

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ
ФЕДЕРАЦИИ
федеральное государственное бюджетное образовательное учреждение
высшего образования
«Тольяттинский государственный университет»

Институт машиностроения

(наименования института полностью)

Кафедра «Промышленная электроника»

(наименование)

11.03.04 Электроника и нанoeлектроника

(код и наименование направления подготовки, специальности)

Электроника и робототехника

(направленность (профиль) / специализация)

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА

(БАКАЛАВРСКАЯ РАБОТА)

на тему ЛАЗЕРНО-ОПТИЧЕСКИЙ КОМПЛЕКС ПОЗИЦИОНИРОВАНИЯ
ОБЪЕКТА. СИСТЕМА КОМПЕНСАЦИИ ПОГРЕШНОСТЕЙ.

Студент

И.А. Паначев

(И.О. Фамилия)

(личная подпись)

Руководитель

к.т.н., Д.Г. Токарев

(ученая степень, звание, И.О. Фамилия)

Тольятти 2021

АННОТАЦИЯ

Название бакалаврской работы: «Лазерно-оптический комплекс позиционирования объекта. Компенсация погрешностей».

Выпускная работа состоит из введения, трёх глав, заключения, таблиц, списка литературы, включая зарубежные источники, и графической части на 6 листах формата А1.

Целью выпускной квалификационной работы является выявление погрешностей и ошибок на основе лазерно-оптического комплекса в процессе диагностики точности промышленного оборудования для последующей отладки и юстировки.

Выпускная квалификационная работа может быть разделена на следующие три логически взаимосвязанные части (главы): виды погрешностей и их влияние на точность изготовления продукции; выбор комплектующих для лазерно-оптического комплекса; программная часть, практическая модель.

Во введении кратко описывается актуальность проблемы возникновения погрешностей в процессе позиционирования объекта, а также общее представление о процессе изготовления изделий на станках с ЧПУ.

В первой главе рассмотрены виды погрешностей и ошибок, их влияние на производственный процесс изготовления изделия и исследован принцип компенсации погрешностей.

Во второй главе проанализированы и подобраны необходимые комплектующие для создания макета лазерно-оптического комплекса позиционирования объекта.

В третьей главе составлена блок-схема алгоритма, разработана программная часть для управления перемещения и нахождения погрешностей, приведена примерная структурная схема лазерно-оптического комплекса.

Результатом выполненной бакалаврской работы является программа и практическая модель лазерно-оптического комплекса позиционирования объекта, которая способна считывать координату местонахождения датчика, выявлять погрешность и выводить информацию о местоположении шпиндельной головки (лазера) на экран дисплея.

ABSTRACT

The title of the graduation work is Laser-optical complex for object positioning. Compensation of errors.

The senior thesis consists of an introduction, three parts, a conclusion, tables, the list of references including foreign sources and the graphic part on 6 A1 sheets.

The aim of the work is to identify errors on the basis of a laser-optical complex in the process of diagnosing the accuracy of industrial equipment for subsequent debugging and adjustment.

The graduation work may be divided into several logically connected parts which are types of errors and their impact on the accuracy of product manufacturing; selection of components for the laser-optical complex; software part, practical model.

We start with describing the relevance of the errors problem in the process of positioning an object, as well as a general idea of the process of manufacturing products on CNC machines.

We first examine the types of errors, their impact on the production process of manufacturing a product and investigates the principle of error compensation.

We then analyze the necessary components and create a layout of the laser-optical complex for object positioning.

Next, we draw up a block diagram of the algorithm, design a software part for controlling the movement and finding errors, propose an approximate block diagram of the laser-optical complex.

In conclusion we'd like to stress as a result of the graduation project we designed a program and proposed a practical model of a laser-optical complex for object positioning, which is capable of reading the coordinate of the sensor location, detecting an error and displaying information about the location of the spindle head (laser) on the display screen.

Содержание

Введение.....	9
1. Виды погрешностей и их влияние на точность изготовления изделий.....	11
1.1. Механические погрешности. Виды погрешностей при обработке... 11	
1.1.1. Систематические погрешности обработки	13
1.1.2. Случайные погрешности обработки	13
1.2. Программные погрешности	15
1.3. Точность станков и координатно-измерительных машин (КИМ)....	16
1.4. Принцип компенсации погрешностей	18
1.4.1. Принцип вычисления абсолютной погрешности	19
1.4.2. Принцип вычисления относительной погрешности	22
1.4.3. Расчёт суммарной погрешности обработки.....	24
1.5. Определение погрешностей с помощью лазерно-оптического комплекса	25
2. Выбор комплектующих для макета лазерно-оптического комплекса.....	27
2.1. Выбор микроконтроллера	27
2.2. Выбор элементов для двухкоординатного стола.....	31
2.2.1. Выбор двигателя	31
2.2.2. Выбор драйвера.....	33
2.2.3. Выбор концевого датчика	34
2.3. Выбор фотодатчика.....	37
2.3.1. Выбор фотодиода.....	37
2.3.2. Выбор усилителя.....	40
2.4. Полупроводниковый лазер.....	41
2.5. Выбор устройства вывода информации	42
2.6. Выбор платформы для установки	46
2.7. Выбор блока питания.....	47
3. Программная часть. Практическая модель.....	48
3.1. Блок-схема алгоритма.....	48
3.2. Программа для практической модели.....	52

3.2.1 Используемые библиотеки и вводные данные	52
3.2.2 Функция «setup».....	53
3.2.3 Функции «buttonTick» и «loop»	54
3.2.4. Остальные функции.....	56
Список используемой литературы	59
Приложение А Графики абсолютной погрешности при перемещении по осям Ox и Oy при различных значениях оси Oz.....	62
Приложение Б Возможные состояния матрицы при попадании лазерного луча	63
Приложение В Примерная структурная схема макета лазерно-оптического комплекса.....	64
Приложение Г Перечень элементов к схеме принципиальной электрической	65
Приложение Д Блок-схема алгоритма работы макета лазерно-оптического комплекса.....	66
Приложение Е Код программы для макета лазерно-оптического комплекса	69

Введение

Погрешность измерения - неотъемлемая часть любого измерения. С развитием приборов и методов измерения человек стремится уменьшить влияние этого явления на конечный результат измерения.

Определение местоположения объекта в данном случае должно выполняться с высокой точностью, и погрешность измерений не должна превышать размеры измерительных приборов (датчиков).

Оптически-координатные приборы используются для измерения расстояния (координат) перемещаемого объекта. В частности, лазерно-оптический комплекс отличается высокой точностью, бесконтактным измерением и простотой установки. Обычно в нем используется гелий-неоновый лазер со стабилизированной длиной волны и имеет разрешение порядка нанометров для диапазонов измерения в несколько десятков метров. Но промышленные экземпляры довольно большие и дорогие, поэтому они интегрируются только в оборудование стоимостью в сотни миллионов или используются в качестве эталона для контроля качества.

Вычисление погрешности (или ошибки) сильно зависит от конкретной задачи, которую необходимо решить, а также от вида погрешности, которая необходима.

Как правило, с абсолютной точностью невозможно узнать истинное значение измеренного значения, следовательно, невозможно указать отклонение измеренного значения от истинного. Это отклонение обычно называют ошибкой измерения. Оценить величину этого отклонения можно только, например, с помощью статистических методов. На практике вместо истинного значения используется фактическое значение величины, то есть значение физической величины, полученное экспериментально и настолько близко к истинному значению, что его можно использовать в поставленной задаче измерения. Это значение обычно рассчитывается как среднее значение, полученное путём статистической обработки результатов серии измерений. Полученное значение не является точным, просто наиболее

вероятным. Поэтому при записи результатов измерений необходимо указывать их точность.

Первоначальные ошибки фиксируются станками с ЧПУ на этапе макетирования и разработки конструкции, при выборе материалов для деталей машин, а также при производстве и сборке.

В процессе эксплуатации станков с ЧПУ используются дополнительные погрешности из-за упругих деформаций из-за действия сил резания, температурных деформаций деталей и узлов станка, а также из-за износа различных сопряжений станков (направляющих, подшипников и так далее.). Все указанные машинные ошибки могут иметь разное значение, действовать в разных направлениях и, таким образом, по-разному влиять на выходные параметры точности станка.

По своей природе ошибки возникающие в станке могут быть систематическими и случайными.

Систематические ошибки характеризуются предсказуемостью, постоянством размера и характером их изменения в процессе работы станка. Эти ошибки можно измерить заранее и оценить, как они влияют на точность станка.

Сложнее обстоит дело со случайными ошибками, величина и характер проявлений которых постоянно меняются, поэтому для их определения необходимо постоянно определять их величину и характер влияния на выходные параметры системы.

В данной работе будет рассмотрена система, входящая в состав лазерно-оптического комплекса для нахождения и компенсации погрешностей (ошибок) в ходе работы устройства. Основным исследуемым параметром погрешности является выявление ошибки перемещения по оси Z. Происходит измерение смещения на каждом участке приращения координат, что позволяет получить точную картину погрешностей.

Возникающие погрешности описаны ниже.

1. Виды погрешностей и их влияние на точность изготовления изделий

Погрешность измерения – это отклонение результата измерения от заданного истинного значения измеряемой величины. Ошибка измерения – это сумма ошибок, каждая из которых имеет свою причину. Причины погрешности получаемых результатов подразделяются на механические и программные.

1.1. Механические погрешности. Виды погрешностей при обработке.

В процессе промышленной автоматизации используется большое количество измерительных датчиков и преобразователей, устройств ввода аналоговых сигналов и т. Д. На неточность конечного результата обработки может влиять большое количество факторов, таких как:

- сопротивление проводов;
- отношение между входным сопротивлением средства измерений и выходным сопротивлением датчика;
- качество заземления и экранирования, мощность источников помех;
- погрешность метода косвенных, совместных или совокупных измерений;
- наличие внешних влияющих факторов, если они не учтены в дополнительной погрешности средства измерений;
- погрешность обработки результатов измерений программным обеспечением.

Все ошибки, которые не учитываются в процессе сертификационных испытаний и вносятся в паспорт средства измерений, но появляются при определённых условиях использования, являются методическими. В отличие от них инструментальные погрешности нормируются в процессе

изготовления измерительного прибора и вносятся в его эксплуатационную документацию. Таким образом, если в собранную автоматизированную систему измерений входят средства измерений со нормированными погрешностями, то погрешность, вызванная указанными факторами, носит методический характер. Если будет проведена аттестация всей системы измерения, то методологические ошибки могут быть учтены в общесистемных ошибках, а затем станут инструментальными.

Общие рекомендации по расчету или измерению методической погрешности дать сложно. Каждый конкретный случай требует отдельного рассмотрения.

Все погрешности механической обработки подразделяются на 2 основных класса:

- а) систематические погрешности;
- б) случайные погрешности.

Систематические ошибки определяются как ошибки, размер, время возникновения и направление действия которых можно вычислить на основе различных зависимостей в физике, химии и математике.

Все систематические погрешности подразделяются на 2 основные группы:

- а) постоянные систематические погрешности;
- б) переменные систематические погрешности.

Постоянные систематические ошибки практически не меняются на протяжении всего технологического процесса или операции.

Переменные систематические ошибки – это ошибки, которые изменяются естественным образом в ходе процесса или операции.

Случайные ошибки - это ошибки, которые нельзя определить ни по размеру, ни по направлению, ни при их возникновении.

1.1.1. Систематические погрешности обработки

Причины возникновения систематических ошибок обработки, следующие:

- 1) Геометрическая неточность станка (его приводов и устройств, оборудования и технологических инструментов). Указанные геометрические погрешности переносятся непосредственно на готовое изделие;
- 2) Погрешности от деформации в технологической системе «станок – приспособление – инструмент – деталь» (СПИД);
- 3) Погрешности от температурных деформаций;
- 4) Погрешности, связанные с износом инструмента;
- 5) Методические погрешности – погрешности, которые связаны с приближённым проектированием инструмента, оснастки или используемых механизмов и устройств станка.

1.1.2. Случайные погрешности обработки

В процессе обработки партии деталей на стационарных станках при отсутствии влияния систематических постоянных или переменных погрешностей размеры обрабатываемых поверхностей не остаются постоянными.

Фактические размеры отдельных частей обрабатываемой партии отличаются друг от друга и от заданного количества на величину случайной ошибки и колеблются в определённых пределах.

В результате случайных ошибок обработки размеры деталей, обрабатываемых в одинаковых условиях, различаются.

Разброс размеров вызван сочетанием многих причин случайного характера, которые не подходят для точного определения и которые проявляют свои эффекты независимо друг от друга. Эти причины включают:

- 1) колебания твёрдости обрабатываемого материала;
- 2) колебания величины снимаемого припуска;

- 3) колебания положения заготовки в приспособлении, связанные с погрешностями её устойчивости (базирование и закрепление);
- 4) колебания температурного режима обработки;
- 5) затупление инструмента;
- 6) колебания упругих отжатий элементов технологической системы СПИД под влиянием нестабильных сил резания.

При определённых условиях одни и те же ошибки могут быть как систематическими, так и случайными.

Методы математической статистики, основанные на теории вероятностей, успешно используются для выявления и анализа закономерностей распределения размеров деталей по мере их разброса в процессе обработки.

Рассмотрим основные законы дисперсии (распределения) величин. Из-за случайных ошибок при обработке партии деталей на настроенном станке фактический размер каждой детали является случайной величиной непрерывного характера, поскольку может принимать разные значения в пределах определённого интервала размеров.

Набор значений действительных размеров деталей, обрабатываемых в постоянных условиях и упорядоченных по возрастанию с указанием частоты повторения этих размеров (или конкретной), называется распределением размеров детали.

В частности, это отношение количества деталей одного размера в диапазоне к общему количеству деталей в партии. На практике при тестировании случайных величин непрерывного типа измеренное значение фактического размера детали делится на интервалы таким образом, чтобы величина интервала (разница между наибольшим и наименьшим размером в интервале) была немного больше, чем встречный шаг. Это компенсирует ошибку измерения. В этом случае особое значение имеет отношение количества деталей, фактические размеры которых находятся в заданном диапазоне, к общему количеству деталей, подлежащих измерению в партии.

В рассматриваемой системе к механическим погрешностям и ошибкам будут относиться механические части станка:

- жёсткость конструкции;
- моторы, двигатели, приводы;
- стыки деталей;
- крепление подвижных деталей (стол, шпиндель);
- фиксация статичных деталей (фотоэлементов);

стабильность электропитания

1.2. Программные погрешности

Погрешность программного обеспечения (ПО) оценивается как разница между результатами измерений, полученными с помощью этого программного обеспечения и эталонного программного обеспечения. Эталонное программное обеспечение - это программное обеспечение, которое доказало свою высокую точность в ходе нескольких тестов и экспериментов. Концепция эталонного программного обеспечения условна и определяется соглашением между заказчиком аттестации и исполнителем. Уже сертифицированное программное обеспечение можно использовать в качестве справочника.

К основным источниками погрешностей ПО относятся:

- ошибки записи исходного текста программы и ошибки трансляции программы в объектный код;
- ошибки в алгоритме решения измерительной задачи;
- ошибки в таблицах для линеаризации нелинейных характеристик преобразования;
- применение неустойчивых или медленно сходящихся алгоритмов при решении плохо обусловленных измерительных задач;
- ошибки преобразования форматов данных;
- ошибки округления и др.

Надёжность (достоверность) ПО обеспечивается средствами защиты от несанкционированных изменений, которые могут явиться причиной появления не учтённых при аттестации погрешностей.

В рассматриваемой системе к программным погрешностям и ошибкам будут относиться следующие элементы:

- отсутствие (некорректное значение) сигнала на пути приёма – передачи информации;
- точность калибровки применяемых фотоэлементов;
- ширина и правильность окружности лазерного луча.

1.3. Точность станков и координатно-измерительных машин (КИМ)

На точность станков и координатно-измерительных машин воздействует множество факторов. Ниже перечислены основные причины снижения геометрической точности оборудования, которые влияют на качество обработки деталей:

Ошибки кинематики.

Это – устойчивые погрешности, вызванные несовершенством геометрии, неправильной юстировкой и недостатками сборки станка. Они проявляются постепенно из-за осадки фундамента, износа оборудования или устаревания материалов. Также причиной появления таких ошибок могут стать столкновения.

Термомеханические ошибки.

Внешние и внутренние источники тепла могут привести к температурным деформациям элементов станка и повлиять на его кинематику. Степень и скорость проявления таких деформаций зависит от продолжительности температурных воздействий, термической массы и охлаждения станка.

Масса детали.

Иногда масса и положение заготовки сильно влияют на геометрию станка. Повторяющиеся деформации, вызванные тяжелыми заготовками, могут попадать в категорию ошибок кинематики, но в некоторых случаях их следует измерять и моделировать отдельно.

Динамика станка.

На траекторию станка также влияет динамическая жёсткость его замкнутой конструкции. Тогда к деформациям приводят различные воздействия вроде силы резания и сил, вызванных ускорением или торможением. Однако прецизионная обработка и высокоточные измерения часто проходят на малых скоростях подачи с небольшим ускорением, торможением и силой резания.

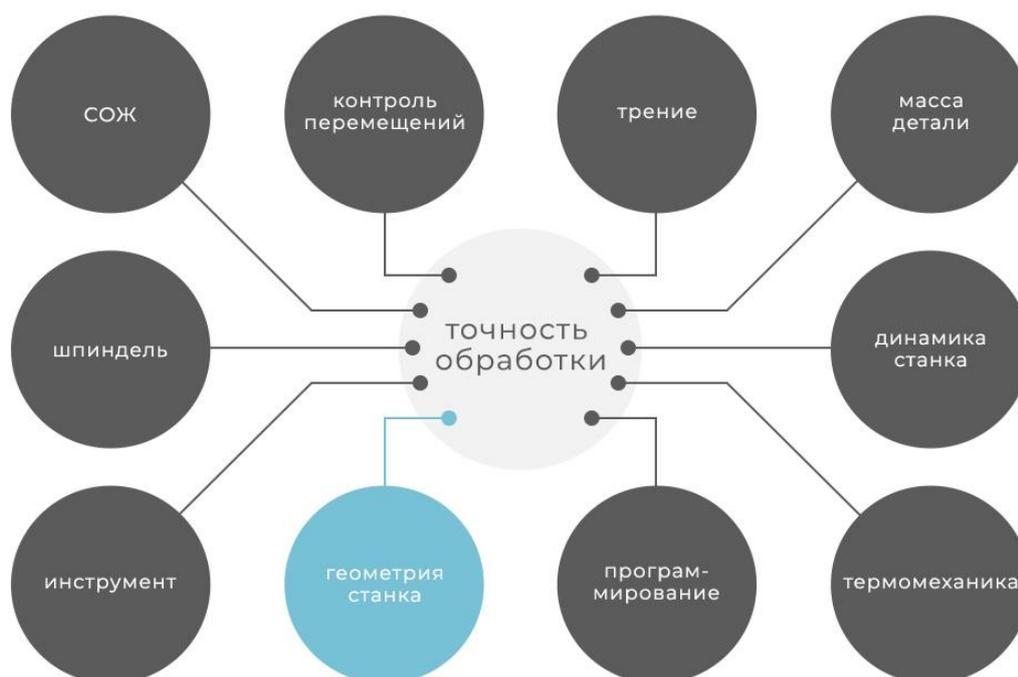


Рисунок 1.1 – Факторы влияния на точность обработки детали.

Рассмотрим погрешности, возникающие у оборудования. Релевантные погрешности станка – это относительные ошибки перемещений между деталью и инструментом. Каждое движение оси станка можно описать шестью степенями свободы: тремя поступательными и тремя

вращательными. За обозначение движений осей отвечает стандарт ISO 841, где X, Y и Z – линейные перемещения, а A, B и C – соответственно вращение вокруг осей X, Y и Z.

Для номинальных линейных перемещений различают шесть основных системных погрешностей каждой оси:

- ошибка положения;
- две ошибки прямолинейности движения;
- ошибка крена;
- ошибки наклонного движения.

При условии, что речь идёт о твёрдом теле, эти погрешности относятся только к номинальным перемещениям и не зависят от положения других осей.

Для номинальных поворотных перемещений также существует шесть системных погрешностей:

- две ошибки радиального движения;
- ошибка движения по осям;
- ошибка углового позиционирования;
- две ошибки наклонного движения.

В ходе выполнения работы координатно-расточного станка большинство погрешностей возникает на оси Z, где расположена шпиндельная бабка. Так как возникает сопротивление при работе с материалами, то может нарушаться определение координат.

1.4. Принцип компенсации погрешностей

Для того чтобы понимать на сколько шпиндель отклоняется от координат необходимо взять некоторую опорную точку. Данной точкой является фотоэлемент, расположенный на столе.

При попадании лазерного луча на фоточувствительный элемент на микроконтроллер подаётся аналоговый электрический сигнал. Чем большую

площадь покрывает лазерный луч, тем выше выдаваемое значение отдельным фотодиодом.

Задача системы заключается в том, чтобы направить лазерный луч точно по центру используемого фотодиода. Как только данная цель достигнута происходит приращение координаты по оси Z. В случае если смещения луча не произошло, и он остался на прежнем месте, то процедура повторяется. В противном случае если лазерный луч сместился по осям X и Y, то происходит повторная процедура нахождения центра фотоэлемента.

Принцип вычисления абсолютной, относительной и суммарной погрешности для практического применения в системе лазерно-оптического комплекса представлен ниже.

1.4.1. Принцип вычисления абсолютной погрешности

Чтобы вычислить абсолютную погрешность, необходимо вычесть меньшее число из большего.

Рассмотрим формулу абсолютной погрешности. Обозначим точное число буквой «A», а буквой « α_i » – приближенное к точному числу. Приближенное число – это число, которое не существенно отличается от точного числа и обычно заменяет его при расчётах. Тогда формула будет выглядеть так:

$$\Delta\alpha = A - \alpha_i \quad (1)$$

где $\Delta\alpha$ – абсолютная погрешность в данной реперной точке; A – заданное программное точное местоположение координаты; а α_i – измеримое местоположение координаты в данный момент времени.

Нахождение абсолютной погрешности по формуле для первой точки представлено ниже.

При высоте Oz равной 30мм от датчика абсолютная погрешность составит:

$$\begin{cases} O_x: \Delta x_{i1} = X_1 - x_{i1} \\ O_y: \Delta y_{i1} = Y_1 - y_{i1} \end{cases} \quad (2)$$

После вычисления абсолютной погрешности измеряемая ошибка учитывается и записывается, для последующей компенсации погрешности в данной реперной точке.

При высоте O_z равной 60мм от датчика абсолютная погрешность составит:

$$\begin{cases} O_x: \Delta x_{i2} = X_1 - x_{i2} \\ O_y: \Delta y_{i2} = Y_1 - y_{i2} \end{cases} \quad (3)$$

После вычисления абсолютной погрешности измеряемая ошибка учитывается и записывается, для последующей компенсации погрешности в данной реперной точке.

При высоте O_z равной 100мм от датчика абсолютная погрешность составит:

$$\begin{cases} O_x: \Delta x_{i3} = X_1 - x_{i3} \\ O_y: \Delta y_{i3} = Y_1 - y_{i3} \end{cases} \quad (4)$$

После вычисления абсолютной погрешности измеряемая ошибка учитывается и записывается, для последующей компенсации погрешности в данной реперной точке.

Далее составляются графики абсолютной погрешности при перемещении по осям O_x и O_y при различных значениях оси O_z (рис.1.2 и рис.1.3 соответственно).

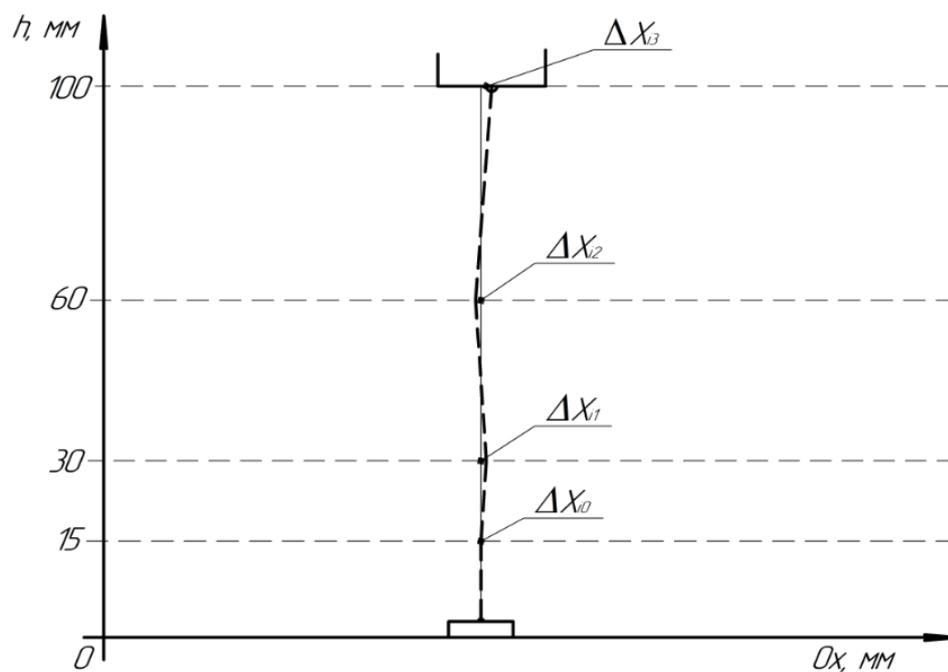


Рисунок 1.2 – Графики абсолютной погрешности при перемещении по оси Ox при различных значениях оси Oz

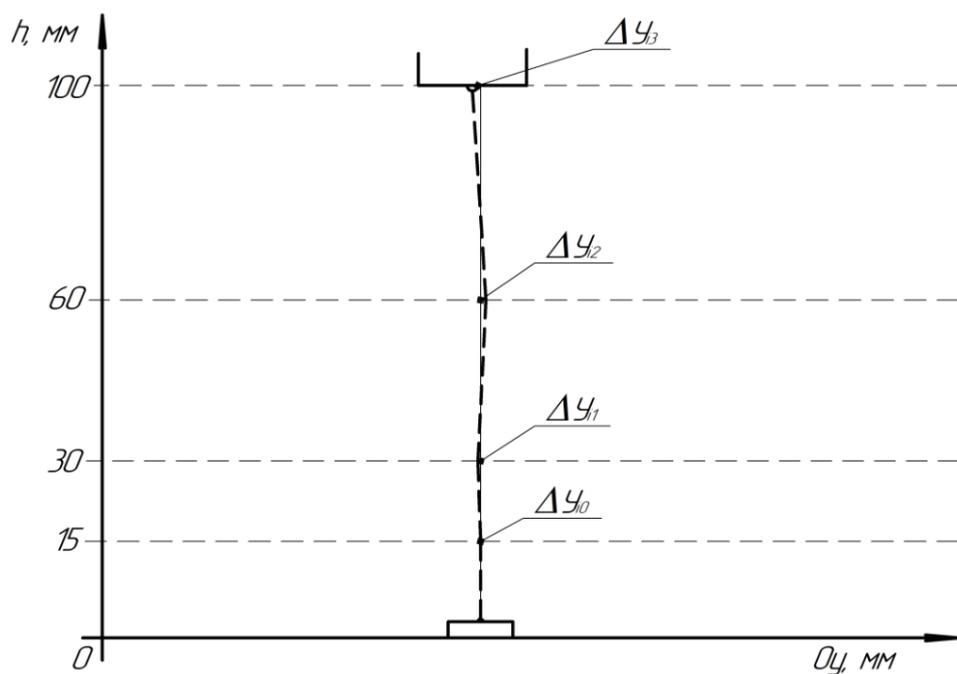


Рисунок 1.3 – Графики абсолютной погрешности при перемещении по оси Oy при различных значениях оси Oz

Абсолютные погрешности Δx и Δy при измерении могут как совпадать по номиналу погрешности распределения фоточувствительности, так и различаться.

1.4.2. Принцип вычисления относительной погрешности

Точность числа определяется его относительной погрешностью. Относительная погрешность – это отношение абсолютной погрешности к указанному измеренному значению. Относительную погрешность принято выражать в процентах (%), т.е. полученное соотношение умножают на 100.

Существует формула (5) относительной погрешности. Обозначим абсолютную погрешность в данной точке Δx_i , а буквой x – заданное значение. Тогда формула (5) для нахождения относительной погрешности по оси Ox будет выглядеть следующим образом:

$$\delta(x) = \frac{\Delta x_i}{x} \times 100\% \quad (5)$$

где $\delta(x)$ – относительная погрешность в данной реперной точке; x – заданное программное точное местоположение координаты; а Δx_i – абсолютная погрешность в данной точке.

Соответственно, чтобы определить погрешность по оси Oy необходимо воспользоваться следующей формулой (6):

$$\delta(y) = \frac{\Delta y_i}{y} \times 100\% \quad (6)$$

где $\delta(y)$ – относительная погрешность в данной реперной точке; y – заданное программное точное местоположение координаты; а Δy_i – абсолютная погрешность в данной точке.

Вспомним, что абсолютную погрешность для первой точки можно найти по формуле (1). Тогда можно найти относительную погрешность для первой точки по оси Ox и Oy . В качестве примера, приведём расчёты погрешности для данных указанных в таблице 1.

Таблица 1 – Пример полученных значений с датчика при различной высоте Oz

Высота, Oz (l,мм)	Номинальное значение, мм		Фактическое значение, мм	
	Ox	Oy	Ox	Oy
15	80	80	80	80
30	80	80	81	79,5
60	80	80	79	81
100	80	80	82	78,5

Тогда при высоте Oz равной 30мм от датчика относительная погрешность по оси Ox и Oy составит:

$$\delta(x) = \frac{(80-81)}{80} \times 100\% = -0,0215 \times 100\% = -1,25\% \quad (7)$$

$$\delta(y) = \frac{(80-79,5)}{80} \times 100\% = 0,00625 \times 100\% = 0,625\% \quad (8)$$

Отрицательный знак полученного значения означает лишь направление смещения шпиндельной головки (лазера) После вычисления относительной погрешности измеряемая ошибка учитывается и записывается, для последующей компенсации погрешности в данной реперной точке.

При высоте Oz равной 60мм от датчика относительная погрешность по оси Ox и Oy составит:

$$\delta(x) = \frac{(80-79)}{80} \times 100\% = 0,0215 \times 100\% = 1,25\% \quad (9)$$

$$\delta(y) = \frac{(80-81)}{80} \times 100\% = -0,0215 \times 100\% = -1,25\% \quad (10)$$

Отрицательный знак полученного значения означает лишь направление смещения шпиндельной головки (лазера) После вычисления относительной погрешности измеряемая ошибка учитывается и записывается, для последующей компенсации погрешности в данной реперной точке.

При высоте Oz равной 100мм от датчика относительная погрешность по оси Ox и Oy составит:

$$\delta(x) = \frac{(80-82)}{80} \times 100\% = -0,025 \times 100\% = -2,5\% \quad (11)$$

$$\delta(y) = \frac{(80-78,5)}{80} \times 100\% = 0,01875 \times 100\% = 1,875\% \quad (12)$$

Отрицательный знак полученного значения означает лишь направление смещения шпиндельной головки (лазера) После вычисления относительной погрешности измеряемая ошибка учитывается и записывается, для последующей компенсации погрешности в данной реперной точке.

Зная данные показания погрешностей становится возможным учитывать их для компенсации ошибок в процессе работы станка.

1.4.3. Расчёт суммарной погрешности обработки

Суммарная погрешность представляет из себя сумму абсолютно всех погрешностей как механических, так и программных.

Из теории вероятностей известно, что дисперсия суммы нескольких независимых случайных величин равна сумме дисперсий этих значений. Учитывая, что дисперсию можно рассчитать по формуле (13), то общая предельная погрешность измерения или изготовления, состоящая из систематических и случайных погрешностей определяется по формуле (14).

$$D(x) = \sigma_x^2 \quad (13)$$

где σ_x^2 – составляющая погрешности.

$$\sigma_\Sigma = \sqrt{\sum_{i=1}^n \sigma_{xi}^2} \quad (14)$$

Формула (14) верна, если законы распределения всех случайных погрешностей близки к нормальным. При определении наибольшей предельной погрешности (наихудший случай) для квадратичной суммы случайных погрешностей тот же знак принимается как сумма систематических погрешностей.

Для определения суммарной погрешности для первой точки по оси Ox и Oy можно составить следующие уравнения:

$$\Delta\sigma_x = k \sqrt{\Delta x_{i1}^2 + \Delta x_{i2}^2 + \Delta x_{i3}^2 \dots \Delta x_{in}^2} \quad (15)$$

$$\Delta\sigma_y = k \sqrt{\Delta y_{i1}^2 + \Delta y_{i2}^2 + \Delta y_{i3}^2 \dots \Delta y_{in}^2} \quad (16)$$

где $\Delta\sigma_x$ и $\Delta\sigma_y$ – суммарная погрешность для первой точки по оси Ox и Oy соответственно, k – коэффициент, определяемый принятой доверительной вероятностью, Δx_i^2 и Δy_i^2 – различные составляющие погрешности.

1.5. Определение погрешностей с помощью лазерно-оптического комплекса

Для выявления погрешностей с помощью лазерно-оптического комплекса используется матрица из 4-х фотодиодов. При попадании луча на область фоточувствительного элемента система определяет в какую сторону необходимо продолжить движение чтобы были затронуты все 4 области (фотодатчика). Матрица из фотодиодов будет иметь вид, представленный на рис.1.4.

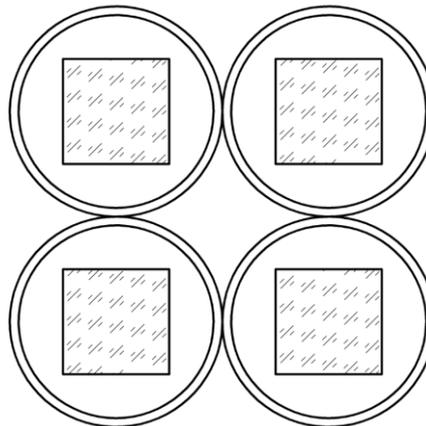


Рисунок 1.4 – Пример матрицы из 4-х фотодиодов

Красным цветом обозначен лазерный луч. А фоточувствительная область фотодиода заливается светло-красным цветом при его попадании. Рассмотрим четыре состояния в которых может находиться данная матрица из фотодиодов.

На рис.1.5а можно увидеть, как лазерный луч перекрывает только один элемент матрицы. В таком случае лазер перемещается вправо (рис.1.5б). Как только лазерный луч перекрывает второй датчик и световой поток на обоих датчиках выравнивается лазерный луч перемещается вниз (рис.1.5в). Если произошло смещение и затронуты только три фотодиода, то происходит регулировка положение стола до полного попадания в центр матрицы из фотодиодов (рис.1.5г).

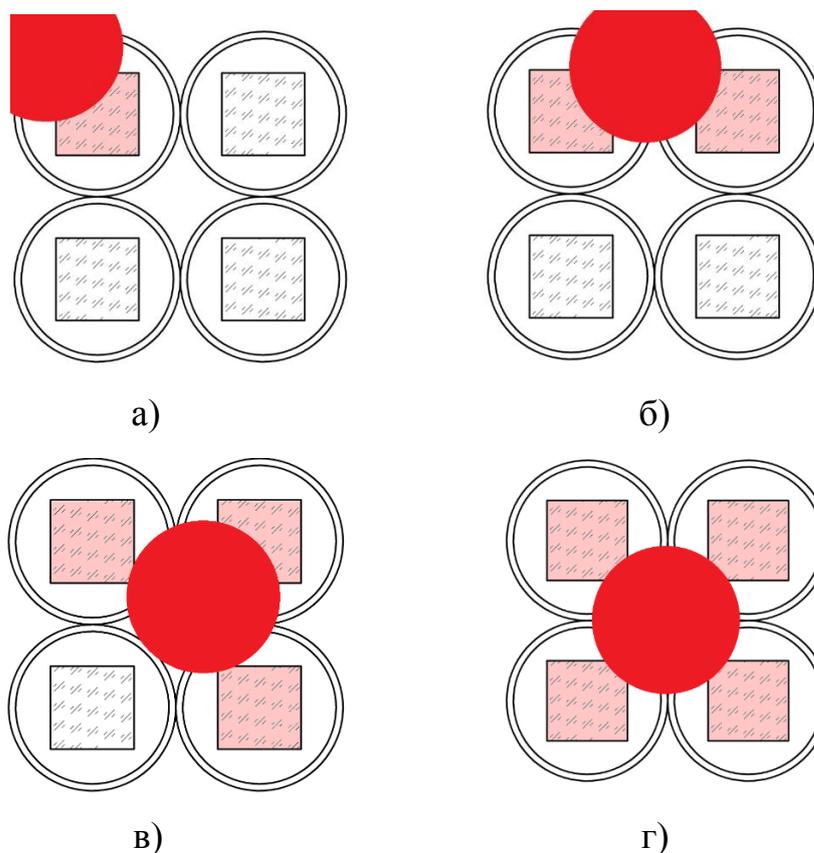


Рисунок 1.5 – Возможные состояния матрицы при попадании лазерного луча

Вывод по разделу

В данном разделе были рассмотрены причины возникновения различного рода погрешности, приведены примеры расчёта абсолютной, относительной и суммарной погрешностей и способы выявления отклонений при работе с применением лазерно-оптического комплекса.

2. Выбор комплектующих для макета лазерно-оптического комплекса

Для того чтобы подобрать необходимые комплектующие для создания макета лазерно-оптического комплекса нужно составить примерную структурную схему макета лазерно-оптического комплекса. Примерная структурная схема макета лазерно-оптического комплекса представлена на рис.2.1.

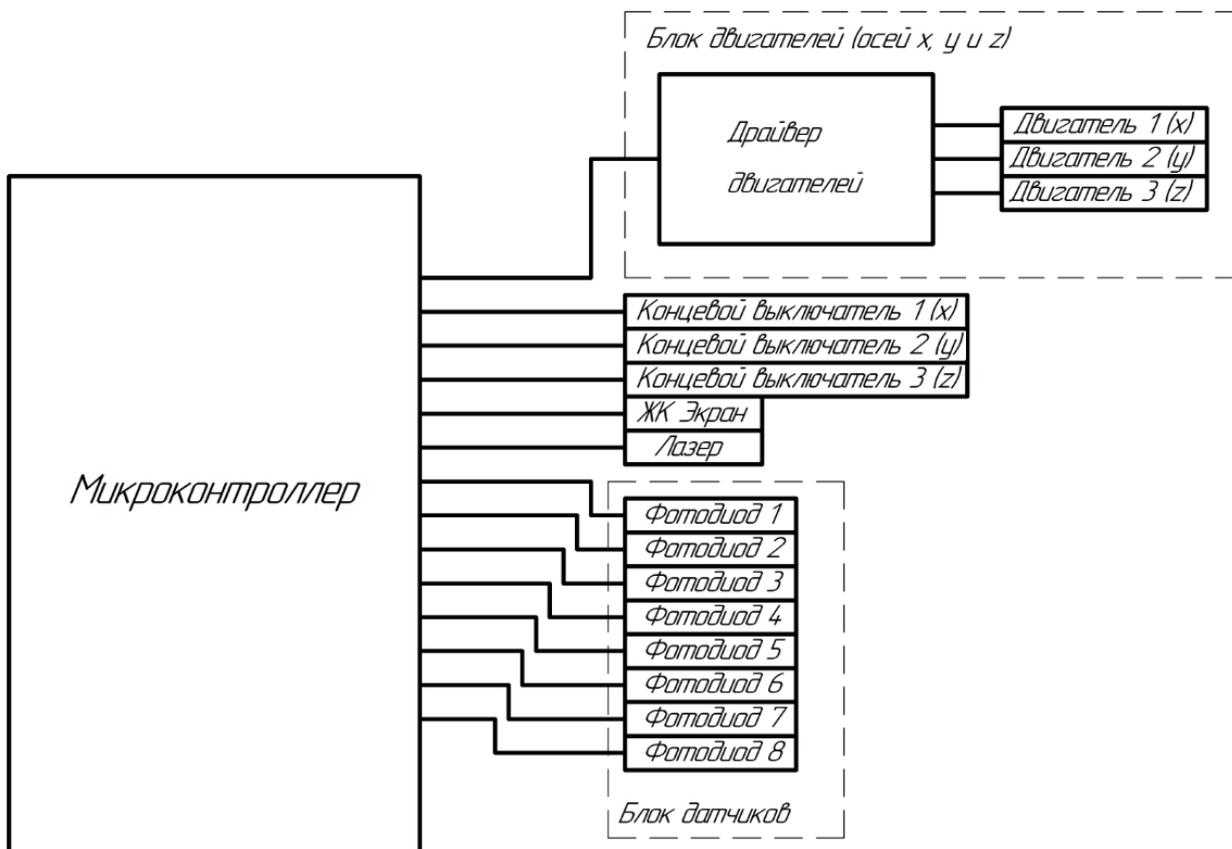


Рисунок 2.1 – Примерная структурная схема макета лазерно-оптического комплекса

Далее рассмотрим более детально компоненты системы лазерно-оптического комплекса.

2.1. Выбор микроконтроллера

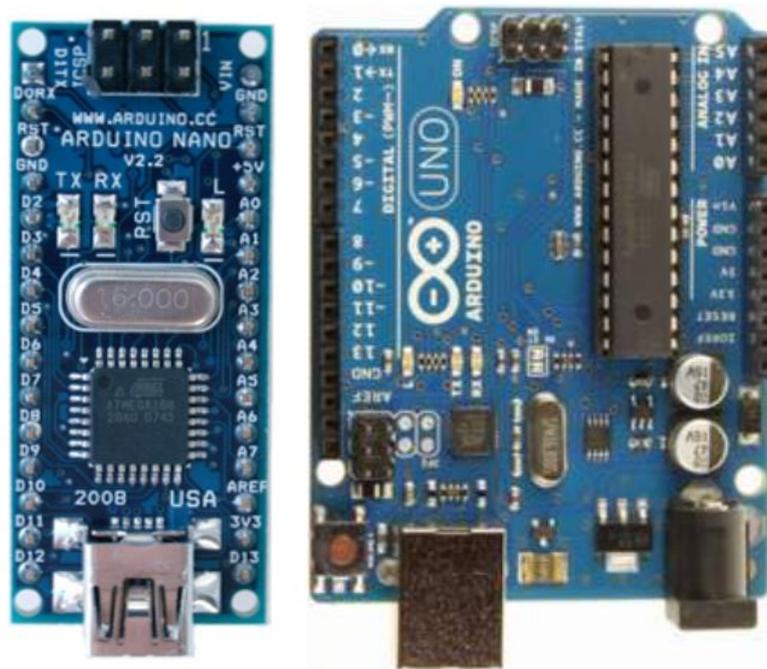
Сегодня на рынке представлен большой выбор программируемых микроконтроллеров, они обладают различными характеристиками:

количеством портов ввода-вывода, объёмом и архитектурой памяти, разрядностью, а также системой команд. В последние годы наиболее распространены следующие виды микроконтроллеров:

- 8-битные микроконтроллеры PIC фирмы Microchip Technology и AVR фирмы Atmel;
- 16-битные MSP430 фирмы TI;
- 32-битные микроконтроллеры, архитектуры ARM.

Поскольку в процессе выполнения необходимых задач при работе лазерно-оптического комплекса будет достаточно 8-ми битного микроконтроллера было принято решение взять готовую платформу на базе микросхем фирмы Atmel.

Примером такой платформы являются отладочные комплексы фирмы Arduino. В них уже интегрированы стабилизатор напряжения, программатор, защитные диоды и т.д. Наиболее распространёнными моделями являются Uno R3 и Nano (рис.2.2), построенные на базе микроконтроллера ATmega328. В связи с этим, характеристики данных плат идентичны и различаются лишь габаритными размерами и разъёмами для подключения к персональному компьютеру. Также на Arduino Uno изготовлено большое количество плат расширения для подключения различных элементов, например, двигателей, датчиков, Wi-Fi и Ethernet модулей. Цена на них варьируется в зависимости от оригинальности платы. Стоимость копии составляет около 150 рублей за версию Nano и около 300 за Uno.



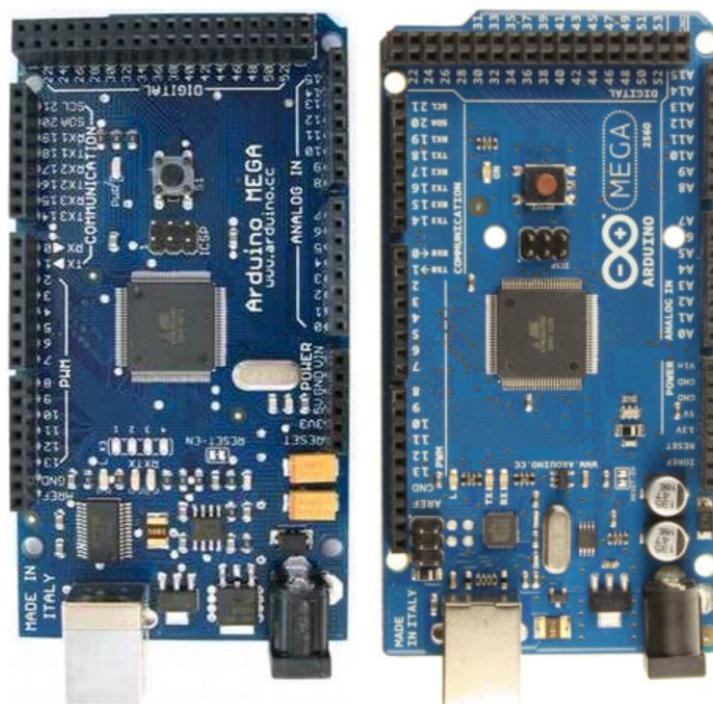
а)

б)

Рисунок 2.2 – Модели отладочных плат *Arduino Nano* (а) и *Uno R3* (б)

Также существуют модели *Mega* и *Mega2560* (рис.2.3). В их основе лежат микроконтроллеры *ATmega1280* и *ATmega2560* соответственно. Это серия более «прокаченных» плат с улучшенными характеристиками. Основным отличием является увеличенное количество портов ввода/вывода информации, а также больший объем памяти. Большинство плат расширения от *Uno* без проблем подходят к *Mega*. Цена варьируется в зависимости от оригинальности платы. Стоимость копии составляет около 700 рублей. В таблице 2 представлены характеристики вышеописанных моделей.

Поскольку для создания лазерно-оптического комплекса требуется наличие большого количества фотодиодов, то лучшим вариантом для реализации проекта будет применение *Arduino Mega*.



а)

б)

Рисунок 2.3– Модели отладочных плат *Mega* (а) и *Mega2560* (б)

Таблица 2 – Характеристики платформ

Наименование	Uno R3	Nano	Mega	Mega2560
Рабочее напряжение, В	5	5	5	5
Входное напряжение (рекомендуемое), В	7-12	7-12	7-12	7-12
Входное напряжение (предельное), В	13-20	13-20	13-20	13-20
Цифровые Входы/Выходы, шт	14	14	54	54
Выходы применяемые для ШИМ, шт	6	6	15	15
Аналоговые входы (подключённые к АЦП), шт	6	6	16	16
Разрядность АЦП, бит	10	10	10	10
Постоянный ток через вход/выход, мА	40	40	40	40
Постоянный ток для вывода 3.3 В, мА	50	50	50	50
Максимальный выходной ток порта 5 В, мА	800	800	800	800
Флеш-память, Кб	32	32	128	256
Оперативное запоминающее устройство, байт	2048	2048	8192	8192
EEPROM, байт	1024	1024	4096	4096
Тактовая частота, МГц	16	16	16	16
Габариты, мм	69×53	18.5×42	101.6×53.3	101.6×53.3

2.2. Выбор элементов для двухкоординатного стола

Для возможности перемещений двухкоординатного стола необходимо подобрать двигатели, а также сопутствующие элементы необходимые для корректной работы и возможности выполнения точных манипуляций.

2.2.1. Выбор двигателя

Среди возможных вариантов использования электромоторов наибольшей точностью позиционирования обладают шаговые двигатели. Они представляют собой бесщеточный электромотор с несколькими обмотками. Движение ротора на заданный угол осуществляется за счет последовательной подачи напряжения на каждую из обмоток. Управление двигателем осуществляется за счет формирования дискретных импульсов на драйвере. На сегодняшний день применяется несколько видов шаговых двигателей, они отличаются конструкцией, типом обмоток, особенностями управления и другими параметрами.

По количеству и способу коммутации обмоток существует два вида шаговых двигателей – биполярный и униполярный

«Биполярный двигатель имеет одну обмотку в каждой фазе, которая для изменения направления магнитного поля должна переполюсовываться драйвером. Для такого типа двигателя требуется мостовой или полумостовой драйвер. Всего биполярный двигатель имеет две обмотки и, соответственно, четыре вывода. Примером распространенного биполярного двигателя может быть шаговый двигатель марки 17HS4401» (рис 2.4) [32].

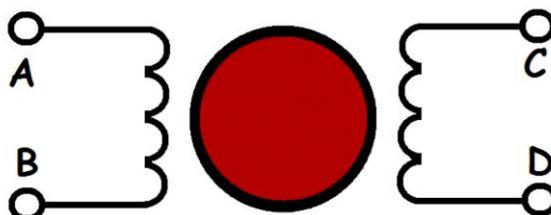


Рисунок 2.4 – устройство биполярного шагового двигателя

Униполярный двигатель также имеет одну обмотку в каждой фазе, но от середины обмотки сделан отвод. Это позволяет изменять направление магнитного поля, создаваемого обмоткой, простым переключением половинок обмотки. При этом существенно упрощается схема драйвера, который в случае униполярного двигателя должен иметь только 4 простых ключа. Средние выводы обмоток могут быть объединены внутри двигателя, поэтому такой двигатель может иметь 5 выводов, как на рисунке 2.5, или 6 выводов в случае если выводы АВ и CD разъединены. Униполярный двигатель с двумя обмотками и отводами можно использовать в биполярном режиме, если отводы оставить неподключенными.

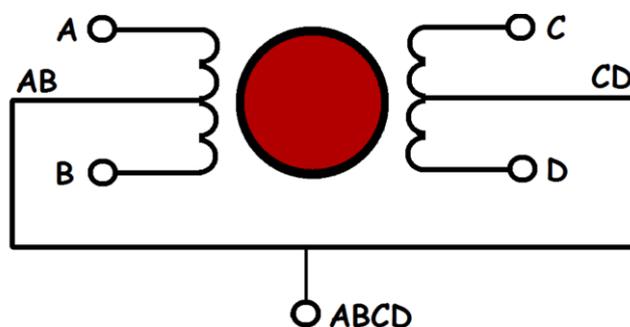


Рисунок 2.5 – устройство униполярного шагового двигателя

Примером распространенного униполярного двигателя с пятью выводами может быть шаговый двигатель марки 28BYJ-48 (рис.2.5). Данный двигатель можно переделать в биполярный, разделив выводы АВ и CD, для чего достаточно перерезать одну из перемычек на плате.

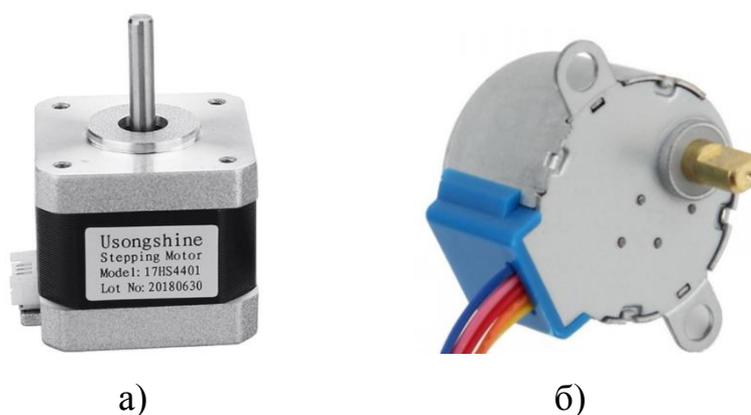


Рисунок 2.6 – примеры шаговых двигателей 17HS4401(а) и 28BYJ-48(б)

При сравнении вышеуказанных шаговых двигателей стоит отметить, что биполярные при одинаковых размерах имеют более высокую мощность и соответственно обеспечивают больший крутящий момент. Это достигается за счет того, момент пропорционален силе магнитного поля, создаваемое статором. Поскольку в биполярном шаговом двигателе в каждый момент времени используются две обмотки, в то время как в униполярном используется лишь половина мощность первого будет примерно на 40% выше. В связи с этим для изготовления лазерно-оптического комплекса гораздо лучше подойдет биполярный шаговый двигатель.

2.2.2. Выбор драйвера

Поскольку, Arduino имеет ограничение по силе тока присоединённой к ней нагрузки, которое равно 800 мА на всю плату и 40 мА для каждого контакта, то подключение двигателей напрямую не представляется возможным.

Чтобы подключить шаговый двигатель необходимо воспользоваться драйвером. Для работы лазерно-оптического комплекса необходимо использование как минимум трех электроприводов. Соответственно число драйверов также не должно быть меньшим.

Сегодня существует достаточно много моделей драйверов для двигателей. Наиболее часто встречающимися являются L293D и L298N (рис.2.7а и рис.2.7б). В плате расширения Motor Shield используются два драйвера L293D, но подключить возможно также только два шаговых двигателя. При использовании драйвера L298N можно подключать только один двигатель к одному модулю, что является недостаточно функциональным и занимает большее пространство.

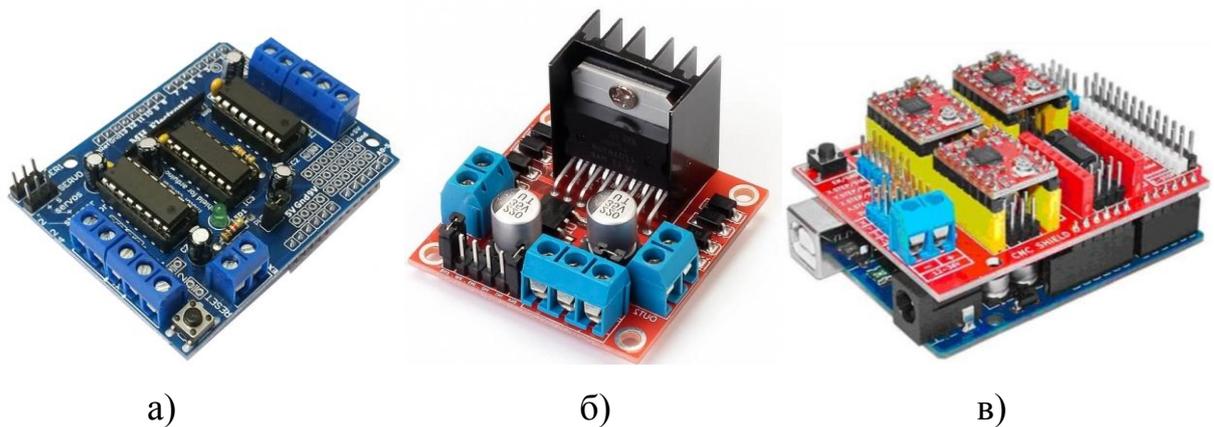


Рисунок 2.7 – Модели плат для подключения двигателей *Motor Shield* (а), *L298N* (б) и *CNC Shield* (в)

Решением данной проблемы может являться применение платы расширения CNC Shield (рис.2.7в) используемой для создания станков с ЧПУ.

Характеристики платы CNC Shield:

- модель - CNC Shield version 3.0;
- размер - Arduino UNO и другие совместимые платы;
- количество осей - 4 (X, Y, Z, A);
- напряжение питания логической части - 5 В;
- напряжение питания силовой части - 12-36 В;
- драйверы - A4988 или DRV8825 и другие;
- интерфейсы - UART, I2C;
- размеры - 65 x 55 x 20 мм;
- вес - 32 г.

К данной плате расширения возможно подключить 4 шаговых двигателя, что является достаточным для создания макета лазерно-оптического комплекса позиционирования.

2.2.3 Выбор концевого датчика

Поскольку в выбранном шаговом двигателе отсутствует устройство для определения количества оборотов и нахождения начальной координаты

необходимо использование концевых датчиков. С их помощью определяется «нулевая» точка. Благодаря ей появляется возможность перемещать платформу зная количество шагов до необходимой координаты.

Существует 3 основных вида концевых датчиков:

- механические;
- бесконтактные;
- магнитные.

В свою очередь данные виды датчиков также подразделяются на более узкоспециализированные подвиды. Рассмотрим наиболее распространённые модели, используемые на небольших станках с ЧПУ и 3D принтерах.

Механический кнопочный концевой выключатель (рис.2.8), он имеет контакты COM, NO, NC. В разомкнутом положении контакты COM и NC замкнуты и находятся под напряжением +5V. Контакт NO подключён к общей шине (GND). При срабатывании датчика контакты COM и NC размыкаются, а COM и NO замыкаются. При срабатывании слышен щелчок. Стоимость данного датчика варьируется от 10 до 100 рублей.

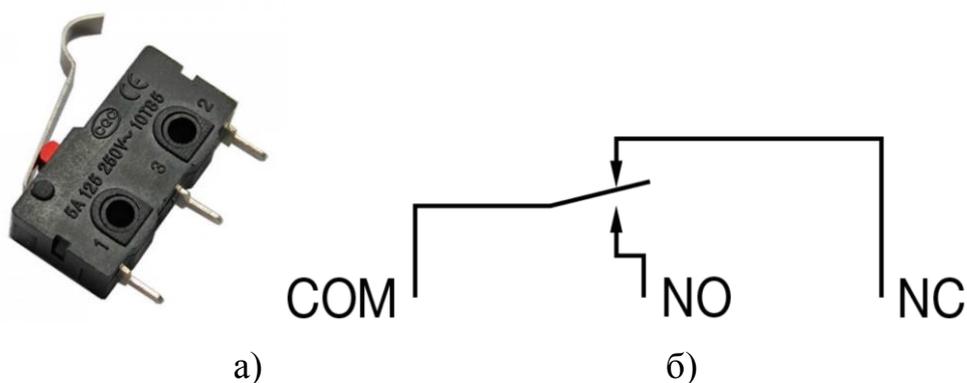


Рисунок 2.8 – Внешний вид механического кнопочного концевой выключателя (а) и его схема подключения (б)

Среди бесконтактных датчиков стоит рассмотреть оптический концевик (рис.2.9). Внутри него установлена оптопара, которая срабатывает при появлении преграды попадающей в зазор. Данные концевые датчики

считаются более точными и долговечными по сравнению с механическими, поскольку нет двигающихся элементов.

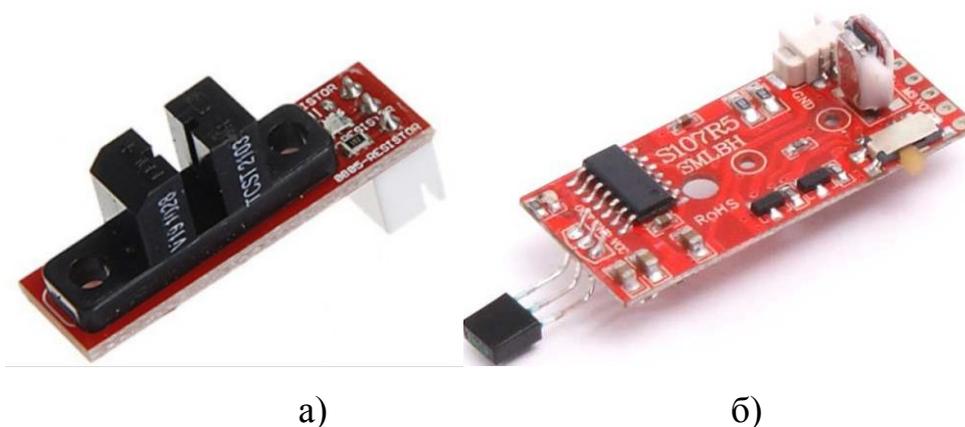


Рисунок 2.9 – Внешний вид оптического (а) и магнитного (б) концевого выключателя

Срабатывание происходит безшумно, для индикации исправности загорается светодиод. Данный датчик может работать некорректно при наличии пыли на фоточувствительном элементе, а также при попадании солнечных лучей. Стоимость варьируется в диапазоне от 70 до 500 рублей.

Магнитный концевой датчик (датчик Холла) – работает при возникновении магнитного поля. Существуют аналоговые и цифровые датчики. В свою очередь цифровые подразделяются на униполярные и биполярные.

Униполярные датчики реагируют на появление магнитного поля определенной полярности, отключение происходит при уменьшении индукции поля. Биполярные же реагируют на изменение полярности. Одна полярность включает биполярный датчик, а другая выключает. Стоимость таких «концевиков» варьируется в диапазоне от 100 до 500 рублей.

Для изготовления макета лазерно-оптического комплекса экономически и конструктивно целесообразно применение механических концевых датчиков.

2.3. Выбор фотодатчика

Чтобы обеспечить точное позиционирования лазерно-оптического комплекса необходимо применение фотодатчиков. Как описывалось ранее, для выявления погрешностей необходимо использование матрицы фотоэлементов. В этом случае использование фоторезисторов и фототранзисторов не является возможным, поскольку среди данных элементов отсутствуют подобные заводские исполнения. Поэтому выбор пал на модели из матрицы фотодиодов в готовом решении.

2.3.1. Выбор фотодиода

Одним из оптимальных фотоэлементов является фотодиод ФДК-142. Предназначен для применения в качестве приёмников и датчиков инфракрасного излучения в составе оптико-электронной аппаратуры, систем фотоэлектрической автоматики и бесконтактного измерения температуры, вычислительной и измерительной техники, программно-управляемого оборудования и приборов, работающих в диапазоне длин волн от 0,4 до 1,1 мкм. Он представляет из себя матрицу из 4-х фотодиодов в одном корпусе. Расстояние между данными фотодиодами составляет 300 мкм. Технические параметры представлены на рис.2.10 и в табл.3.

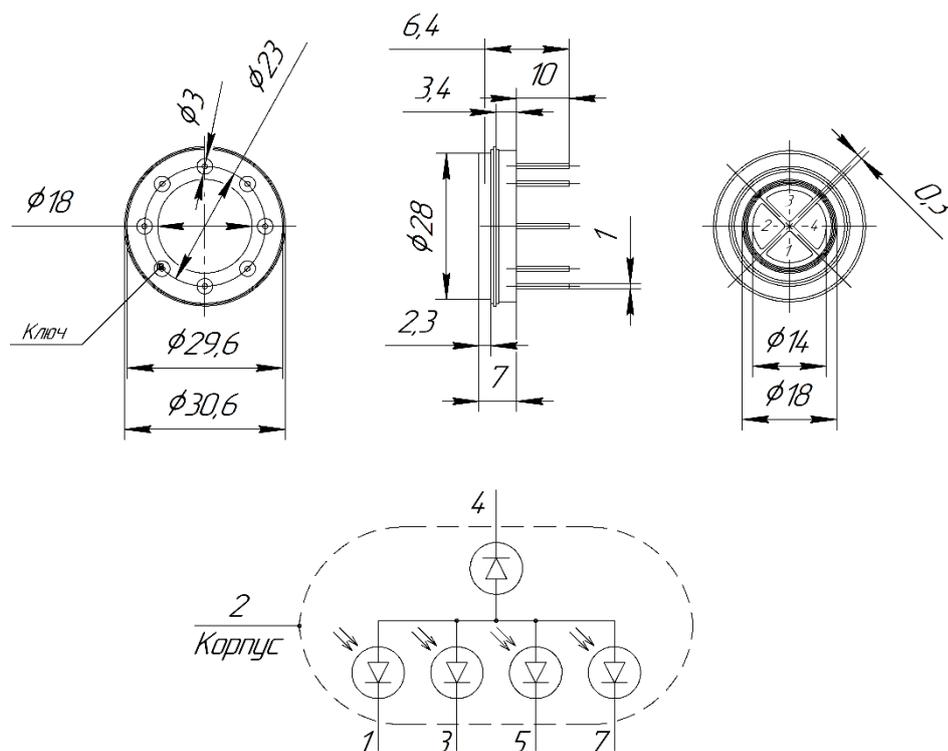


Рисунок 2.10 – Технические параметры ФДК-142

Более современным аналогом ФДК-142 является фотодиод ФДК-157. Предназначен для использования в опико-электронных системах определения координат источников излучения, в том числе в условиях высокой фоновой засветки. Он широко используется в микроэлектронной и электротехнической аппаратуре для обеспечения электрической развязки при передаче информационных сигналов. Технические параметры представлены в табл.3.

Разница между данными двумя фотодиодами заключается в первую очередь в геометрических размерах, а также в их рабочем напряжении $U_{РАБ}$.

Таблица 3 – Основные технические характеристики фотодиодов ФДК-142 и ФДК-157

Наименование фотодиода	Размеры фоточувствительного элемента	Значения основных параметров фотодиодов					
		$\Delta\lambda$	λ_{\max}	$U_{\text{РАБ}}$	I_{T}	$S_{\text{ИНТ}}$	Масса
	мм ²	мкм	мкм	В	мкА	мА/лм	г
ФДК-142	Ø13,7	0,3...1,1	0,9	120	1,15	10	21
ФДК-142-01	Ø13,7	0,4...1,1	0,9	120	10	3	21
ФДК-157	Ø 0,2	0,4...1,1	0,78...0,88	10	0,001	0,31	5

Условные обозначения электрических параметров фотодиодов:

$\Delta\lambda$ – Диапазон спектральной характеристики.

λ_{\max} – Максимальная спектральная характеристика фотодиода.

$U_{\text{РАБ}}$ – Рабочее напряжение фотодиода.

I_{T} – Темновой ток.

$S_{\text{ИНТ}}$ – Интегральная токовая чувствительность.

При попадании лазерного луча на фоточувствительный элемент на микроконтроллер подаётся аналоговый электрический сигнал. Чем большую площадь покрывает лазерный луч, тем выше выдаваемое значение отдельным фотодиодом.

Поскольку найти выше указанный фотодиод довольно проблематично, поэтому в данной работе так же рассмотрено применение других более дешёвых и распространённых фотоэлементов. Но необходимо учитывать, что точность таких фотодатчиков существенно ниже чем у выше упомянутых. А также ширина лазерного луча должна быть больше для того чтобы покрывать все фоточувствительные элементы.

2.3.2. Выбор усилителя

При использовании фотодиодов возникает необходимость применения усилителя, для повышения уровня входного сигнала на аналоговых выводах микроконтроллера.

Для преобразования тока фотодиода в напряжение с оптимальным отношением сигнал/шум обычно используется трансимпедансный усилитель, состоящий из инвертирующего операционного усилителя (ОУ) с резистором обратной связи. При проектировании малошумящих усилителей для фотодиодов с небольшой площадью чувствительного элемента требуются уделять больше внимания входным токовым шумам ОУ и паразитным ёмкостям.

Согласно Филипп С. Д. Хоббс «для измерения напряжения схема должна иметь высокий импеданс, чтобы ток, протекающий через её вход, был минимальным. Это условие обеспечено в схеме, показанной на рис.2.11. Здесь фотодиод включён последовательно с входом операционного усилителя, через который в идеале ток не течёт. Цепь обратной связи, состоящая из резисторов R_1 и R_2 , задаёт усиление напряжения на фотодиоде так же, как если бы оно было приложено к входу усилителя. Очевидно, что измерение напряжения будет нелинейным. Отношение выходного напряжения к входной световой энергии будет логарифмическим, так как чувствительность фотодиода изменяется в зависимости от приложенного к нему прямого напряжения» [17].

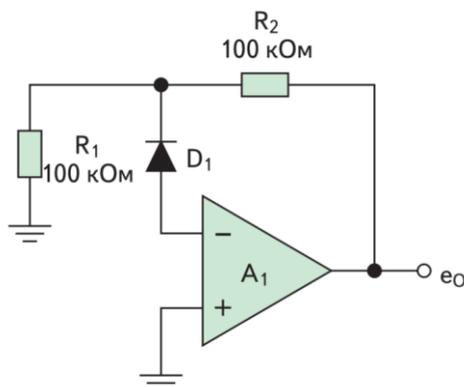


Рисунок 2.11 – Схема подключения усилителя

Для вышеуказанного усилителя подходит операционный усилитель AD820. Данный ОУ построен на полевых транзисторах (FET), обладает преимуществом перед биполярным – низким током утечки, поэтому в схемах-конвертерах ток-напряжение – это очень важно. Также усилитель имеет Rail-to-Rail выход, то есть размах выходного напряжения может приближаться очень близко к шинам питания.

2.4. Полупроводниковый лазер

Для корректной работы фотодатчика необходимо правильно подобрать лазерный модуль. Фотодиод ФДК-157 воспринимает свет в диапазоне длин волн от 400 до 1100 нм. Данный диапазон входит в видимый человеческому глазу спектр волн (рис.2.12), а также частично уходит в инфракрасное излучение. Исходя из этого можно сделать вывод, что для наглядного применения достаточно использование лазерного модуля с видимым светом.

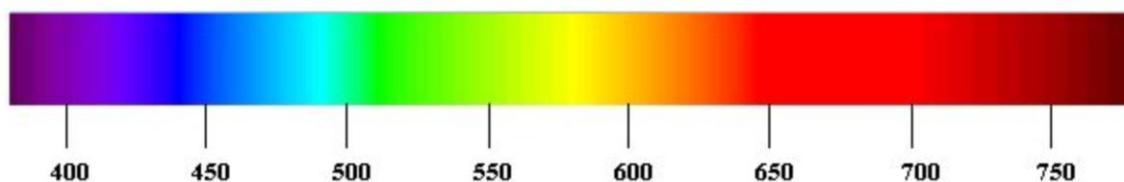


Рисунок 2.12 – диапазон видимого спектра излучения

Лазерная структура состоит из р–n-перехода, смещённого в прямом направлении. Электроны перемещаются из n-области и дырки из p-области соответственно. Из-за диодных характеристик р–n перехода полупроводниковые лазеры часто также называют лазерными диодами.

На сегодняшний день на рынке представлен модуль лазерного передатчика для подключения к Arduino KY-008 (рис.2.13). Он представляет

собой красный лазерный передатчик, основанный на светодиоде с медным радиатором для охлаждения.

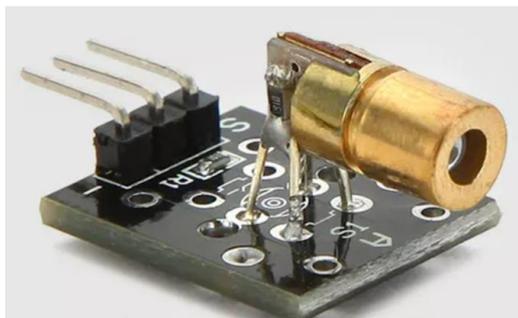


Рисунок 2.13 – Модуль лазерного передатчика KY-008

Спецификация модуля лазерного передатчика KY-008:

- рабочее напряжение - 5 В
- выходная мощность - 5 мВт
- длина волны - 650 нм
- рабочий ток - менее 40 мА
- рабочая температура - от -10°C до 40°C
- размеры - 18,5 мм x 15 мм

Данный модуль соответствует техническим требованиям предъявляемых для использования с фотодатчиками ФДК-157. Его использования будет достаточно для корректной работы лазерно-оптического комплекса.

2.5. Выбор устройства вывода информации

Существуют несколько путей вывода информации. К ним относятся:

- устройства для вывода графической информации (монитор, принтер, проектор);
- устройства для вывода звуковой информации (динамик, колонки, наушники).

Для наилучшего восприятия и максимальной скорости получения данных целесообразно использовать дисплей, подключённый к

микроконтроллеру. А также, в случае необходимости, можно отображать данные на мониторе компьютера.

Рассмотрим различные виды дисплеев, которые возможно подключить к плате Arduino.

Выбор дисплея

Вывод текста может быть осуществлён с помощью нескольких вариантов электронных модулей:

OLED дисплей

Данный дисплей изготовлен из органических светодиодов (англ. organic light-emitting diode, сокр. OLED) – полупроводниковых приборов, производимых из органических соединений. Такие соединения эффективно излучают свет при прохождении тока. Пример экрана изображён на рис.2.14.



Рисунок 2.14 – OLED дисплей

OLED-технология нашла свое применение в создании дисплеев для вывода графической информации.

Достоинства:

- самый дешевый вариант для вывода текста;
- высокая яркость;

- низкое энергопотребление.

Недостатки:

- маленький размер у большинства монохромных моделей и как следствие, большая ограниченность в отображении данных;
- низкий срок службы;
- хрупкость моделей, используемых в связке с микроконтроллером Arduino.

Текстовый LCD дисплей

Дисплей на жидких кристаллах (Liquid Crystal Display) используется для отображения графической или текстовой информации в компьютерных мониторах (также и в ноутбуках), телевизорах, телефонах, цифровых фотоаппаратах, электронных книгах, навигаторах, планшетах, электронных переводчиках, калькуляторах, часах и т. п., а также во многих других электронных устройствах. В данном случае рассматривается вариант дисплея с отдельными ячейками для отображения символов и разделением на строки. Такие дисплеи используются в кассовых аппаратах, станках ЧПУ и везде, где требуется отображение небольшого количества текстовой информации, удобной для прочтения. Пример дисплея изображён на рис.2.15.



Рисунок 2.15 – LCD дисплей

Достоинства:

- отображение символов пригодным для чтения размером шрифта;

- достаточный объем отображаемых данных;
- небольшое число контактов (при использовании платы I2C).

Недостатки:

- относительно большой размер;
- большое количество подключаемых контактов, при использовании без специальной шины.

TFT дисплей

Жидкокристаллический дисплей с активной матрицей (TFT LCD, англ. thin-film transistor – тонкоплёночный транзистор) – разновидность жидкокристаллического дисплея, в котором используется активная матрица, управляемая тонкоплёночными транзисторами. В данном случае рассматриваются модели, применяемые в связке с микроконтроллером Arduino. Пример дисплея показан ниже на рис.2.16.

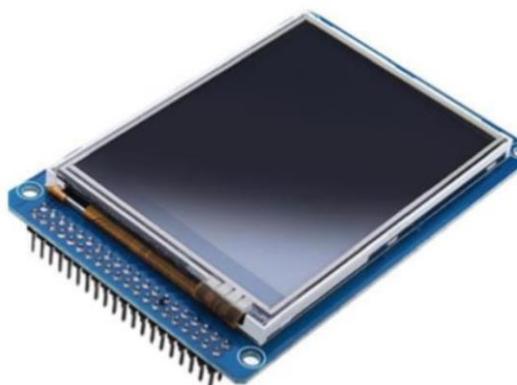


Рисунок 2.16 – TFT дисплей

Достоинства:

- большое пространство для отображения данных;
- возможность выбора цвета;
- возможность выбрать подходящий размер.

Недостатки:

- большое количество выводов;

- высокая стоимость;
- непригодность для вывода большого объема текста крупным шрифтом.

Рассмотрев возможные решения, выбор был сделан в пользу LCD экрана, так как данный вид дисплеев идеально подходит для отображения текстовой и числовой информации.

2.6. Выбор платформы для установки

Сама идея лазерно-оптического комплекса подразумевалась модульного типа, для установки и калибровки различного промышленного оборудования. Данный комплекс можно установить на любой станок промышленного типа.

В данной установке существуют элементы подвижного и статичного типа. Важно понимать какой элемент подвижный, а какой статичный. В качестве примера можно предложить следующие модели:

а) Подвижные элементы: стол перемещается по осям X и Y, а шпиндель (лазер) только по оси Z.

б) Подвижные элементы: стол перемещается только по оси Y, а шпиндель (лазер) – по осям X и Z.

в) Из подвижных элементов только шпиндель (лазер), который перемещается по всем трём осям (X, Y и Z). Статичный элемент – стол.

Соответственно для каждого типа установки необходимо написать соответствующую программу, которая будет удовлетворять параметрам элементов системы.

Для того чтобы собрать и продемонстрировать работу лазерно-оптического комплекса было принято решение собрать показательный макет данной установки. Данный макет должен иметь жёсткий каркас, быть устойчивым, устанавливаться только на ровную поверхность, быть антистатичным или заземлённым, длина проводов должна обеспечивать

нормальное движение подвижных элементов (провода должны не болтаться и не тянуть-вырывать элементы), иметь место для нормального распределения всех элементов (двигателей, концевых выключателей и т.д.), при необходимости иметь охлаждение.

2.7. Выбор блока питания

Для того чтобы обеспечить нормальную работу всех компонентов данного макета лазерно-оптического комплекса необходимо подобрать соответствующее питание (блок питания).

Самыми энергопотребляемыми элементами являются драйвер и двигатели. Для того, чтобы обеспечить напряжение питания логической и силовой части драйвера необходим источник постоянного напряжения от 5 В до 12 В. Суммарная мощность всей установки не превышает 20 Вт. В связи с этим подойдёт любой блок питания с запасом по мощности 30% от максимальной мощности потребления макета лазерно-оптической установки.

Выводы по разделу

В данном разделе были рассмотрены различные комплектующие элементы, которые возможно использовать при создании макета лазерно-оптического комплекса. Выбраны оптимальные элемент с учётом экономической и конструктивной целесообразности.

3. Программная часть. Практическая модель.

Чтобы все выбранные компоненты работали слажено и в правильном порядке необходимо написать программную часть лазерно-оптического комплекса, которую будет выполнять микроконтроллер. Важным моментом при написании программы является проработка логических функций и действий, т.е. правильная последовательности команд, подаваемых на элементы системы.

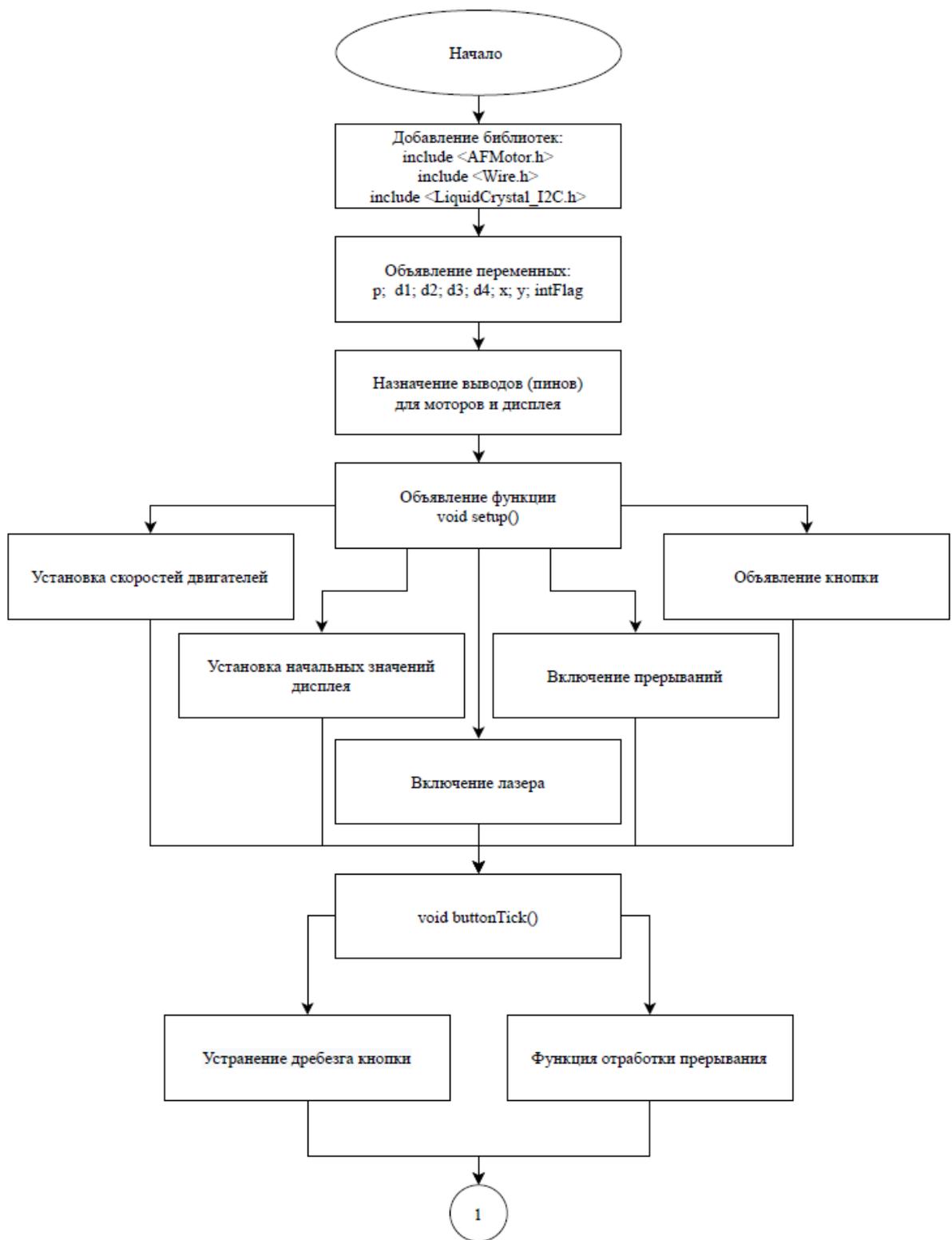
Началом разработки любой программы является составление блок-схемы алгоритма.

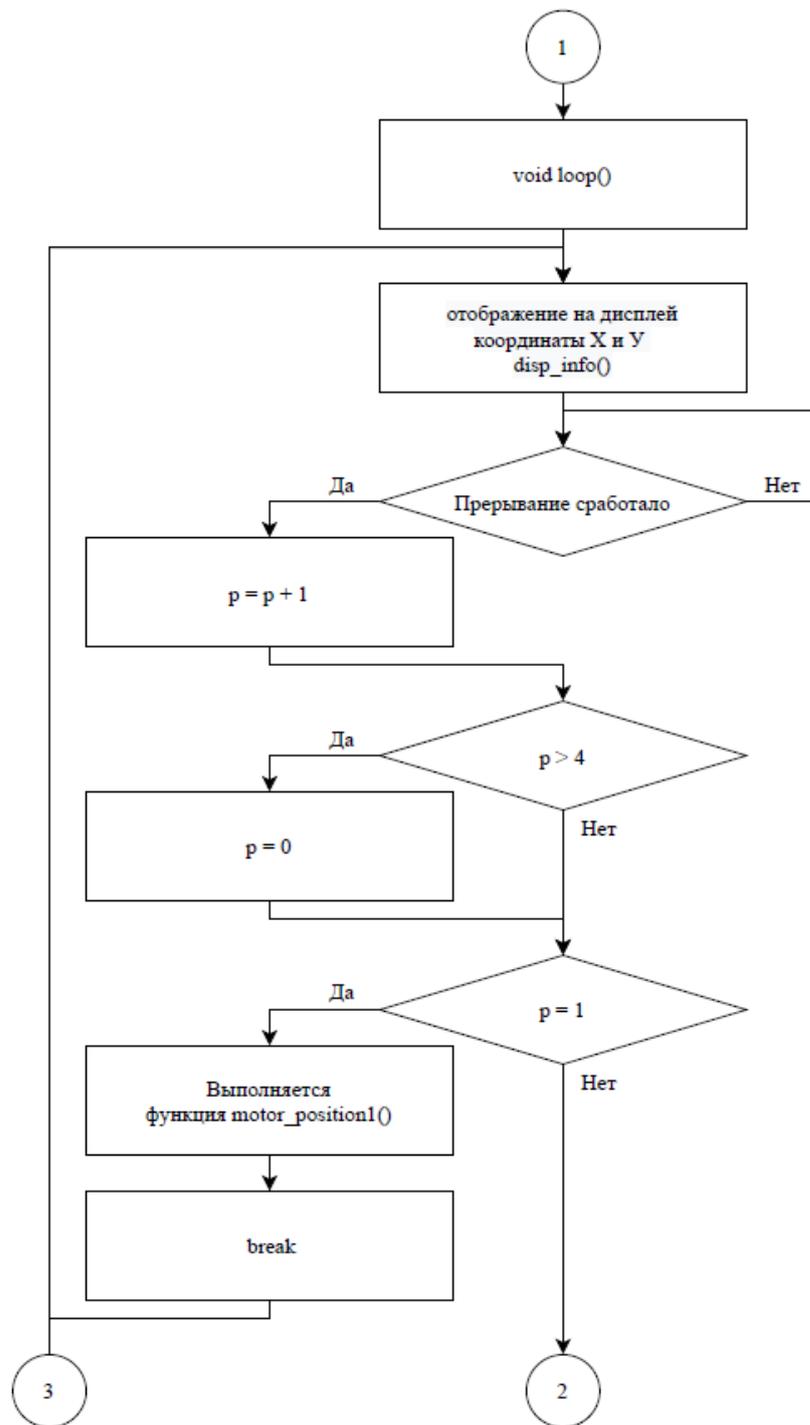
3.1. Блок-схема алгоритма

Блок-схема алгоритма представляет работу макета лазерно-оптического комплекса. Алгоритм представляет из себя набор функций и циклов необходимых для выполнения действий по перемещению и считыванию фотодатчиков.

При первоначальном нажатии кнопки происходит перемещение лазера из «нулевой» позиции к первому датчику. Шаговые двигатели совершают оборот на заданное количество шагов. Датчик передаёт сигнал на микроконтроллер, тем самым определяя местоположение шпинделя. Далее система ожидает последующих действий (нажатия на кнопку) от пользователя (оператора). Данный алгоритм происходит циклически, пока пользователь или оператор не выключит установку.

Блок-схема алгоритма работы макета лазерно-оптического комплекса представлена на рис.3.1.





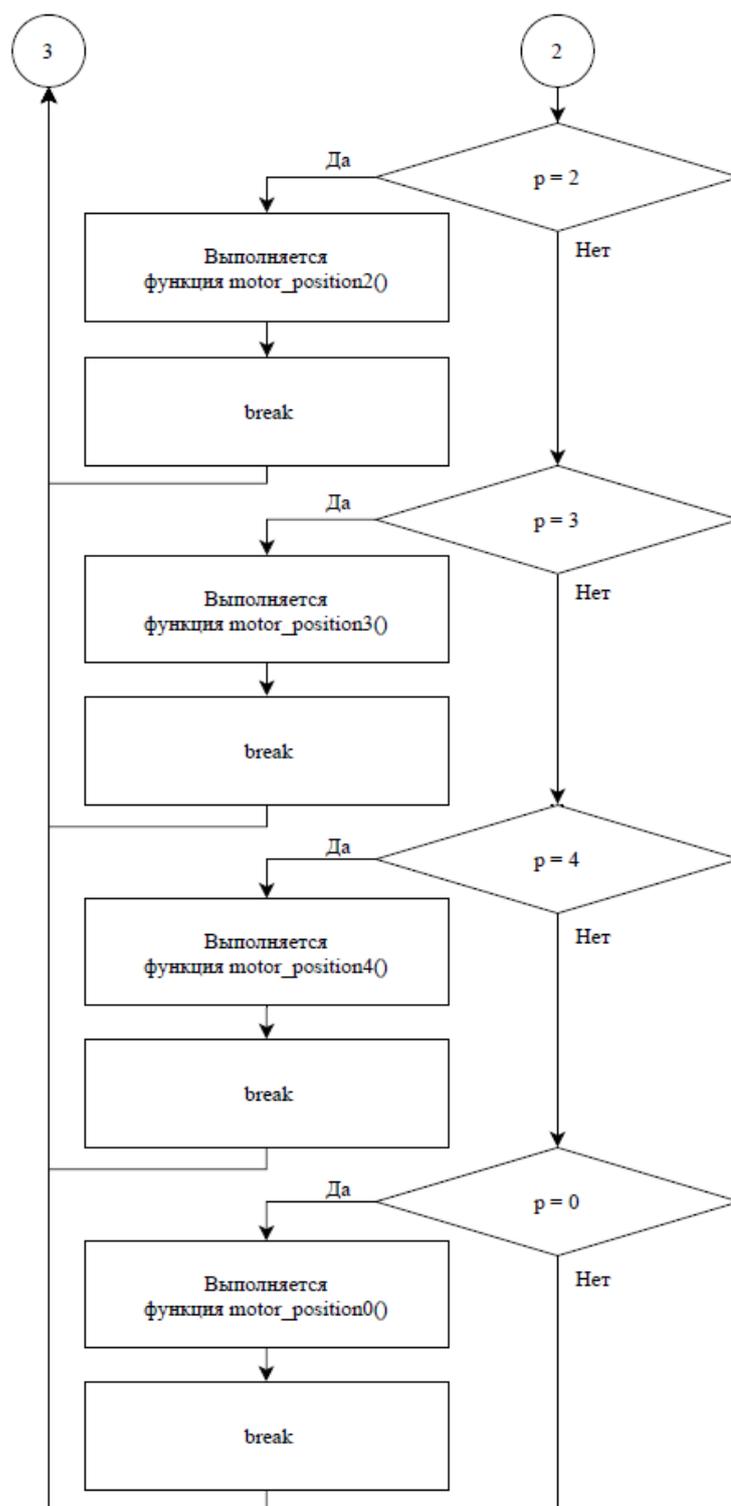


Рисунок 3.1 – Блок-схема алгоритма работы макета лазерно-оптического комплекса

3.2. Программа для практической модели

Основой для программирования Arduino служит упрощённый язык C++. Чтобы написать программу для работы с отладочным комплексом требуется:

- программное обеспечение Arduino IDE, которое производитель предоставляет в бесплатном доступе. Его необходимо скачать и установить на ПК с официального сайта или многочисленных форумов;

- сама плата Arduino;

- кабель USB – для подключения.

Код программы, написанный с использованием вышеуказанного программного обеспечения, носит название «скетч». Для сокращения величины кода создаются библиотеки, многие из которых являются стандартными и встроенными сразу при установке. Также есть большое количество сторонних библиотек, созданных другими пользователями, которые также можно загрузить для использования.

Полный текст программы представлен в приложении А.

3.2.1 Используемые библиотеки и вводные данные

В скетче, используемом для создания макета, применяются следующие библиотеки:

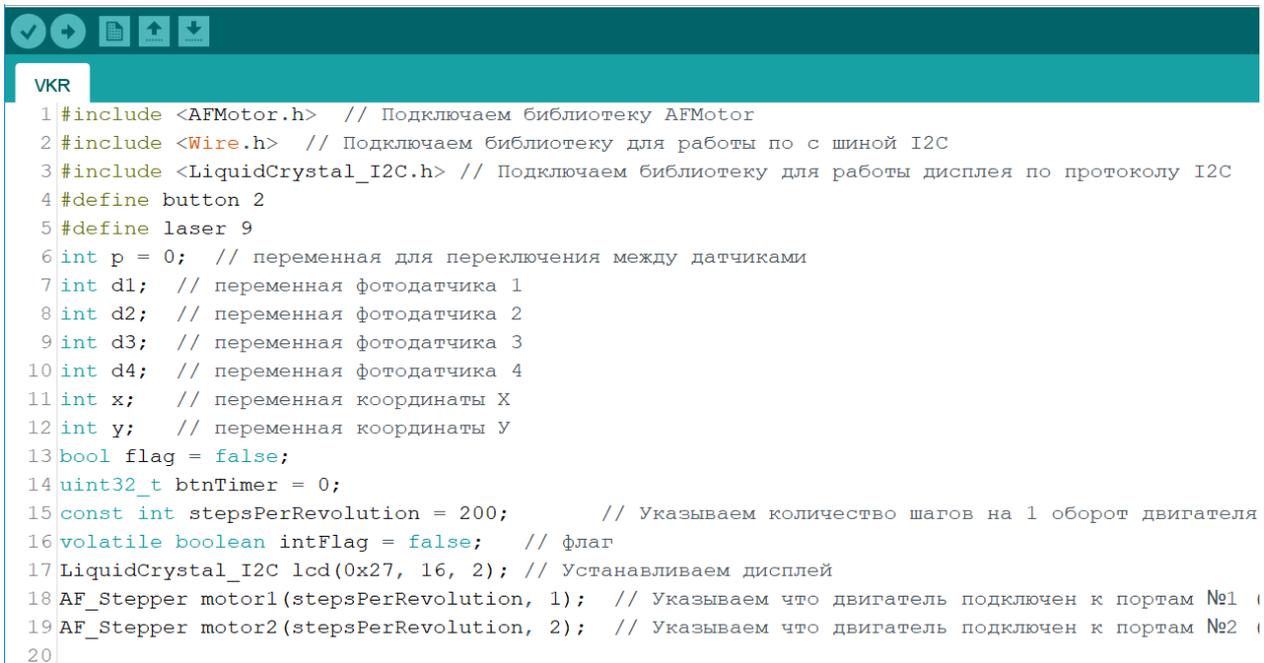
- `#include <Wire.h>` – предназначена для взаимодействия с устройствами по шине I2C, которые используются в данном проекте;

- `#include <LiquidCrystal_I2C.h>` – служит для подключения и управления выбранным дисплеем;

- `#include <AFMotor.h>` – необходима для работы с драйвером и управлением двигателями.

После того, как в код добавили все библиотеки, описанные выше, определяем глобальные переменные для обозначения датчиков, координат и

кнопки с прерыванием, задаём размер и адрес дисплея, а также указываем количество шагов и обработку движения шаговых двигателей (рис.3.2).



```
1 #include <AFMotor.h> // Подключаем библиотеку AFMotor
2 #include <Wire.h> // Подключаем библиотеку для работы по с шиной I2C
3 #include <LiquidCrystal_I2C.h> // Подключаем библиотеку для работы дисплея по протоколу I2C
4 #define button 2
5 #define laser 9
6 int p = 0; // переменная для переключения между датчиками
7 int d1; // переменная фотодатчика 1
8 int d2; // переменная фотодатчика 2
9 int d3; // переменная фотодатчика 3
10 int d4; // переменная фотодатчика 4
11 int x; // переменная координаты X
12 int y; // переменная координаты Y
13 bool flag = false;
14 uint32_t btnTimer = 0;
15 const int stepsPerRevolution = 200; // Указываем количество шагов на 1 оборот двигателя
16 volatile boolean intFlag = false; // флаг
17 LiquidCrystal_I2C lcd(0x27, 16, 2); // Устанавливаем дисплей
18 AF_Stepper motor1(stepsPerRevolution, 1); // Указываем что двигатель подключен к портам №1
19 AF_Stepper motor2(stepsPerRevolution, 2); // Указываем что двигатель подключен к портам №2
20
```

Рисунок 3.2 – Начальные данные

3.2.2 Функция «setup»

Следующим шагом прописываем функцию «setup» (рис.3.3), она выполняется однократно сразу после запуска программы. В данной функции указывается скорость двигателей, включается подсветка дисплея и указывается нулевое значение координат. Также подключаем порт входа с подтягивающим резистором для кнопки и порт выхода для включения лазерного модуля. В завершении указывается условие для прерывания и функция, исполняемая при его срабатывании.

```
VKR
20
21 void setup() {
22   Serial.begin(9600); // открыли порт для связи
23   // подключили кнопку на D2 и GND
24   motor1.setSpeed(70); // Скорость двигателя 1 в минуту
25   motor2.setSpeed(70); // Скорость двигателя 2 в минуту
26   lcd.init();
27   lcd.backlight(); // Включаем подсветку дисплея
28   lcd.setCursor(0, 0); // Устанавливаем курсор в начало 1 строки
29   lcd.print ("X=");
30   lcd.print (0);
31   lcd.print (" Y=");
32   lcd.print (0);
33   lcd.print (" ");
34   pinMode(button, INPUT_PULLUP);
35   pinMode(laser, OUTPUT);
36   // D2 это прерывание 0
37   // обработчик - функция buttonTick
38   // FALLING - при нажатии на кнопку будет сигнал 0, его и ловим
39   attachInterrupt(0, buttonTick, FALLING);
40 }
```

Рисунок 3.3 – Функция «setup»

3.2.3 Функции «buttonTick» и «loop»

Для применения аппаратного прерывания применяется тактовая кнопка. При её нажатии логическая переменная «intFlag» принимает значение «true». Поскольку, кнопка не идеальна и контакт замыкается не сразу, то какое-то время возникает явление «дребезга» контактов. Для устранения данного явления можно воспользоваться программными и аппаратными методами. В функции «buttonTick» (рис.3.4) представлен программный метод, основанный на «millis», время гашения дребезга примем 200 миллисекунд.

```
42 void buttonTick() {
43   bool btnState = !digitalRead(2);
44   if (btnState && !flag && millis() - btnTimer > 200 ) {
45     intFlag = true;
46     btnTimer = millis();
47   }
48 }
```

Рисунок 3.4 – Функции «buttonTick»

Функция «loop» (рис.3.5) является бесконечным циклом, который микроконтроллер будет выполнять раз за разом. В нем используются и другие функции, описанные ниже.

Первоначально программа считывает датчики и посылает информацию на дисплей. Затем, если нажать кнопку и сработает прерывание начинается выполнение следующих действий:

1. к переменной счётчика нажатий «p» добавляется единица, её значение отображается во второй строке дисплея;
2. начинается выполнение условия в зависимости от конкретного значения переменной «p» начинается выполнение функции работы шаговых двигателей, которые перемещаются на заданные координаты.

```
50 void loop() {
51   scan();
52   disp_info();
53   if (intFlag) {
54     intFlag = false;    // сбрасываем флаг
55     p = p + 1;
56     if (p > 4) {
57       p = 0;
58     }
59     lcd.setCursor(0, 1); // Устанавливаем курсор в начало 2 строки
60     lcd.print (p);
61     switch (p) {
62       case 1:
63         motor_position1(); //выполнение функции перемещения стола
64         break;
65       case 2:
66         motor_position2();
67         break;
68       case 3:
69         motor_position3();
70         break;
71       case 4:
72         motor_position4();
73         break;
74       case 0:
75         motor_position0();
76         break;
77     }
78   }
79 }
```

Рисунок 3.5 – Функции «loop»

3.2.4. Остальные функции

Оставшиеся дополнительные функции выполняют действия, которые было возможно вынести от основного тела функции «loop» или которые повторяются несколько раз. Например, программы функций «moving» и «disp_info» (рис.3.6) предназначены для вывода на дисплей данных выполнении перемещения стола, а также о сканировании датчиков и отображении значения координат.

```
117 void moving() {
118   lcd.setCursor(0, 0);
119   lcd.print ("Moving      ");
120 }
121 void disp_info() {
122   lcd.setCursor(0, 0); // Устанавливаем курсор в начало 1 строки
123   lcd.print ("X=");
124   lcd.print (x);
125   lcd.print (" Y=");
126   lcd.print (y);
127   lcd.print ("      ");
128 }
```

Рисунок 3.6 – Функции «moving» и «disp_info»

Функция «scan» (рис.3.7) выполняет считывание аналогового сигнала с фоторезисторов и в зависимости от полученных значений определяет где находится шпиндельная головка с лазером задавая координаты для переменных «x» и «y».

```
130 void scan() {
131   d1 = analogRead (A0); //считывание сигнала с резистора 1
132   d2 = analogRead (A1); //считывание сигнала с резистора 2
133   d3 = analogRead (A2); //считывание сигнала с резистора 3
134   d4 = analogRead (A3); //считывание сигнала с резистора 4
135   if (d1 > 800) x = 20, y = 200; //вывод координат
136   else if (d2 > 800) x = 190, y = 190;
137   else if (d3 > 800) x = 100, y = 155;
138   else if (d4 > 800) x = 30, y = 115;
139   else x = 0, y = 0;
140 }
```

Рисунок 3.7 – Функция «scan»

Для движения шаговых двигателей используются функции «motor_position» (рис.3.8). В них указано какое количество шагов необходимо сделать, в каком направлении вращения, а также обозначение типа шага («SINGLE», «DOUBLE», «INTERLEAVE» или «MICROSTEP»).

```
82 void motor_position1()  
83 {  
84   moving();  
85   motor1.step(174, BACKWARD, INTERLEAVE );  
86   motor2.step(1445, BACKWARD, INTERLEAVE );  
87 }
```

Рисунок 3.8 – Функция «motor_position»

Выводы по разделу

В третьем разделе рассмотрены алгоритм работы программы, который необходим для более понятного отображения функционирования скетча. Иными словами, что будет, если случится, то или иное событие, отражённое в алгоритме. После чего, был разработан скетч – код программы устройства и рассмотрен процесс подключения и программирования микроконтроллера – для чего используется среда Arduino IDE и сопутствующие элементы, такие как: библиотеки. Затем описана процедура действий, выполняемых макетом лазерно-оптического комплекса.

Заключение

Итогом выполнения данной выпускной квалификационной работы является создание макета лазерно-оптического комплекса, который является практической визуализацией данной работы.

В ходе выполнения данной выпускной квалификационной работы было необходимо уделить внимание трём основным разделам.

В первом разделе были рассмотрены причины возникновения различного рода погрешности, приведены примеры расчёта абсолютной, относительной и суммарной погрешностей и способы выявления отклонений при работе с применением лазерно-оптического комплекса.

Во втором разделе были рассмотрены различные комплектующие элементы, которые возможно использовать при создании макета лазерно-оптического комплекса. Выбраны оптимальные элемент с учётом экономической и конструктивной целесообразности.

В третьем разделе рассмотрены алгоритм работы программы, который необходим для более понятного отображения функционирования скетча. Иными словами, что будет, если случится, то или иное событие, отражённое в алгоритме. После чего, был разработан скетч – код программы устройства и рассмотрен процесс подключения и программирования микроконтроллера – для чего используется среда Arduino IDE и сопутствующие элементы, такие как: библиотеки. Затем описана процедура действий, выполняемых макетом лазерно-оптического комплекса.

Несмотря на объем проделанной работы, устройство имеет недостатки и может быть модернизировано.

Список используемой литературы

1. Arduino IDE [Электронный ресурс]: «Download the Arduino IDE» URL: <https://www.arduino.cc/en/main/software> (дата обращения: 20.02.2021).
2. Arduino Mega [Электронный ресурс]: «Arduino Mega.» URL: <https://store.arduino.cc/usa/mega-2560-r3> (дата обращения: 20.02.2021).
3. Arduino Nano [Электронный ресурс]: «Arduino Nano.» URL: <https://store.arduino.cc/usa/arduino-nano> (дата обращения: 20.02.2021).
4. Arduino Uno [Электронный ресурс]: «Arduino Uno.» URL: <https://store.arduino.cc/usa/arduino-uno-rev3> (дата обращения: 20.02.2021).
5. Васюков А.В., Богуш Р.П. «ЭЛЕКТРОННЫЕ ПРИБОРЫ И УСТРОЙСТВА» / Васюков А.В., Богуш Р.П. учеб.-метод. комплекс / сост. и общ. ред. С.Н. Абраменко. В2 ч. Ч.1. – Новополюцк: ПГУ, 2008г. – 360 с.
6. Гжиров Р.И. «Программирование обработки на станках с ЧПУ» / Гжиров Р.И., – Л.: Машиностроение. Ленингр. отд-ние. 1990 г. – 588 с.
7. ГОСТ 20523-80 [Электронный ресурс]: «Устройства числового программного управления станками» URL: <https://docs.cntd.ru/document/1200023225> (дата обращения: 02.05.2021).
8. Грибковский В.П. «Полупроводниковые лазеры» /. Грибковский В.П., Учеб. пособие по спец. «Радиофизика и электроника». – Мн.: Университетское, 1988г. – 304 с.
9. Единая система конструкторской документации, «ОБОЗНАЧЕНИЯ БУКВЕННО-ЦИФРОВЫЕ В ЭЛЕКТРИЧЕСКИХ СХЕМАХ» [Электронный ресурс]: ГОСТ 2.710-81.URL: <http://www.pntd.ru/2.710.html> (дата обращения: 30.04.2021).
10. Информационный студенческий ресурс –Студопедия.Нет [Электронный ресурс]: «Принцип действия полупроводникового лазера» URL: https://studopedia.ru/23_3035_poluprovodnikovie-lazeri.html (Дата обращения: 01.02.2019).
11. Каримова К.Р. «Исследование возможности применения технологии LoRa при проектировании систем позиционирования.» /

Каримова К.Р., Киреева Ю.М., Юмадилова И.Р. [и др.] УГАТУ, Уфа, Россия, 2019г. – 82с.

12. Манаев, Е. И. Основы радиоэлектроники / Манаев Е. И. — 3-е изд. — Москва: Радио и связь, 1985. — 511 с.

13. Менщиков Ю. «Ардуино на пальцах» / Менщиков Ю. – Минск, б/и, 2017г. – 62с.

14. Модуль лазера KY-008 [Электронный ресурс]: «Модуль лазера (красный)» URL: <https://datasheetspdf.com/pdf-file/1415009/ETC/KY-008/1> (дата обращения: 21.05.2021).

15. Описание интерфейса I2C [Электронный ресурс]: «Интерфейсная шина ИС (I2C).» URL: <http://easyelectronics.ru/interface-bus-iic-i2c.html> (дата обращения: 15.03.2021).

16. Петров С.А., Нефедьев А.И. «МИКРОКОНТРОЛЛЕРЫ ИЗМЕРИТЕЛЬНЫХ СИСТЕМ» / Петров С.А., Нефедьев А.И. учеб. Пособие, ВолгГТУ. – Волгоград, 2015. – 112 с.

17. Сборник технических статей ТЕН-LIB.RU [Электронный ресурс]: «Полупроводниковые лазеры» URL: <http://www.teh-lib.ru/кое/poluprovodnikovye-lazery.html> (Дата обращения: 01.02.2019).

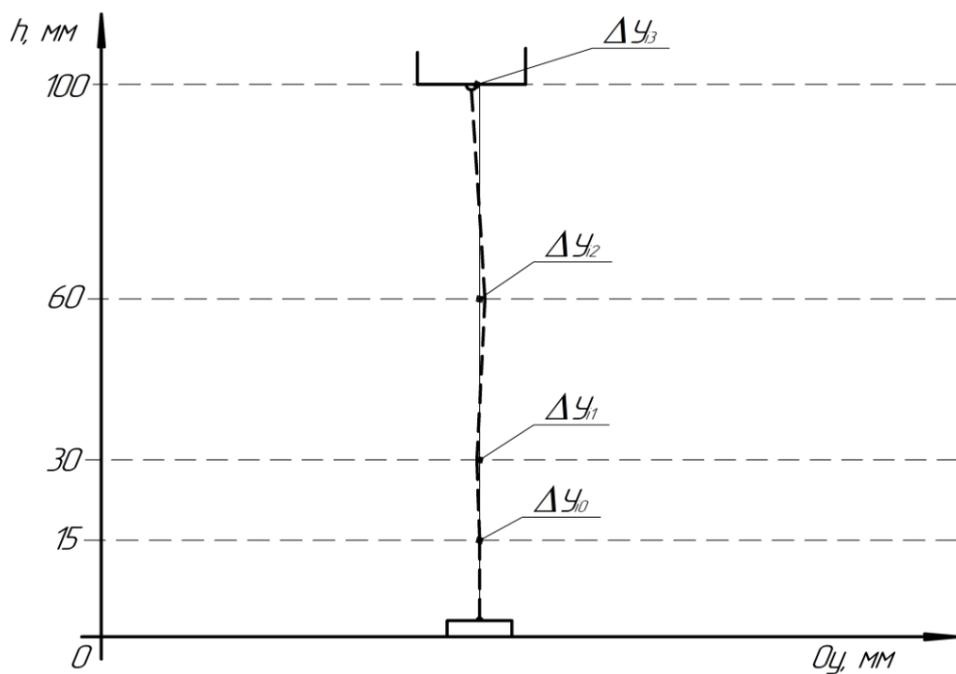
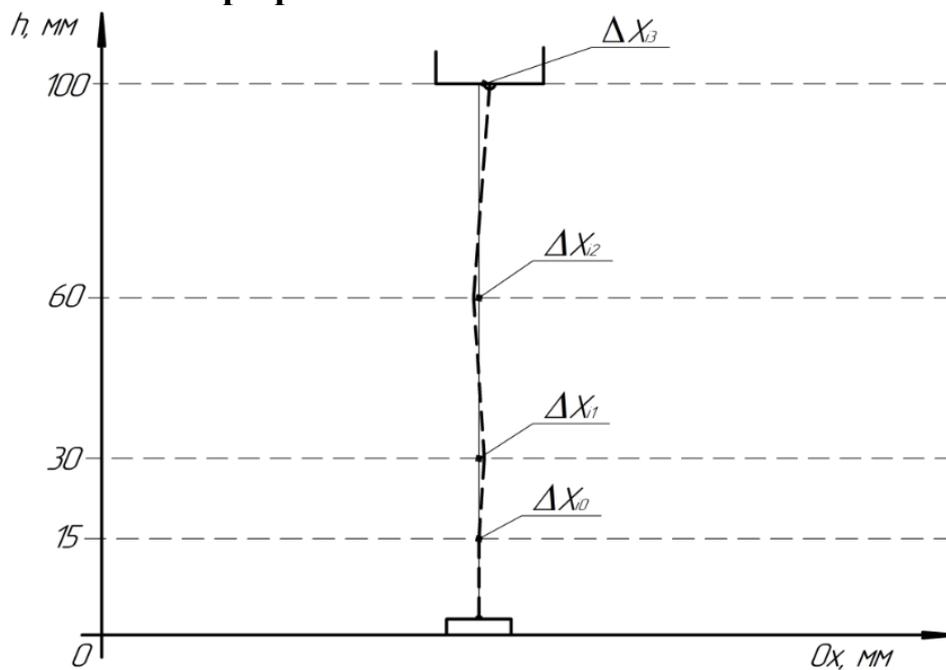
18. Схемы и справочные материалы – Электроника начинающим [Электронный ресурс]: «Фоторезисторы, фотодиоды, фототранзисторы. Общие понятия» URL: <http://begin.eschema.ru/?p=698> (Дата обращения: 01.02.2019).

19. Филачев А.М., Кравченко Н.В., Хакуашев П.Е., [и др.] [Электронный ресурс] «Кремниевый pin-фотодиод большой площади» URL: <http://bankpatentov.ru> (Дата обращения: 01.02.2019).

20. Филипп С. Д. ХОББС [Электронный ресурс]: «Усилители для фотодиодов на операционных усилителях» URL: https://www.elibrary.ru/download/elibrary_15136055_43602731.pdf (дата обращения: 15.03.2021).

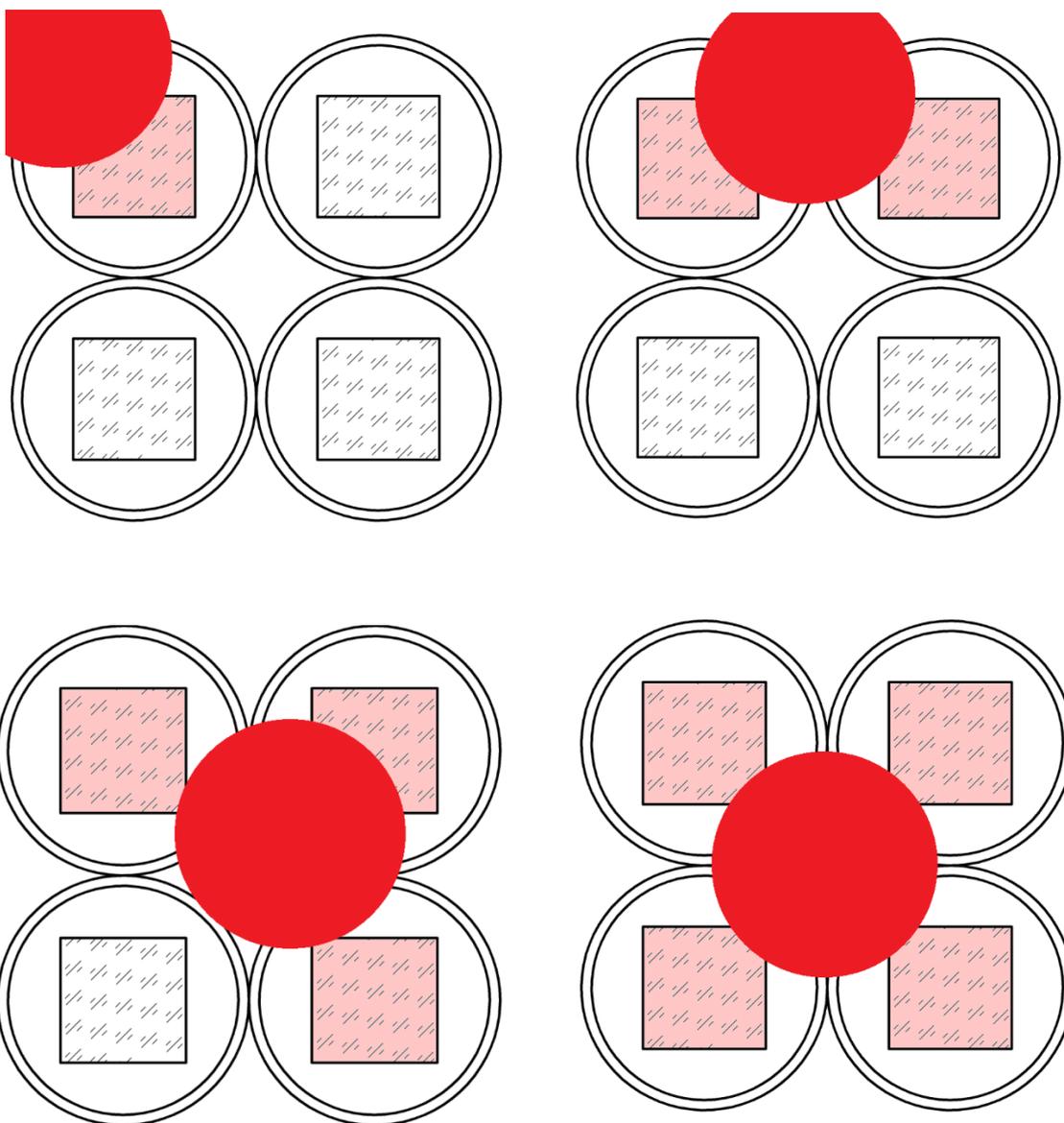
21. Agilent Technologies «Laser and Optics» / Agilent Technologies – User’s Manual, Volume Inc. 1992, 1996, 2000, 2002, 2007yy. – 310p.
22. Glenn E. Stutz «POLYGONAL SCANNERS: COMPONENTS, PERFORMANCE, AND DESIGN Optical and Laser Scanning» / Gerald F. Marshall, Niles, Michigan, handbook of, Lincoln Laser Company 2015y. – 272p.
23. Michael J Peterson «CNC Programming: Reference Book Paperback» / Michael J Peterson, 2008y. – 150p.
24. Mohsen Soori, Behrooz Arezoo, Mohsen Habibi « Dimensional and geometrical errors of three-axis CNC milling machines in a virtual machining system » / Mohsen Soori, Behrooz Arezoo, Mohsen Habibi, CAD/CAPP/CAM Research Center, Department of Mechanical Engineering, Amirkabir University of Technology (Tehran Polytechnic), 424 Hafez Avenue, Tehran 15875-4413, Iran, 2013y. – 10p.
25. User Manual. Datasheet CNC/Stepper Motor Shield «Handson Technology. 3-Axis CNC/Stepper Motor Shield for Arduino» / User Manual. Datasheet CNC/Stepper Motor Shield 2013y. – 12 p.

Приложение А
Графики абсолютной погрешности при перемещении по осям Ox и Oy
при различных значениях оси Oz



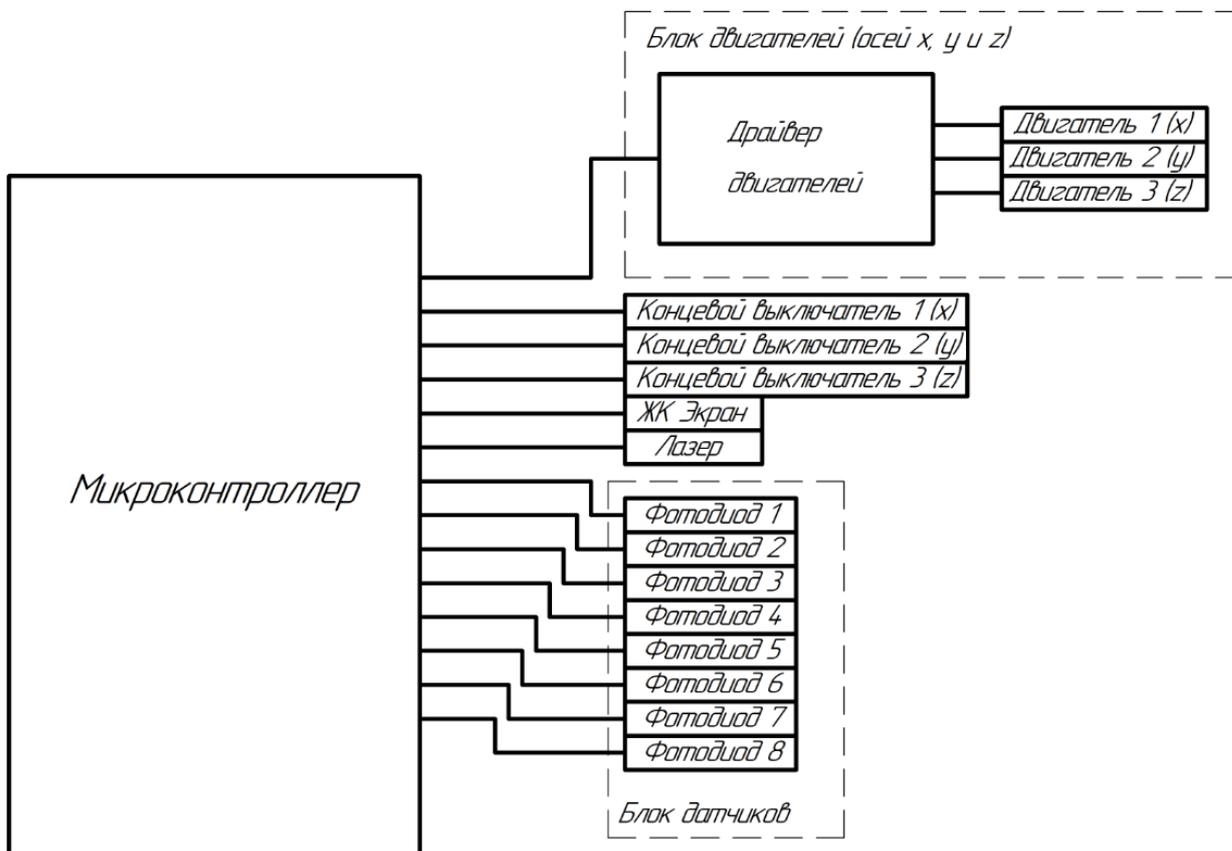
Приложение Б

Возможные состояния матрицы при попадании лазерного луча



Приложение В

Примерная структурная схема макета лазерно-оптического комплекса



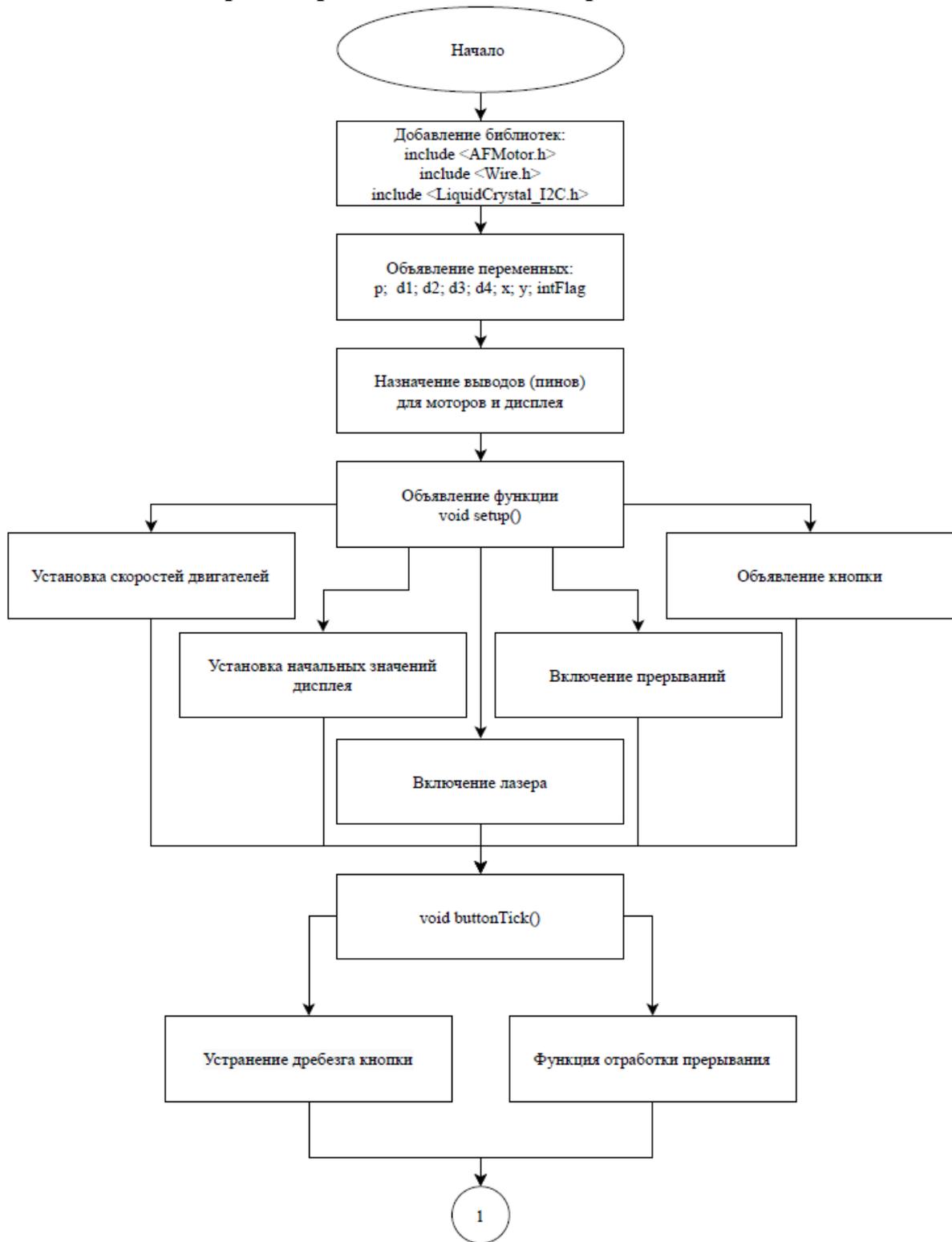
Приложение Г

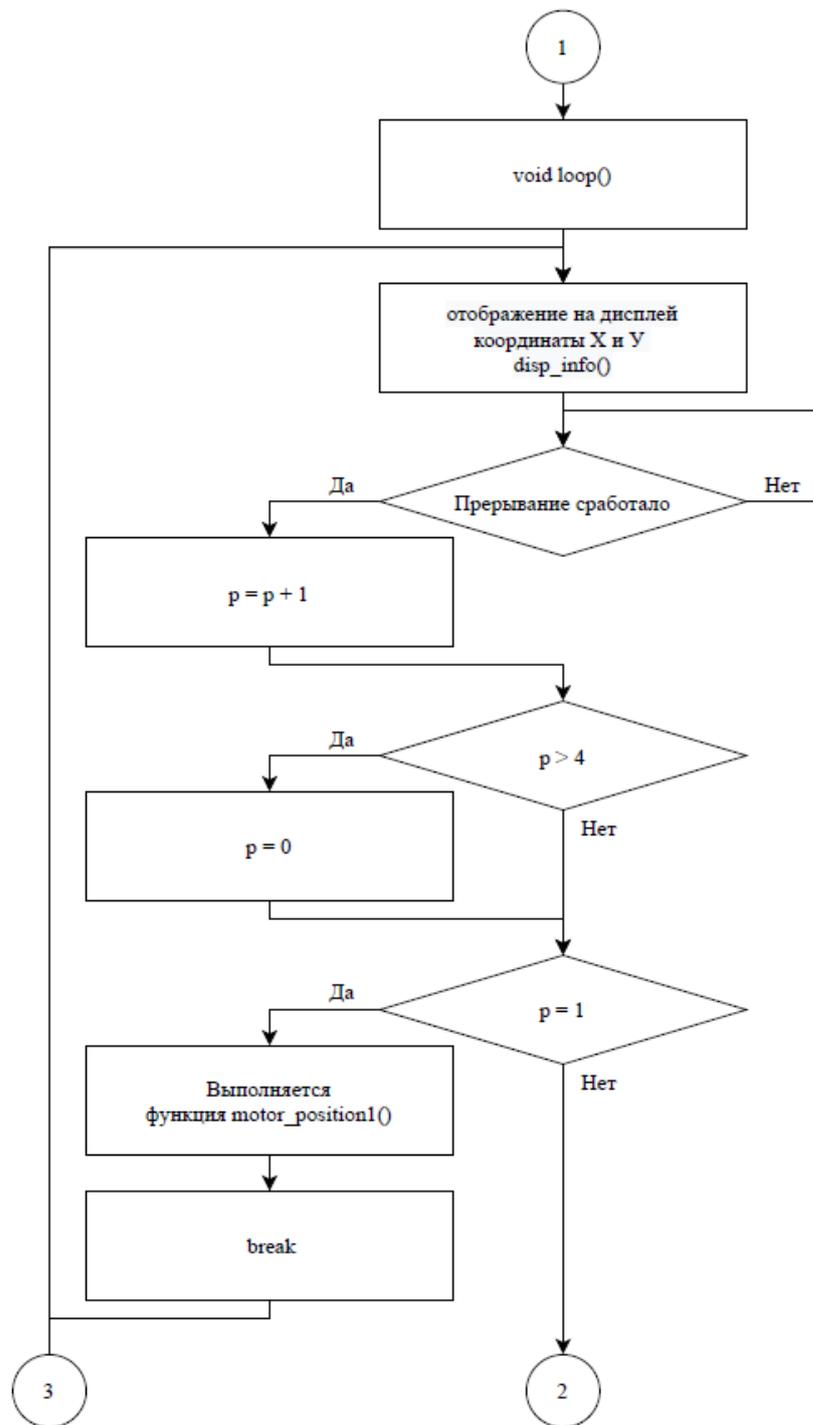
Перечень элементов к схеме принципиальной электрической

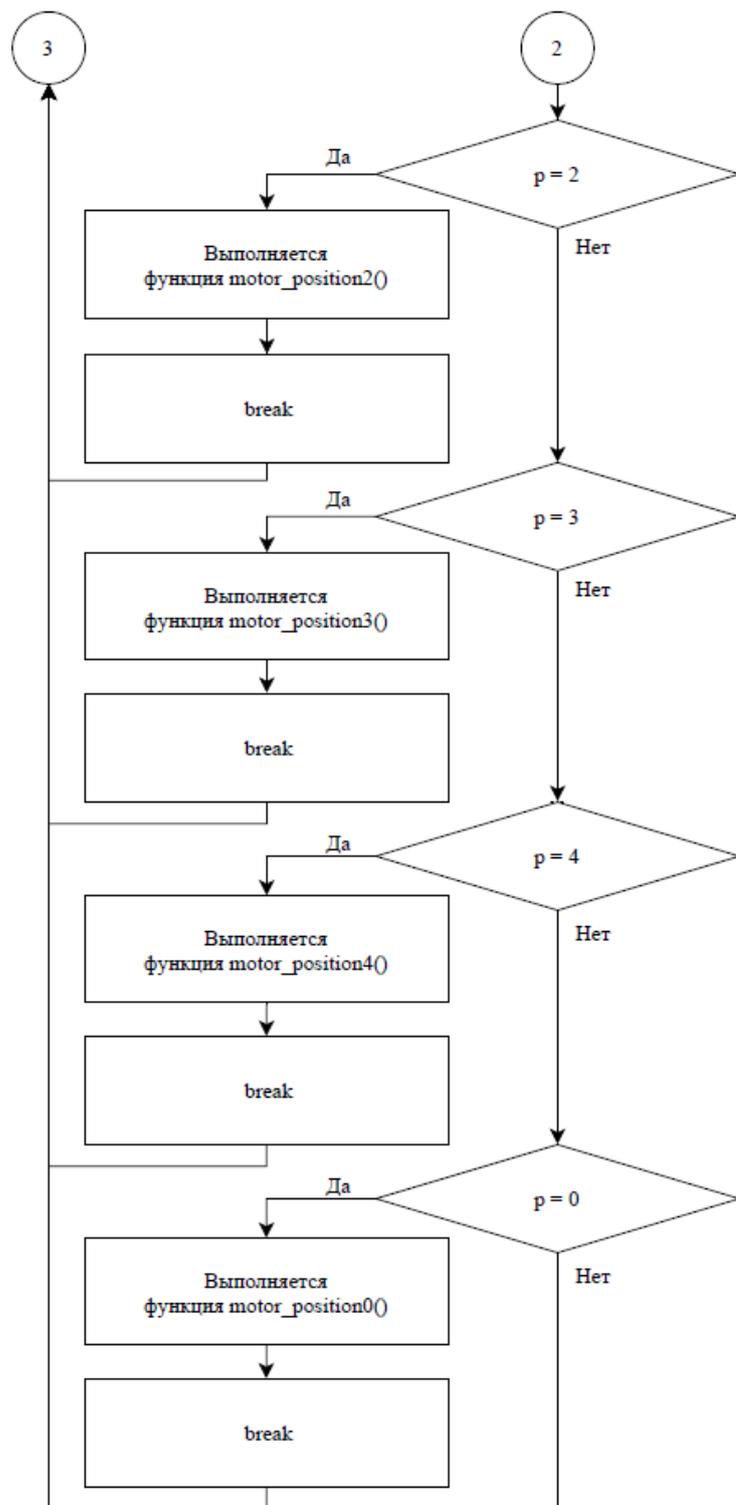
Поз.	Наименование	Кол.	Примечание				
<u>Микросхемы</u>							
1	АТмега 2560 на платформе Arduino Mega 2560	1	DD1				
2	I2C конвертер для дисплеев LCD1602	1	DD2				
3	Драйвер шагового двигателя A4988	3	DD3 ... DD5				
4	Операционный усилитель AD820	32	DA1 ... DA32				
<u>Резисторы</u>							
5	CF-100 (C1-4) на 100кОм	64	R1 ... R64				
<u>Диоды</u>							
6	Лазерный модуль KY-008	1	VD1				
7	Фотодиод ФДК-157	8	VD2 ... VD9				
<u>Выключатели (ключи)</u>							
8	Концевой выключатель KW12	3	SQ1 ... SQ3				
9	Кнопка тактовая KLS7-TS1204-7.3-180 (TC-12ET) (SWT-9)	1	SB1				
<u>Прочее</u>							
10	ЖК дисплей Arduino LCD 1602	1					
11	Шаговый двигатель Nema 17 (17HS4401)	3	M1 ... M3				
21-110304.53/09.214.04 ПЭ							
Изм.	Лист	№ докум.	Подп.	Дата	Лазерно-оптической модуль		
Разраб.	Паначев И.А.						
Проб.	Токарев Д.Г.						1
Н.контр.	Токарев Д.Г.				Перечень элементов ТГУ, ЭЛД-1702а		
Утв.	Шевцов А.А.						

Приложение Д

Блок-схема алгоритма работы макета лазерно-оптического комплекса







Приложение Е

Код программы для макета лазерно-оптического комплекса

```
#include <AFMotor.h> // Подключаем библиотеку AFMotor
#include <Wire.h> // Подключаем библиотеку для работы по с шиной I2C
#include <LiquidCrystal_I2C.h> // Подключаем библиотеку для работы дисплея по
протоколу I2C
#define button 2
#define laser 10
int p = 0; // переменная для переключения между датчиками
int d1; // переменная фотодатчика 1
int d2; // переменная фотодатчика 2
int d3; // переменная фотодатчика 3
int d4; // переменная фотодатчика 4
int x; // переменная координаты X
int y; // переменная координаты Y
bool flag = false;
uint32_t btnTimer = 0;
const int stepsPerRevolution = 200; // Указываем количество шагов на 1 оборот
двигателя
volatile boolean intFlag = false; // флаг
LiquidCrystal_I2C lcd(0x27, 16, 2); // Устанавливаем дисплей
AF_Stepper motor1(stepsPerRevolution, 1); // Указываем что двигатель подключен к портам
№1 (M1 - M2)
AF_Stepper motor2(stepsPerRevolution, 2); // Указываем что двигатель подключен к портам
№2 (M3 - M4)

void setup() {
  Serial.begin(9600); // открыли порт для связи
  // подключили кнопку на D2 и GND
  motor1.setSpeed(70); // Скорость двигателя 1 в минуту
  motor2.setSpeed(70); // Скорость двигателя 2 в минуту
  lcd.init();
  lcd.backlight(); // Включаем подсветку дисплея
  lcd.setCursor(0, 0); // Устанавливаем курсор в начало 1 строки
  lcd.print ("X=");
  lcd.print(0);
  lcd.print (" Y=");
  lcd.print(0);
  lcd.print (" ");
  pinMode(button, INPUT_PULLUP);
  pinMode(laser, OUTPUT);
  // D2 это прерывание 0
  // обработчик - функция buttonTick
  // FALLING - при нажатии на кнопку будет сигнал 0, его и ловим
  attachInterrupt(0, buttonTick, FALLING);
}

void buttonTick() {
  bool btnState = !digitalRead(2);
  if (btnState && !flag && millis() - btnTimer > 200 ) {
```

```

    intFlag = true;
    btnTimer = millis();
  }
}

void loop() {
  scan();
  disp_info();
  if (intFlag) {
    intFlag = false; // сбрасываем флаг
    p = p + 1;
    if (p > 4) {
      p = 0;
    }
    Serial.println(p);
    lcd.setCursor(0, 1); // Устанавливаем курсор в начало 2 строки
    lcd.print (p);
    switch (p) {
      case 1:
        digitalWrite(laser, HIGH);
        motor_position1(); //выполнение функции перемещения стола
        break;
      case 2:
        motor_position2();
        break;
      case 3:
        motor_position3();
        break;
      case 4:
        motor_position4();
        break;
      case 0:
        motor_position0();
        break;
    }
  }
}

void motor_position1()
{
  moving();
  motor1.step(110, BACKWARD, SINGLE );
  motor2.step(665, BACKWARD, SINGLE );
}

void motor_position2()
{
  moving();
  motor1.step(455, BACKWARD, SINGLE );
  motor2.step(215, FORWARD, SINGLE );
}

```

```

void motor_position3()
{
  moving();
  motor1.step(360, BACKWARD, SINGLE );
  motor2.step(250, BACKWARD, SINGLE );
}

void motor_position4()
{
  moving();
  motor1.step(8, BACKWARD, SINGLE );
  motor2.step(555, FORWARD, SINGLE );
}

void motor_position0()
{
  moving();
  motor1.step(933, FORWARD, SINGLE );
  motor2.step(145, FORWARD, SINGLE );
}

void disp_info() {
  lcd.setCursor(0, 0); // Устанавливаем курсор в начало 1 строки
  lcd.print ("X=");
  lcd.print(x);
  lcd.print (" Y=");
  lcd.print(y);
  lcd.print (" ");
}

void scan() {
  d1 = analogRead (A0); //считывание сигнала с резистора 1
  d2 = analogRead (A1); //считывание сигнала с резистора 2
  d3 = analogRead (A2); //считывание сигнала с резистора 3
  d4 = analogRead (A3); //считывание сигнала с резистора 4
  if (d1 > 900) x = 20, y = 200; //вывод координат
  else if (d2 > 900)x = 190, y = 190;
  else if (d3 > 900) x = 100, y = 155;
  else if (d4 > 900) x = 30, y = 115;
  else x = 0, y = 0;
}

void moving() {
  lcd.setCursor(0, 0);
  lcd.print ("Moving ");
}

```