

Олимпиада «Высшая проба» проводится при поддержке Сбера, приветствуем участников соревнования!



Сбер – лидер по созданию инноваций, которые эффективно интегрируются в жизнь людей.

Поздравляем – ты являешь участником заключительного этапа олимпиады по направлению «Инженерные науки».

Вперед к наукоемким технологиям! Желаем тебе блистательной победы!

Время выполнения – 200 минут.

Пишите разборчиво. Ответ пишите на странице с соответствующим номером вопроса. Если используете дополнительный лист, обязательно напишите об этом на основном листе ответа. Если не знаете ответа, ставьте прочерк. Черновики не оцениваются.

Нахождение каждого ответа должно сопровождаться подробной аргументацией. Участником должны быть приведены необходимые для объяснения логики решения рисунки, формулы, аналитические обоснования.

Приступая к выполнению заданий, вы подтверждаете, что профиль и класс заданий соответствуют сведениям, указанным вами при регистрации.

(Максимальное количество баллов – 100)

Задача 1 (Максимум – 25 баллов).

Четыре ретранслятора связи №1, №2, №3 и №4 образуют ориентированную сеть. Наличие канала задано логическими высказываниями: если высказывание истинно - ребро существует, ложно - ребра нет. Дано: $A=1$, $B=0$. Операции: $*$ — И, $\oplus$ — XOR, $\rightarrow$ — импликация, $\neg$ — НЕ.

Матрица (логические высказывания):

	1	2	3	4
1	$(A*B)\oplus(\neg A)$	$(A\rightarrow B)\oplus A$	$\neg(A\oplus B)$	$(\neg B)\rightarrow A$
2	$(A\rightarrow B)*\neg B$	$(A*B)\oplus B$	$(A\oplus B)\rightarrow A$	$\neg(A*B)$
3	$(A*B)\rightarrow(A\oplus B)$	$(\neg B)\oplus(A\rightarrow A)$	$(A*B)\oplus(A\rightarrow B)$	$(A\oplus B)*\neg A$
4	$\neg B$	$(A\rightarrow B)\oplus(\neg B)$	$(A\oplus B)\oplus(A*B)$	$(A*B)\rightarrow B$

Если ребро существует, то его вес (в часах) записан в указанной системе счисления. Для отсутствующих рёбер вес равен 0.

Матрица весов:

	1	2	3	4
1	0	111001_2	0	20_8
2	0	0	57_8	10111_2
3	$2D_{16}$	0	0	0
4	11010_2	$3A_{16}$	52_8	0

Ответьте на вопросы:

- 1) По логическим высказываниям определите количество ориентированных рёбер в графе (без петель).
- 2) Переведите все ненулевые веса в десятичную систему и найдите максимальный вес ребра.
- 3) Найдите общий вес графа (сумма весов всех существующих рёбер в десятичной системе).
- 4) Определите минимальный вес пути из узла 1 в узел 4.
- 5) Определите максимальный вес пути из узла 2 в узел 4 по простым путям (без повторения вершин).

В ответы вносите целые числа, полученные при решении.

Решение

1. Значения высказываний при $A=1$, $B=0$ дают матрицу смежности (без диагонали): $[[0,1,0,1],[0,0,1,1],[1,0,0,0],[1,1,1,0]]$. Число рёбер: 8.
2. Перевод весов в десятичную: $111001_2=57$, $20_8=16$, $57_8=47$, $10111_2=23$, $2D_{16}=45$, $11010_2=26$, $3A_{16}=58$, $52_8=42$. Максимальный вес ребра: 58.
3. Общий вес графа: 314.
4. Кратчайший путь $1 \rightarrow 4$: прямое ребро, вес 16.
5. Максимальный простой путь $2 \rightarrow 4$: $2 \rightarrow 3 \rightarrow 1 \rightarrow 4$, вес $47+45+16=108$.

Ответы

- 1) 8
- 2) 58
- 3) 314
- 4) 16
- 5) 108

Критерии оценивания:

- Правильный ответ в пункте - 5 баллов.
- Если допущены серьезные логические/алгебраические ошибки, то 1 балл за пункт.

Задача 2 (Максимум – 25 баллов).

При разработке блока питания портативного медицинского устройства используется схема, представленная на рисунке 1. Номиналы сопротивлений $R1 = 5 \text{ кОм}$, $R2 = 3 \text{ кОм}$, $R3 = 2 \text{ кОм}$, $R4 = 4,5 \text{ кОм}$, $R5 = 2 \text{ кОм}$, $R6 = 3 \text{ кОм}$, $R7 = 6 \text{ кОм}$. Номиналы емкостей конденсаторов $C1 = 0,1 \text{ мкФ}$, $C2 = 0,02 \text{ мкФ}$. Индуктивность катушки $L1 = 0,4 \text{ мГн}$ и внутреннее сопротивление $RL1 = 2 \text{ Ом}$. Диод $D1$ считать идеальным. Напряжение источника ЭДС $U = 36 \text{ В}$. Ключ Key замкнут в первый момент времени ($t = 0 \text{ с}$). В рамках проверки стабильности питания необходимо определить значения электрических параметров схемы для установившегося режима (после завершения переходных процессов).

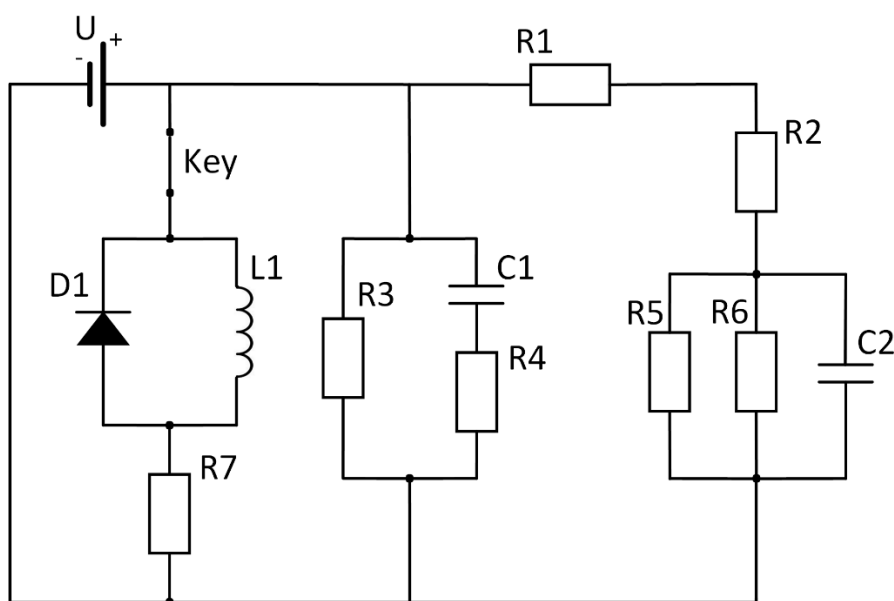


Рисунок 1. Электрическая схема.

Вопросы:

1. Определите значение тока через источник U для случая замкнутого ключа.
2. Определите ток через сопротивление $R5$ для случая замкнутого ключа.
3. Определите мощность ($P3$) выделяемую на сопротивление $R3$ для случая замкнутого ключа.
4. Определите заряд ($Q1$) на конденсаторе $C1$ для случая замкнутого ключа.
5. Определите энергию ($WL1$) катушки индуктивности $L1$ для случая замкнутого ключа.
6. Опишите какую функцию выполняет диод $D1$ в случае размыкания ключа в момент времени после завершения переходных процессов и имеет ли значение каким образом подключен диод

Критерий оценивания:

1. Формулы для расчета эквивалентных сопротивлений записаны верно – +4 балла
2. Значение тока через источник ЭДС рассчитано верно – +2 балла
3. Формулы для определения тока через сопротивление R_5 записаны верно – +2 балла
4. Значение тока через сопротивление R_5 определено верно – +2 балла
5. Формула для расчета мощности записана верно – +2 балла
6. Значение P_3 рассчитано верно – +2 балла
7. Формула для определения заряда конденсатора через емкость записана верно – +2 балла
8. Значение заряда на конденсаторе C_1 определено верно – +1 балла
9. Формула для энергии катушки индуктивности записана верно – +2 балла
10. Значение энергии катушки индуктивности W_{L1} определено верно – +1 балла
11. Приведено описание процесса происходящего в катушке индуктивности при отключении ее от источника тока – +3 балла
12. Приведено описание необходимости использования именно обратно параллельно включённого диода – +2 баллов

Ответы:

1. 27,91 мА.
2. 2,348 мА.
3. 648 мВт.
4. 3,6 мкКл.
5. 7,2 нДж.
6. Диод D_1 является защитным диодом. При открытии ключа (разрыв цепи от источника тока) возникает ЭДС самоиндукции, которая стремится поддержать протекающий через катушку ток. Но наличие обратно параллельно включённого диода обеспечивает путь для затухания тока через катушку.

Решение

1. В задании сказано, что необходимо определить значение электрических параметров после завершения переходных процессов. Это означает, что конденсаторы C_1 и C_2 можно представить в виде разрыва цепи, а катушку индуктивности L_1 в виде короткого замыкания. Проведем расчеты для такой схемы. Значение тока через источник напряжения можно рассчитать через эквивалентные сопротивления.

$$R_{56} = \frac{R5 \cdot R6}{R5 + R6} = \frac{2000 \cdot 3000}{2000 + 3000} = 1200 \text{ Ом}$$

$$R_{1256} = R1 + R2 + R56 = 5000 + 3000 + 1200 = 9200 \text{ Ом}$$

$$R_{7L} = R7 + RL1 = 6000 + 2 = 6002 \text{ Ом}$$

$$\frac{1}{R} = \frac{1}{R_{7L}} + \frac{1}{R3} + \frac{1}{R_{1256}}$$

$$R \approx 1289,8 \text{ Ом}$$

$$I = \frac{U}{R} = \frac{36}{1289,8} \approx 27,91 \text{ мА}$$

2. Ток через резистор R5 определяется через ток протекающий участке цепи R1, R2, R5, R6, C2:

$$I_{R1256} = \frac{U}{R_{1256}} = \frac{36}{9200} \approx 3,91 \text{ мА}$$

$$U_{56} = I_{R1256} \cdot R56 = 0,00391 \cdot 1200 \approx 4,692 \text{ В}$$

$$I_{R5} = \frac{U_{56}}{R5} = \frac{4,692}{2000} \approx 2,34 \text{ мА}$$

3. Мощность на резисторе R3 определяется по формуле:

$$P_{R3} = \frac{U^2}{R3} = \frac{36^2}{2000} = 0,648 \text{ Вт}$$

4. После завершения переходного процесса все напряжение на участке цепи C1-R4 будет падать на конденсаторе. Заряд конденсатора определяется по следующей формуле:

$$Q_{C1} = U \cdot C1 = 36 \cdot 0,1 \cdot 10^{-6} \approx 3,6 \text{ мкКл}$$

5. Энергия катушки индуктивности определяется с помощью тока, протекающего через нее:

$$I_{7L} = \frac{U}{R_{7L}} = \frac{36}{6002} \approx 5,99 \text{ мА}$$

$$W_{L1} = \frac{LI_{7L}^2}{2} = \frac{0,4 \cdot 10^{-3} (0,00599)^2}{2} \approx 7,17 \text{ нДж}$$

6. Диод D1 является защитным диодом. При открытии ключа (разрыв цепи от источника тока) возникает ЭДС самоиндукции, которая стремится поддерживать протекающий через катушку ток. Но наличие обратно параллельно включённого диода обеспечивает путь для затухания тока через катушку.

Задача 3 (Максимум – 25 баллов).

Для передачи данных о температуре океана на буйковых станциях используется инверсное помехоустойчивое кодирование. Температура измеряется с точностью 0.5°C (16 уровней от 0°C до 7.5°C) и представляется 4-битным словом a (от 0000 до 1111 соответственно).

При передаче по гидроакустическому каналу каждый бит может быть искажён помехой с вероятностью $q = 0.04$. Помехи в разных битах независимы.

Для повышения надёжности применяется кодер $K(a)$, который формирует 8-битное кодовое слово по правилу.

$$K(a) = a \& \begin{cases} a, & \text{если } \omega(a):2 \\ \bar{a}, & \text{иначе} \end{cases}$$

Декодер определяет наличие помех в полученном значении b по принципу чётного веса. Если $\omega(b)$ нечётный, то считают, что произошла одна ошибка, на позицию которой укажет побитовая сумма половин полученного вектора. Допустим полученное кодовое слово b состоит из двух половин b_1 и b_2 . Тогда, анализируется 4-битная строка

$$D(b_1 \& b_2) = b_1 \text{ XOR } \begin{cases} b_2, & \text{если } \omega(b_1):2 \\ \bar{b}_2, & \text{иначе} \end{cases}$$

- Если $\omega(D(b_1 \& b_2)) = 0$, то помех не было, и декодированным словом назовут b_1 .
- Если $\omega(D(b_1 \& b_2)) = 1$, то помеха пришлась на вторую половину и декодированным словом назовут b_1 .
- Если $\omega(D(b_1 \& b_2)) = 3$, то помеха пришлась на первую половину в позиции, где оказался 0 у $D(b_1 \& b_2)$. В b_1 меняют этот бит на противоположный и называют результат декодированным словом.
- Если $\omega(D(b_1 \& b_2))$, то было несколько помех, данная конструкция кода не восстанавливает такие случаи.

Задания:

- 1) Найдите вероятность того, что в переданном слове произойдёт ровно одна ошибка. Ответ округлите до трёх знаков после запятой.
- 2) Как выглядит $K(a)$ для температуры $6,5^{\circ}\text{C}$?
- 3) Декодируйте полученные данные от буя: 11100001 11001110 10100100
В ответе укажите последовательность принятых показателей температуры с точностью до полградуса (без указания единиц измерения), перечисленные в полученном порядке через запятую.
- 4) Получено кодовое слово 11001010. Сколько возможных комбинаций помех могло произойти в канале, если считать, что на одно кодовое слово приходится не более двух?

Комментарии:

- Назовем весом ω кодового слова a количество единиц в нём: $\omega(1001) = 2$.
- Назовём инверсией $\bar{a}$ кодового слова a битовую строку, в которой каждый бит кодового слова a заменён на противоположное значение. Например, $\overline{1001} = 0110$.
- Конкатенация кодовых слов a и b – операция склеивания: $a \& b$

- Побитовая сумма двух строк (или операция XOR) производится следующим образом: если соответствующие биты совпадают, то результат их сложения – ноль. Например, $0011 \text{ XOR } 0101 = 0110$.

Пример работы алгоритма: $K(0010) = 00101101$, если помеха во втором бите, то $D(01101101) = 0110 \text{ XOR } 1101 = 1011$. Декодировано 0010.

Ответы:

- 1) 0.240
- 2) 11010010
- 3) 7.0, 6.0, 5.5
- 4) 28

Решения:

- 1) $P = C_8^1 q(1 - q)^7 = 8 \cdot 0.04 \cdot 0.96^7 \approx 0.240$.
- 2) $a = 1101$. $K(1101) = 11010010$
- 3) 11100001: $D(1110 \& 0001) = 0000$. $\omega(D) = 0$. Ошибок нет. Декодировано: 1110 – 7
11001110: $D(1100 \& 1110) = 1100 \text{ XOR } 1110 = 0010$. Декодировано: 1100 – 6
10100100: $D(1010 \& 0100) = 1010 \text{ XOR } 0100 = 1110$. Декодировано: 1011 – 5.5
- 4) $D(1100 \& 1010) = 0110$. $\omega(D) = 2$, значит произошло более одной помехи. Исходя из условия их ровно 2. Тогда их количество $C_8^2 = 28$

Регламент проверки

Номер задания	Макс балл	Начисления баллов	Штрафы
1	5	○ Использована формула Бернулли: 3 балла. ○ Верные вычисления: 1 балл. ○ Округление до трёх знаков: 1 балл.	Вычислительная ошибка, ошибка в преобразованиях или размерностях: -1 балл.
2	6	○ Верно определено исходное слово a: 2 балла. ○ Проверка чётности веса: 2 балла. ○ Построение $K(a)$: 2 балла.	Вычислительная ошибка, ошибка в преобразованиях или размерностях: -1 балл.
3	7	○ Верно декодировано слово: по 2 балла за каждое ○ Верно указана последовательность температур: 1 балл	Вычислительная ошибка, ошибка в преобразованиях или размерностях: -1 балл.
4	7	○ Определение числа ошибок (2): 3 балла ○ Расчёт количества комбинаций: 4 балла	Вычислительная ошибка, ошибка в преобразованиях или размерностях: -1 балл.

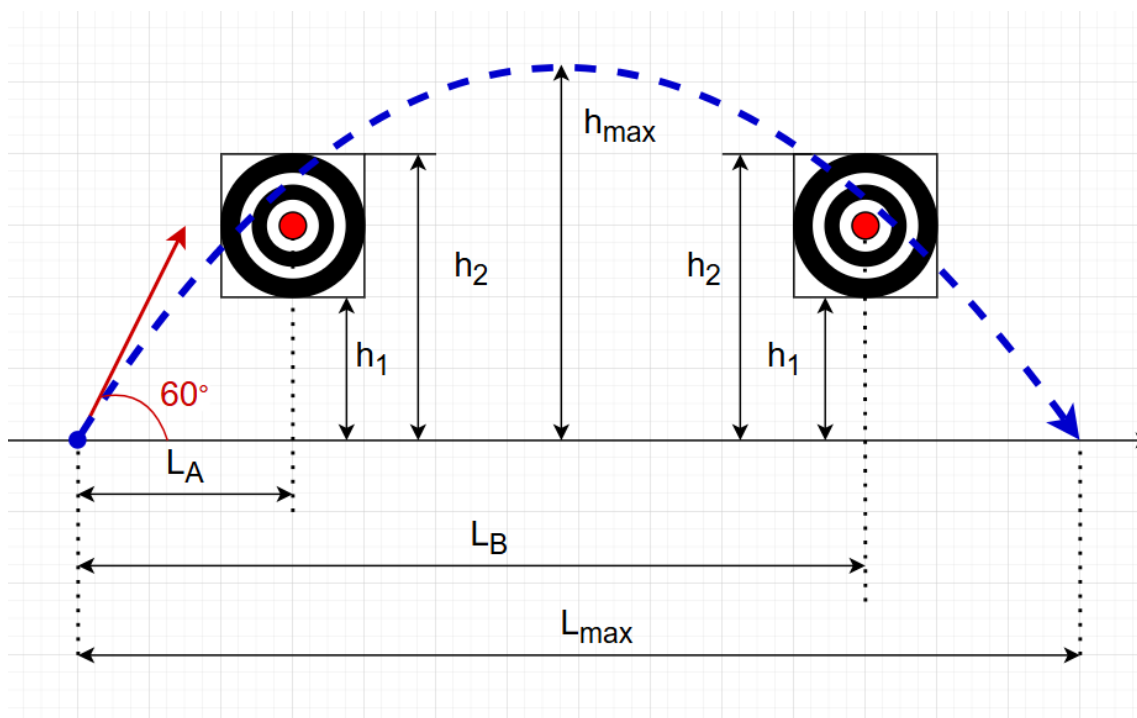
Задача 4 (Максимум – 25 баллов).

На военных учениях пушка выстреливает ядро с начальной скоростью $V_0 = 230$ м/с под углом $\alpha = 60^\circ$ к горизонту. Пушечное ядро пролетает в воздухе по баллистической траектории без трения и по задумке попадает в квадратную мишень, находящуюся на расстоянии L , верхний и нижний край которой расположены на высотах $h_2 = 3,5$ м и $h_1 = 1.5$ м соответственно.

Однако в силу несовершенности технических характеристик пушки присутствует погрешность, т.е. не всегда можно гарантировать точное попадание снаряда в мишень. Считайте, что ядро попадает внутрь круга радиусом $R = 1$ м, центр которого — это расчётная точка попадания снаряда, с вероятностью 100%. Вероятность попадания вне круга равна нулю. Размером ядра пренебречь.

1. Найдите, какие значения подойдут для L , чтобы ядро попадало в мишень с вероятностью более 50% (в метрах, округлив до десятых).
2. Каким должно быть L , чтобы снаряд попадал в мишень со 100% вероятностью (в метрах, округлив до десятых)?
3. С какой расчётной скоростью пробивает ядро мишень, если можно сказать, что оно точно попадёт в мишень (в м/с, округлив до десятых)?
4. До какой максимальной высоты поднялся снаряд (в метрах, округлив до десятых)?
5. При каком угле выстрела снаряд бы улетел на максимальное расстояние, если бы не было мишени (в градусах, округлив до целых)?

Решение:



Вопрос 1 и 2.

Анализ задания в части геометрии мишени и «круга погрешности».

Необходимо учесть, что при расчете в эталонном решении учитывалось $g = 9.8 \text{ м/с}^2$ (итоговый ответ может незначительно отличаться при $g = 10 \text{ м/с}^2$).

Дано:

1. Мишень: Квадрат высотой от 1.5 м до 3.5 м. То есть высота мишени $H = 2 \text{ м}$, а её центр находится на высоте 2.5 м.
2. Круг погрешности: Ядро с вероятностью 100% попадает в круг радиусом $R = 1 \text{ м}$. Это значит, что разброс ядра по вертикали составляет 2 метра (от -1 до +1 м относительно расчетной точки).

Условие 100% вероятности попадания: Так как высота мишени (2 м) в точности равна диаметру круга погрешности (2 м), 100% попадание возможно только в том случае, если расчетная точка (центр круга) находится идеально в центре мишени — на высоте $y = 2.5 \text{ м}$.

Условие вероятности > 50%: При равномерном распределении внутри круга, вероятность попадания более 50% достигается, когда центр круга находится внутри границ мишени (т.е. расчетная высота y находится в диапазоне от 1.5 до 3.5 м). В этом случае более половины площади круга перекрывается мишенью.

Уравнение траектории: Чтобы найти высоту y в любой точке на расстоянии x , используем формулу:

$$y(x) = x \cdot \tan(\alpha) - \frac{g \cdot x^2}{2 \cdot V_0^2 \cdot \cos^2(\alpha)}$$

Подставим числа:

- $\tan(60^\circ) = \sqrt{3} \approx 1.73205$
- $\cos(60^\circ) = 0.5 \Rightarrow \cos^2(60^\circ) = 0.25$
- $V_0^2 = 230^2 = 52900$

Получаем рабочее уравнение:

$$y(x) = 1.73205x - 0.00037051x^2$$

Нахождение дистанции L :

Нам нужно найти такие x , при которых высота y принимает значения центра мишени (2.5 м) или её границ (1.5 и 3.5 м). Решим уравнение для $y = 2.5$:

$$0.00037051x^2 - 1.73205x + 2.5 = 0$$

Это квадратное уравнение дает два корня (две дистанции), так как парабола пересекает высоту 2.5 м дважды. Для нахождения возможных значений L для двух случаев вероятности попадания необходимо учесть 2 возможных случая расположения мишени по оси X (см. рисунок):

1. Случай А: Восходящая ветвь (близко к пушке). Ядро только вылетело и пересекает мишень снизу вверх.
 - Для 100% вероятности ($y = 2.5$): $L \approx 1.4$ м.
 - Для $> 50\%$ вероятности ($y \in [1.5; 3.5]$): L в диапазоне от 0.9 до 2.0 м.
2. Случай Б: Нисходящая ветвь (дальняя дистанция). Ядро пролетело через вершину параболы и падает на мишень.
 - Для 100% вероятности ($y = 2.5$): $L \approx 4673.3$ м.
 - Для $> 50\%$ вероятности ($y \in [1.5; 3.5]$): L в диапазоне от 4672.7 до 4673.9 м.

При $g = 10 \text{ м/с}^2$:

- L_A (на взлете): ≈ 1.4 м.

- L_B (на излете): ≈ 4579.9 м.

Для вероятности $> 50\%$:

- На взлете диапазон L от 0.9 до 2.0 м
- На излете диапазон L примерно от 4578.6 до 4581.3 м.

Вопрос 3.

Согласно закону сохранения энергии, полная скорость ядра на высоте y не зависит от того, летит оно вверх или вниз. Она зависит только от начальной скорости и текущей высоты:

$$V = \sqrt{V_0^2 - 2gy}$$

Для $y = 2.5$ м:

$$V = \sqrt{230^2 - 2 \cdot 9.8 \cdot 2.5} = \sqrt{52900 - 49} = \sqrt{52851} \approx \mathbf{229.9 \text{ м/с}}$$

Вопрос 4.

Максимальная высота достигается в тот момент, когда вертикальная составляющая скорости V_y становится равной нулю. Формула для расчета:

$$H_{\max} = \frac{V_0^2 \sin^2(\alpha)}{2g}$$

Расчет:

1. $\sin^2(60^\circ) = (0.866)^2 = 0.75$
2. $H_{\max} = \frac{230^2 \cdot 0.75}{2 \cdot 9.8}$
3. $H_{\max} = \frac{52900 \cdot 0.75}{19.6} = \frac{39675}{19.6} \approx 2024.234...$

Ответ: 2024.2 м (При $g = 10 \text{ м/с}^2$: 1983.8 м)

Вопрос 5.

Дальность полета $L_{\max}$ определяется формулой:

$$L = \frac{V_0^2 \sin(2\alpha)}{g}$$

Чтобы дальность L была максимальной, значение $\sin(2\alpha)$ должно быть максимально возможным. Максимум функции синуса равен **1**, и он достигается при угле **90**.

Следовательно:

$$2\alpha = 90^\circ$$

$$\alpha = 45^\circ$$

Это классическое правило физики: в вакууме (без учета сопротивления воздуха) максимальная дальность полета достигается при выстреле под углом **45 градусов**.

Критерии оценки

Номер вопроса	Макс балл	Начисление баллов
1	9	Правильная интерпретация условия вероятности попаданий в цель 50%, учтены оба случая расположения мишени: 4 б. Приведено верно уравнение для траектории движения: 3 б. Проведен верный расчет: 2 б.
2	6	Правильная интерпретация условия вероятности попаданий в цель 100%, учтены оба случая положения мишени: 4 б. Проведен верный расчет: 2 б.
3	4	Приведена правильная формула: 3 б. Расчет верен: 1 б.
4	3	Приведена правильная формула: 2 б. Расчет верен: 1 б.
5	3	Приведена правильная формула: 2 б. Расчет верен: 1 б.

Задача 5 (Максимум – 25 баллов).

Рассматривается система автоматического управления межвагонной дверью поезда на базе микроконтроллера Arduino, предназначенная для управления открытием и закрытием двери по сигналу датчика силы нажатия.

На рисунке 2 представлена монтажная схема разработанного прототипа, она включает в себя:

- микроконтроллер Arduino Uno,
- делитель напряжения, состоящий из резисторов R1-R5 с известными сопротивлением $R1 = 24 \text{ кОм}$, $R2 = R3 = 6 \text{ кОм}$, $R4 = 2 \text{ кОм}$,
- идеальный мотор, корректно подключенный через драйвер мотора L293D и обеспечивающий вращение как по часовой стрелке, так и против.
- датчик силы нажатия, выполняющий роль кнопки управления и позволяющий измерять силу в диапазоне $[0; 10] \text{ Н}$.
- RGB-светодиод для индикации текущего состояния системы,
- монтажные платы и соединительные провода.

Проанализируйте монтажную схему и программный код (рисунки 3а,3б), и ответьте на следующие вопросы:

1. Перерисуйте монтажную схему делителя напряжения в виде электрической схемы цепи. Найдите сопротивление R5, если известно, что сопротивление всего делителя напряжения равняется $9,1 \text{ кОм}$.
2. Объясните работу программного кода построчно, поясните основную логику алгоритма работы микроконтроллера, представленных функций.
3. Какие цвета светодиода используются для индикации состояния системы? Запишите их RGB-коды и поясните как формируется цвет при использовании цветовой модели RGB.
4. Нарисуйте диаграмму состояний системы и опишите условия переходов между состояниями.
5. Какую минимальную сильную нужно приложить к датчику силы для инициации процесса открытия двери, если делитель напряжения в данной схеме обеспечивает максимальное значение $\text{fsrValue} = 829$? Учтите, что калибровочная характеристика датчика силы является линейной.

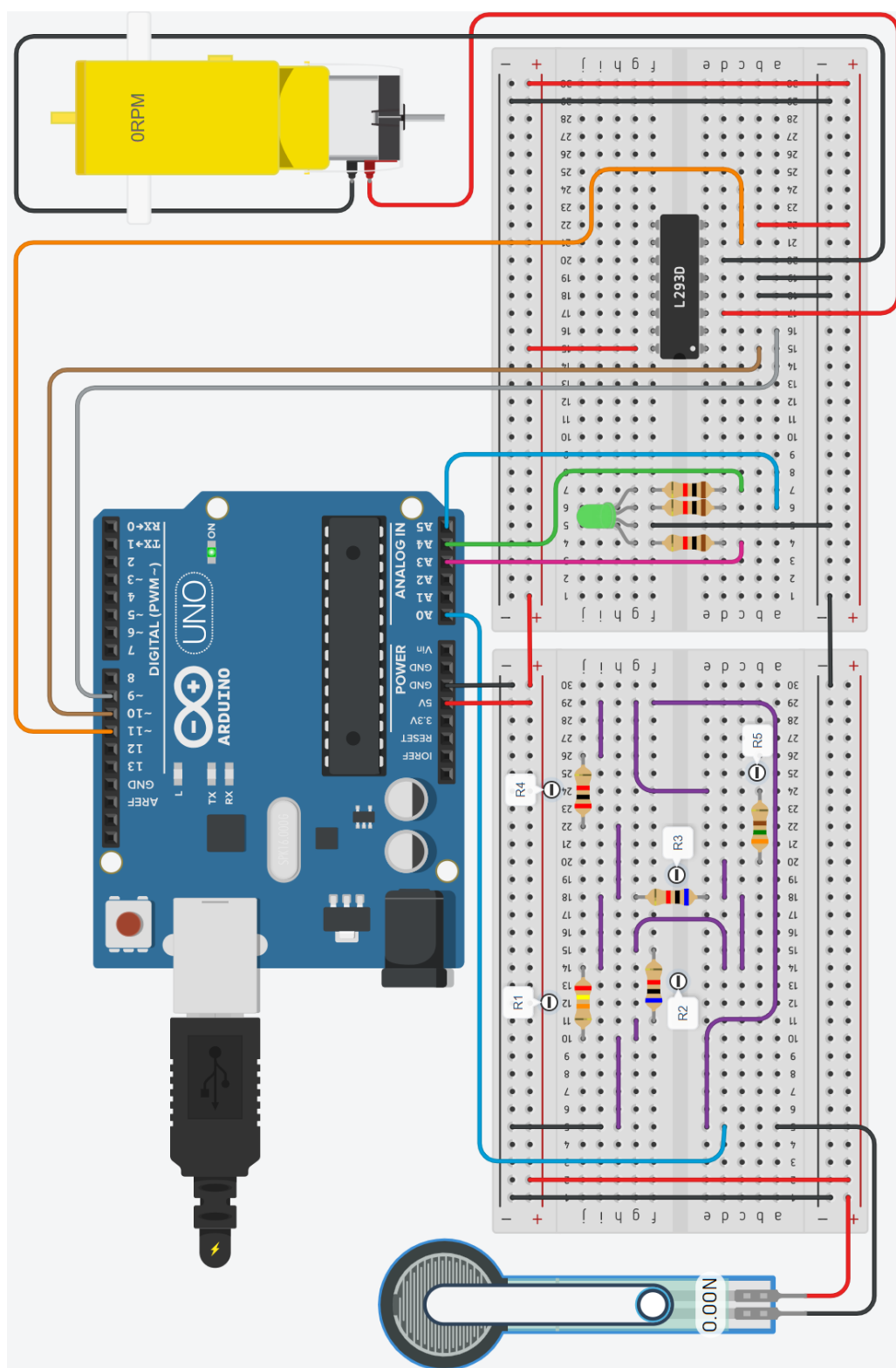


Рисунок 2. Монтажная схема

```
1  const int fsrPin    = A0;
2  const int ledRed    = A3;
3  const int ledGreen  = A4;
4  const int ledBlue   = A5;
5  const int motorPWM  = 10;
6  const int motorIN1  = 9;
7  const int motorIN2  = 11;
8  const int PRESS_THRESHOLD = 100;
9  const int HALF_POWER = 128;
10 const int FULL_POWER = 255;
11 const unsigned long MOTOR_TIME = 6000;
12 const unsigned long NO_PRESS_TIMEOUT = 4000;
13 enum MotorState {
14     IDLE,
15     OPENING,
16     WAIT_BEFORE_CLOSING,
17     CLOSING
18 };
19 MotorState state = IDLE;
20 unsigned long stateTimer = 0;
21 unsigned long lastPressTime = 0;
22
23 void motorStop() {
24     analogWrite(motorPWM, 0);
25     digitalWrite(motorIN1, LOW);
26     digitalWrite(motorIN2, LOW);
27 }
28 void motorOpen() {
29     digitalWrite(motorIN1, HIGH);
30     digitalWrite(motorIN2, LOW);
31     analogWrite(motorPWM, FULL_POWER);
32 }
33 void motorClose() {
34     digitalWrite(motorIN1, LOW);
35     digitalWrite(motorIN2, HIGH);
36     analogWrite(motorPWM, HALF_POWER);
37 }
38 void ledIdle() {
39     analogWrite(ledRed, 0);
40     analogWrite(ledGreen, 255);
41     analogWrite(ledBlue, 0);
42 }
43 void ledActive() {
44     analogWrite(ledRed, 240);
45     analogWrite(ledGreen, 240);
46     analogWrite(ledBlue, 240);
47 }
```

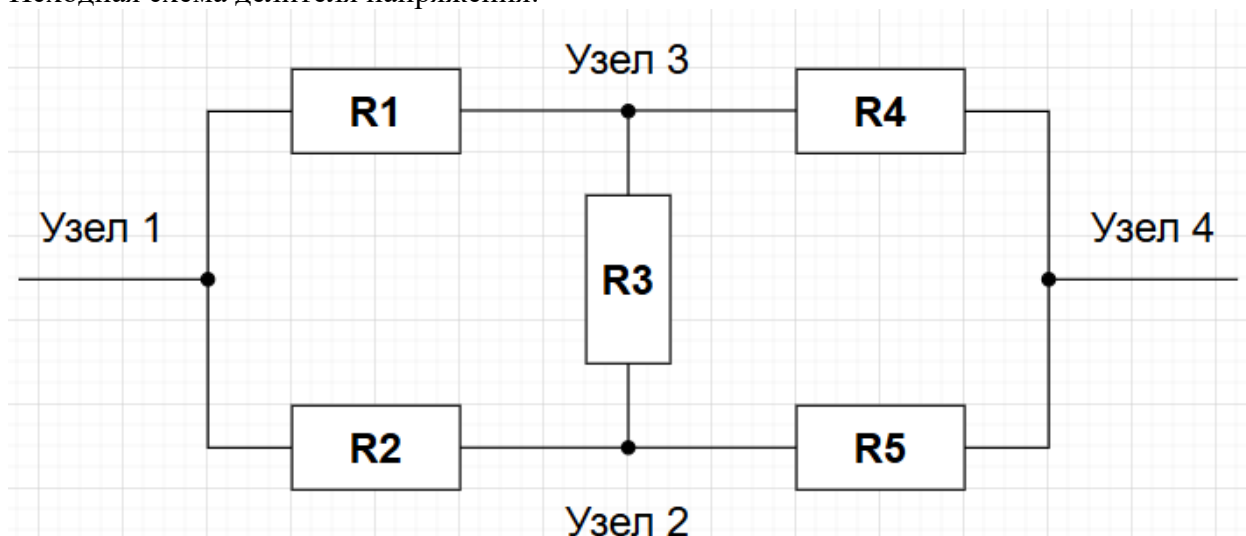
Рисунок 3а. Программный код, загруженный в память Arduino Uno.

```
48 void setup() {
49   pinMode(ledRed, OUTPUT);
50   pinMode(ledGreen, OUTPUT);
51   pinMode(ledBlue, OUTPUT);
52   pinMode(motorPWM, OUTPUT);
53   pinMode(motorIN1, OUTPUT);
54   pinMode(motorIN2, OUTPUT);
55   motorStop();
56   ledIdle();
57 }
58
59 void loop() {
60   unsigned long now = millis();
61   int fsrValue = analogRead(fsrPin);
62   switch (state) {
63     case IDLE:
64       ledIdle();
65       if (fsrValue > PRESS_THRESHOLD) {
66         ledActive();
67         lastPressTime = now;
68         motorOpen();
69         state = OPENING;
70         stateTimer = now;
71       }
72       break;
73     case OPENING:
74       if (now - stateTimer >= MOTOR_TIME) {
75         motorStop();
76         state = WAIT_BEFORE_CLOSING;
77         stateTimer = now;
78       }
79       break;
80     case WAIT_BEFORE_CLOSING:
81       if (fsrValue > PRESS_THRESHOLD) {
82         lastPressTime = now;
83       }
84       if (now - lastPressTime >= NO_PRESS_TIMEOUT) {
85         motorClose();
86         state = CLOSING;
87         stateTimer = now;
88       }
89       break;
90     case CLOSING:
91       if (now - stateTimer >= MOTOR_TIME) {
92         motorStop();
93         state = IDLE;
94       }
95       break;
96   }
97 }
```

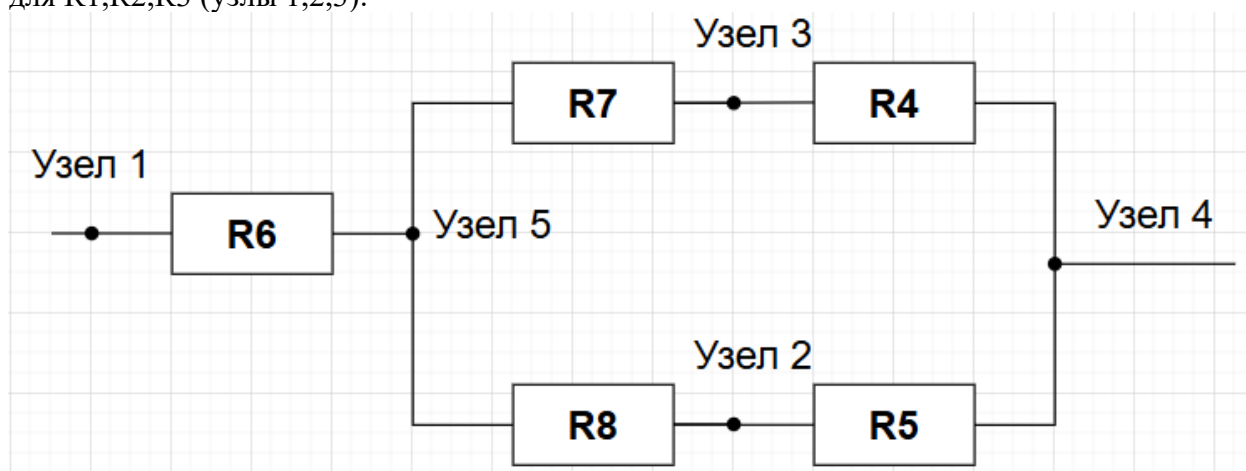
Рисунок 3в. Программный код, загруженный в память Arduino Uno.

Решение**Вопрос 1.**

Исходная схема делителя напряжения:



Необходимо использовать преобразование треугольник-звезда, чтобы привести схему к простым последовательным и параллельным соединениям. Произведем преобразования для R1, R2, R3 (узлы 1, 2, 3):



Необходимо произвести расчет сопротивлений R6, R7, R8:

$$R_6 = \frac{R_1 \times R_2}{R_1 + R_2 + R_3} = \frac{24 \times 6}{24 + 6 + 6} = 4 \text{ кОм}$$

$$R_7 = \frac{R_1 \times R_3}{R_1 + R_2 + R_3} = \frac{24 \times 6}{24 + 6 + 6} = 4 \text{ кОм}$$

$$R_8 = \frac{R_3 \times R_2}{R_1 + R_2 + R_3} = \frac{6 \times 6}{24 + 6 + 6} = 1 \text{ кОм}$$

Теперь запишем формулу эквивалентного сопротивления и получим из нее формулу для вычисления R5:

$$R_{\text{ЭКВ}} = R_6 + \frac{(R_4 + R_7)(R_5 + R_8)}{R_5 + R_6 + R_8 + R_4}$$

$$(R_{\text{ЭКВ}} - R_6)(R_5 + R_6 + R_8 + R_4) = (R_4 + R_7)(R_5 + R_8)$$

$$R_5(R_{\text{ЭКВ}} - R_6) + (R_{\text{ЭКВ}} - R_6)(R_6 + R_8 + R_4) = R_5R_4 + R_5R_7 + R_8R_4 + R_8R_7$$

$$R_5R_{\text{ЭКВ}} - R_5R_6 - R_5R_4 - R_5R_7 =$$

$$R_8R_4 + R_8R_7 - (R_{\text{ЭКВ}} - R_6)(R_6 + R_8 + R_4)$$

$$R_5 = \frac{R_8R_4 + R_8R_7 - (R_{\text{ЭКВ}} - R_6)(R_6 + R_8 + R_4)}{R_{\text{ЭКВ}} - R_6 - R_4 - R_7}$$

$$R_5 = \frac{R_8R_4 + R_8R_7 - (R_{\text{ЭКВ}} - R_6)(R_6 + R_8 + R_4)}{R_{\text{ЭКВ}} - R_6 - R_4 - R_7} = \frac{1 \cdot 2 + 1 \cdot 4 - (9,1 - 4)(4 + 1 + 2)}{9,1 - 4 - 2 - 4}$$

$$= \frac{6 - 5,1 \cdot 7}{-0,9} = 33 \text{ кОм}$$

Вопрос 2.

Основная логика алгоритма

Микроконтроллер постоянно отслеживает показания датчика давления (fsrValue). Система может находиться в одном из четырёх состояний:

1. **IDLE (Ожидание):** Светит зеленый индикатор, мотор стоит. Ждем нажатия.
2. **OPENING (Открытие):** Мотор работает на полную мощность в течение 6 секунд.
3. **WAIT_BEFORE_CLOSING (Пауза):** Мотор стоит, ждем 4 секунды отсутствия активности, чтобы начать закрытие. Если в этот момент снова нажать на датчик, таймер ожидания сбрасывается.
4. **CLOSING (Закрытие):** Мотор работает на половинной мощности в обратном направлении в течение 6 секунд, возвращая систему в начальное состояние.

Построчный разбор кода

Константы и конфигурация (строки 1–21)

- **1–7:** Определение пинов для датчика (A0), RGB-светодиода (A3-A5) и драйвера мотора (9-11).
- **8–10:** Пороги: PRESS_THRESHOLD — чувствительность датчика; HALF_POWER и FULL_POWER — значения ШИМ (PWM) для скорости мотора.
- **11–12:** Временные интервалы в миллисекундах: 6 секунд на работу мотора и 4 секунды задержки перед закрытием.
- **13–18:** Объявление перечисления enum MotorState для удобного управления состояниями автомата.

- **19–21:** Глобальные переменные для хранения текущего состояния и меток времени (используются вместо функции `delay()` для асинхронной работы).

Вспомогательные функции (строки 23–47)

- **motorStop():** Снимает напряжение с мотора, останавливая его.
- **motorOpen():** Подает сигнал HIGH на один вход и LOW на другой (направление вращения), устанавливая максимальную скорость.
- **motorClose():** Меняет полярность (вращение назад) и устанавливает скорость на 50%.
- **ledIdle():** Включает чистый зеленый цвет светодиода.
- **ledActive():** Включает белый цвет (смешение всех каналов R, G, B на высокой яркости).

Функция настройки `setup()` (строки 48–57)

- Конфигурирует пины как выходы (OUTPUT), переводит систему в режим ожидания и выключает мотор.

Главный цикл `loop()` (строки 59–97)

Здесь реализована логика переключений через оператор `switch (state)`:

- **IDLE (64–72):** Если значение с датчика превышает порог, фиксируем время нажатия, включаем белый свет, запускаем мотор на открытие и переходим в состояние OPENING.
- **OPENING (73–79):** Проверяем, прошло ли 6 секунд (MOTOR_TIME). Если да — останавливаем мотор и переходим в фазу ожидания перед закрытием.
- **WAIT_BEFORE_CLOSING (80–89):** * Если датчик нажат, мы обновляем `lastPressTime`, отодвигая момент закрытия.
 - Если с момента последнего нажатия прошло более 4 секунд (NO_PRESS_TIMEOUT), запускаем мотор на закрытие.
- **CLOSING (90–96):** Проверяем завершение 6-секундного цикла работы мотора, после чего возвращаем систему в IDLE.

Вопрос 3.

В программном коде прописаны следующие настройки для RGB-каналов:

1. Зеленый цвет (Режим ожидания - `ledIdle`):

- **R (Красный):** 0
- **G (Зеленый):** 255
- **B (Синий):** 0

- Этот цвет включается в функции `ledIdle()` и соответствует состоянию **IDLE** (ожидание).

2. Серый цвет (оттенок белого) (Активный режим - `ledActive`):

- **R (Красный):** 240
- **G (Зеленый):** 240
- **B (Синий):** 240
- Этот цвет включается в функции `ledActive()` при срабатывании датчика давления и переходе системы в состояния открытия, ожидания или закрытия.

Как формируется цвет в модели RGB

Цветовая модель **RGB** (Red, Green, Blue) основана на аддитивном смешивании трех основных цветов. Работа с ней в микроконтроллере строится на следующих принципах:

- **Смешивание света:** поскольку светодиод излучает свет, цвета «складываются». Если включить все три канала (красный, зеленый и синий) на высокую яркость, человеческий глаз воспримет результат как белый цвет.
- **Интенсивность (ШИМ):** В коде используется функция `analogWrite()`, которая управляет яркостью каждого светодиода по отдельности. Значения варьируются от **0** (светодиод полностью выключен) до **255** (максимальная яркость).
- **Генерация оттенков:** * Чтобы получить **чистый зеленый**, мы подаем максимум (255) на зеленый канал и 0 на остальные.
 - Чтобы получить **серый, близкий к белому**, мы подаем почти максимальные и равные значения (240) на все три канала одновременно.

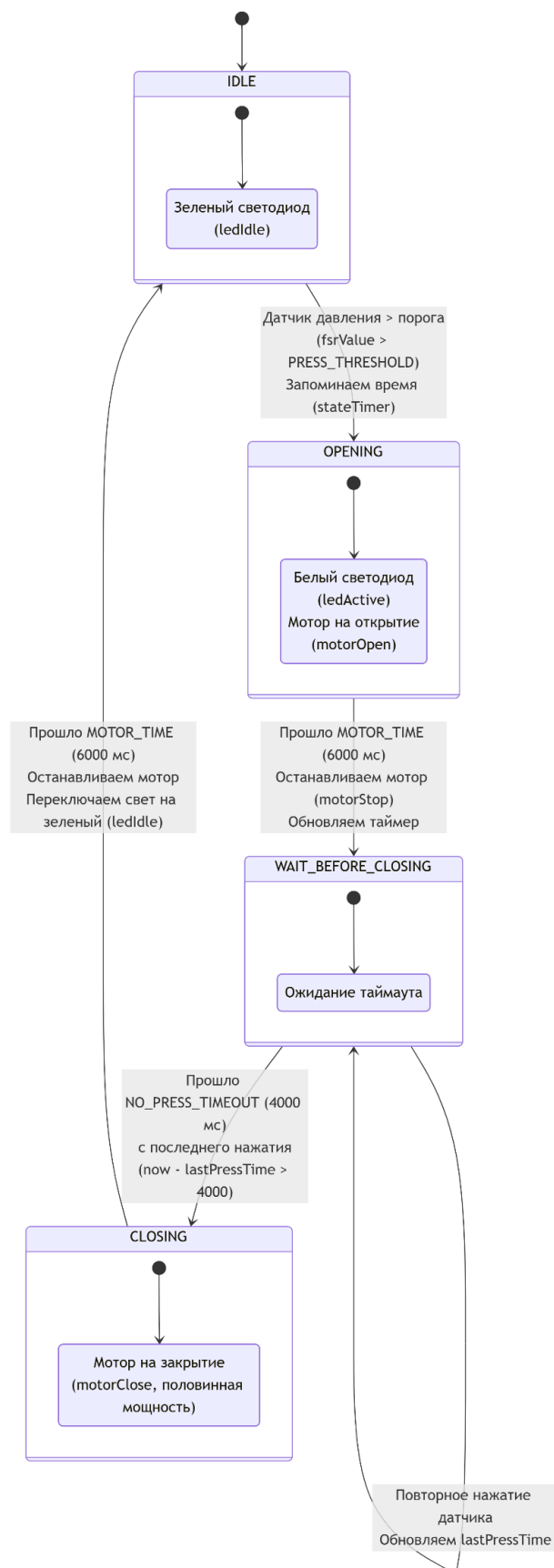
Вопрос 4.

Описание состояний и условий переходов

Ниже приведен детальный разбор логики переключений между состояниями, заложенной в функции `loop()`:

- **IDLE (Ожидание) -> OPENING (Открытие)**
 - **Условие:** Значение с датчика давления (`fsrValue`) превышает порог `PRESS_THRESHOLD` (100).
 - **Действие:** Запоминается текущее время в `stateTimer`, включается белый свет (`ledActive`), мотор запускается на полную мощность (`motorOpen`).
- **OPENING (Открытие) -> WAIT_BEFORE_CLOSING (Пауза)**

- **Условие:** Разница между текущим временем и временем начала открытия ($\text{now} - \text{stateTimer}$) достигает или превышает интервал `MOTOR_TIME` (6000 мс).
- **Действие:** Мотор полностью останавливается (`motorStop`), обновляется таймер состояния.
- **WAIT_BEFORE_CLOSING (Пауза) -> CLOSING (Закрытие)**
 - **Условие:** Прошло более `NO_PRESS_TIMEOUT` (4000 мс) с момента последнего зафиксированного нажатия на датчик ($\text{now} - \text{lastPressTime}$).
 - **Особенность:** Если во время нахождения в этом состоянии датчик нажимается повторно, переменная `lastPressTime` обновляется, и отсчет 4 секунд начинается заново.
 - **Действие:** Мотор запускается в обратном направлении на половинную мощность (`motorClose`).
- **CLOSING (Закрытие) -> IDLE (Ожидание)**
 - **Условие:** Мотор отработал положенное время `MOTOR_TIME` (6000 мс).
 - **Действие:** Мотор останавливается, светодиод переключается на зеленый цвет (`ledIdle`), система возвращается в исходное состояние готовности.



Вопрос 5.

В представленном программном коде за запуск процесса открытия отвечает константа `PRESS_THRESHOLD`:

- Значение константы: 100.
- Условие срабатывания в коде: `fsrValue > PRESS_THRESHOLD`.
- Это означает, что минимальное цифровое значение для начала открытия равно 101.

По условию калибровочная характеристика датчика является линейной. Это значит, что цифровое значение (`fsrValue`) растет прямо пропорционально приложенной силе (F). Нам известны опорные точки системы:

- Максимальная сила ($F_{\max}$) = 10 Н.
- Максимальное значение в программе (`fsrValue_max`) = 829.

Сначала определим, сколько цифровых единиц приходится на 1 Ньютон силы:
Коэффициент = $829 / 10 = 82,9$ единиц на 1 Ньютон.

Теперь разделим программный порог на полученный коэффициент: Минимальная сила = $101 / 82,9 = 1,2183$ Н.

Критерии оценки

Номер вопроса	Макс балл	Начисление баллов
1	7	Приведена верно перерисованная электрическая цепь: 2 балла. Правильно произведено преобразование треугольник-звезда, представлена правильная электрическая схема: 2 балла. Представлены формулы для пересчета сопротивлений треугольник-звезда: 1 балл. Приведена формула расчета эквивалентного сопротивления и из нее формулы для R_5 : 1 балл Расчеты верны, ответ получен верно: 1 балл
2	8	Верно описана процедура инициализации переменных и их назначение: 2 балла Верно описаны функции: 2 балла Верно описана основная логика <code>loop()</code> : 4 балла
3	3	Верно определены цвета (пурпурный (фиолетовый) и светло-серый (не белый!)): 1 балл Верно записаны коды в цветовой модели RGB: 1 балл Приведено верное объяснение формирования цвета в цветовой модели RGB: 1 балл.
4	4	Верно указаны все состояния системы: 2 балла Верно указаны переходы между состояниями и условия переходов: 2 балла
5	3	Приведена формула расчета силы, необходимой для открытия двери: 2 балла Расчет произведен верно, ответ получен верно: 1 балл