *А.В. Пухальская*
Компьютерная верстка *А.Л. Климентьев*

Подписано к печати 04.09.2023. Формат $60 \times 90 \frac{1}{16}$. Усл. печ. листов 2,7.
Уч.-изд. листов 3,4. Тираж 2 экз. Заказ № 224.

Учреждение образования «Витебский государственный технологический университет»
210038, г. Витебск, Московский пр., 72.

Отпечатано на ризографе учреждения образования

«Витебский государственный технологический университет».

Свидетельство о государственной регистрации издателя, изготовителя,
распространителя печатных изданий № 1/172 от 12 февраля 2014 г.

Свидетельство о государственной регистрации издателя, изготовителя,
распространителя печатных изданий № 3/1497 от 30 мая 2017 г.