


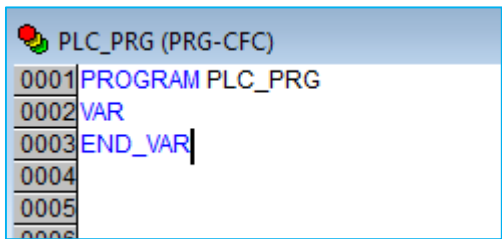
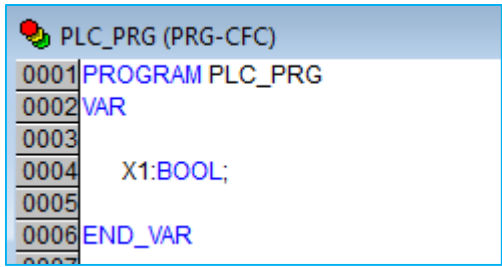
Вопросы для итоговой контрольной работы у студентов 3 курса по дисциплине:
«Программирование логических контроллеров»

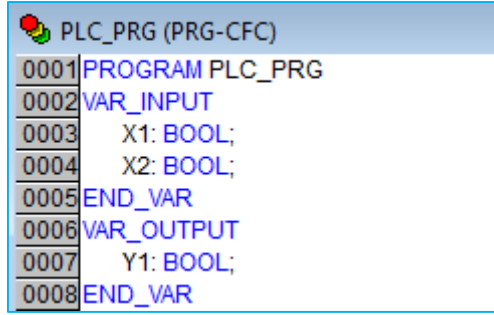
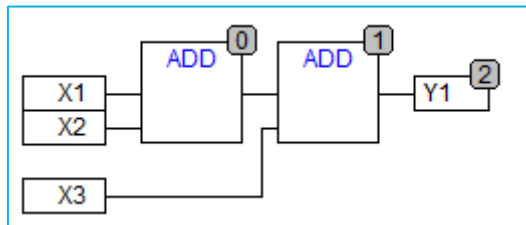
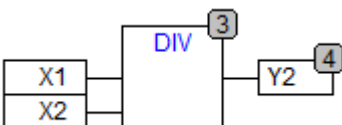
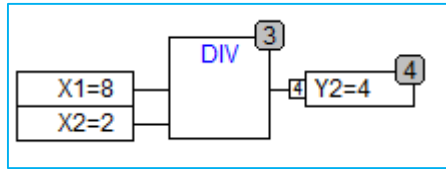
№	Вопросы	
1	Что такое интерфейс в аппаратах автоматики?	Интерфейс это – разъем для подключения в аппаратах автоматики. Например, USB, RJ45, DVI, HDMI, RS485 интерфейсы.
2	Что такое автоматизированная система управления (АСУ)?	Система, состоящая из комплекса средств автоматизации его деятельности и регламентов работы осуществляемые с участием человека (персонала).
3	Что такое система автоматического управления (САУ)?	Система, состоящая из комплекса средств автоматизации его деятельности и регламентов работы осуществляемые в автоматическом режиме без участия человека (персонала).
4	Что такое контроллер в системах управления?	Контроллером в системах управления называют техническое средство, выполняющее функции управления физическими процессами в соответствии с заложенным алгоритмом, с использованием информации, получаемой от датчиков и выводимой на окончательные устройства.
5	Что такое ПЛК?	Программируемый логический контроллер или ПЛК – это тип компьютера, который был адаптирован для управления производственными процессами. Его цель - контролировать ввод или ввод данных и принимать решения на основе пользовательской программы для контроля состояния устройства вывода.
6	Принцип работы ПЛК.	ПЛК работает по циклическому принципу. В самом начале цикла ПЛК сканирует состояния входов, на которые поступают сигналы от датчиков и устройств. Затем в соответствии с алгоритмом программы происходит вычисление состояния выходов. В конце рабочего цикла контроллер устанавливает каждый выход в состояние, которое было определено работой программы.
7	Почему в автоматике не используют ПК (персональный компьютер) вместо ПЛК?	ПК не имеют высокую надежность как ПЛК и по принципу действия не соответствуют для работы в автоматике. Для промышленных систем автоматики надежность является самым высоким приоритетом.
8	Модули ПЛК.	К модулям ПЛК относят - центральный модуль CPU (блок управления), сигнальные модули (модули ввода-вывода), модули блока питания, коммуникационные модули, технологические модули,
9	Для чего предназначен коммуникационный модуль ПЛК?	Коммуникационный модуль используется для подключения ПЛК к другим системам.
10	Какими бывают сигнальные входы и выходы ПЛК?	Сигнальные входы и выходы ПЛК бывают дискретными и аналоговыми.
11	Дискретные входы.	Дискретные входы – предназначены для ввода сигналов от дискретных датчиков (кнопки, тумблеры, концевые выключатели, термостаты и др.)
12	Дискретные выходы.	Дискретные выходы – предназначены для управления устройствами по принципу «включить/выключить» (магнитные пускатели, лампочки, клапаны и др.)

13	Аналоговые входы.	Аналоговые входы – предназначены для ввода непрерывного сигнала с датчиков и других устройств. Существует два основных вида унифицированных аналоговых сигналов: по току – 4 ... 20 мА, по напряжению 0 ... 10 В. Например, датчик температуры имеет диапазон -10 — +70 °С, тогда 4 мА на выходе соответствует -10 °С, а 20 мА – это +70 °С.															
14	Аналоговые выходы.	Аналоговые выходы – предназначены для плавного управления устройствами. Унифицированные значения аналогового сигнала на выходах такое же, как и на входах – 4 ... 20 мА (0 ... 10В). Например, клапан может поворачиваться в пределах от 0° до 90°. Ток 4 мА повернёт его в положение 0°, а 20 мА – в положение 90°. Для того, чтобы повернуть его на 45°, нужно подать на него управляющий сигнал 8 мА. Таким образом, меняя значение силы тока на выходе, контроллер может поворачивать клапан на заданный угол.															
15	Какой сигнал является непрерывным в автоматике?	В автоматике непрерывным является Аналоговый сигнал .															
16	Рабочий диапазон аналогового токового сигнала в автоматике.	Рабочий диапазон аналогового токового сигнала в автоматике это – от 4 мА до 20 мА (4 ... 20мА) .															
17	Какой сигнал отправляют датчик в блок управления, измеряющий непрерывное значение технического параметра?	Датчик, измеряющий непрерывное значение технического параметра, отправляет в блок управления аналоговый сигнал .															
18	Какой сигнал является прерывистым в автоматике?	В автоматике прерывистым является дискретный сигнал .															
19	Какой сигнал отправляют сигнальный-датчик в блок управления?	Сигнальный датчик в блок управления отправляет дискретный сигнал .															
20	Логический сигнал.	Логический сигнал – это сигнал, принимающий два возможных значения – «истина» или «ложь», «высокое» или «низкое» состояние, «0» или «1», «включить» или «отключить».															
21	Логический элемент «И» – конъюнкция, логическое умножение, AND.	«И» – логический элемент, выполняющий над входными данными операцию конъюнкции или логического умножения и имеющий минимум два входа и одного выхода.															
22	Составьте таблицу истинности логического элемента «И» (AND).	<div style="text-align: center;">  </div> <table border="1" style="margin: 10px auto;"> <thead> <tr> <th>Вход X1</th><th>Вход X2</th><th>Выход Y</th></tr> </thead> <tbody> <tr> <td>0</td><td>0</td><td>0</td></tr> <tr> <td>1</td><td>0</td><td>0</td></tr> <tr> <td>0</td><td>1</td><td>0</td></tr> <tr> <td>1</td><td>1</td><td>1</td></tr> </tbody> </table> <p>Таблица истинности для элемента «И» показывает, что на выходе элемента будет логическая единица лишь в том случае, если</p>	Вход X1	Вход X2	Выход Y	0	0	0	1	0	0	0	1	0	1	1	1
Вход X1	Вход X2	Выход Y															
0	0	0															
1	0	0															
0	1	0															
1	1	1															

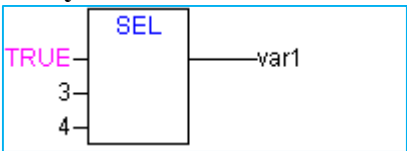
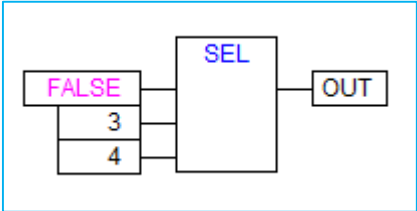
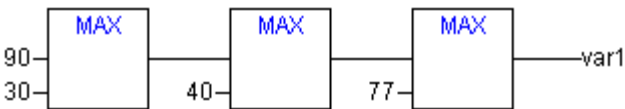
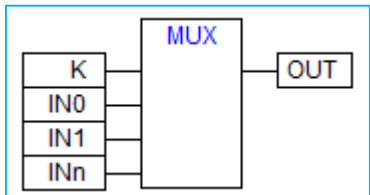
		логические единицы будут одновременно на первом входе «И» на втором входе. В остальных трех возможных случаях на выходе будет ноль.															
23	Логический элемент «ИЛИ» – дизъюнкция, логическое сложение, OR.	«ИЛИ» – логический элемент, выполняющий над входными данными операцию дизъюнкции или логического сложения.															
24	Составьте таблицу истинности логического элемента «ИЛИ» (OR).	 <table border="1" data-bbox="912 465 1455 651"> <thead> <tr> <th>Вход X1</th><th>Вход X2</th><th>Выход Y</th></tr> </thead> <tbody> <tr> <td>0</td><td>0</td><td>0</td></tr> <tr> <td>1</td><td>0</td><td>1</td></tr> <tr> <td>0</td><td>1</td><td>1</td></tr> <tr> <td>1</td><td>1</td><td>1</td></tr> </tbody> </table> <p>Таблица истинности для элемента «ИЛИ» показывает, что для появления на выходе логической единицы, достаточно чтобы логическая единица была на первом входе «ИЛИ» на втором входе. Если логические единицы будут сразу на двух входах, на выходе также будет единица.</p>	Вход X1	Вход X2	Выход Y	0	0	0	1	0	1	0	1	1	1	1	1
Вход X1	Вход X2	Выход Y															
0	0	0															
1	0	1															
0	1	1															
1	1	1															
25	Логический элемент «НЕ» - отрицание, инвертор, NOT	«НЕ» – логический элемент, выполняющий над входными данными операцию логического отрицания. Данный элемент, имеющий один выход и только один вход, называют еще инвертором, поскольку он на самом деле инвертирует (обращает) входной сигнал.															
26	Среда разработки CODESYS. V2.3	Среда разработки CODESYS – это инструментальный программный комплекс, предназначенный для промышленной автоматизации.															
27	Поддерживаемые языки реализации средой разработки CODESYS V2.3.	Язык реализации: <ul style="list-style-type: none"> • IL • LD • FBD • SFC • ST • CFC 															
28	Поддерживаемые типы POU (program organization unit).	Тип POU: <ul style="list-style-type: none"> • Программа • Функциональный блок • Функция 															
29	Стандарт МЭК 61131-3.	Стандарт МЭК 61131-3 обращен на унифицированные наборы языков промышленного программирования ПЛК.															
30	Что такое программа?	Программа (program) – это разработка, написание и тестирование программ пользователя.															
31	Языки промышленного программирования ПЛК.	Это: *Язык LD (LAD, Ladder, язык лестничных диаграмм)															

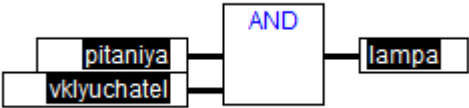
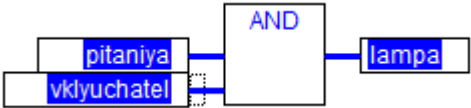
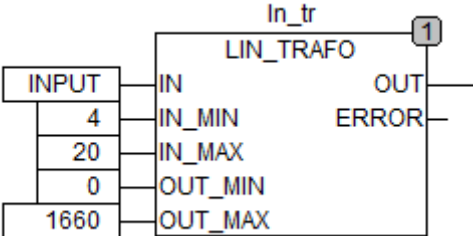
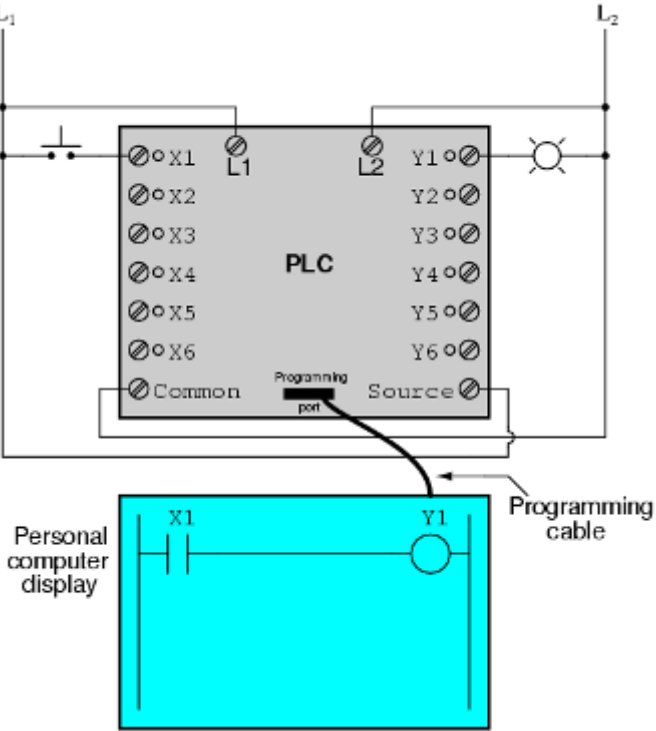
		<p>*Язык FBD (Functional Block Diagram, Диаграмма Функциональных Блоков) –</p> <p>*Язык последовательных функциональных схем SFC (Sequential Function Chart),</p> <p>*Язык ST (Structured Text, Структурированный Текст)</p> <p>*Язык CFC (Continuous Flow Chart)</p>
32	Язык LD (LAD, Ladder, язык лестничных диаграмм)	Язык LD (LAD, Ladder, язык лестничных диаграмм) – является графическим языком разработки, программа на котором представляет собой аналог релейной схемы.
33	Язык FBD (Functional Block Diagram, Диаграмма Функциональных Блоков)	Язык FBD (Functional Block Diagram, Диаграмма Функциональных Блоков) – является языком графического программирования, так же, как и LD, использующий аналогию с электрической (электронной) схемой. Программа на языке FBD представляет собой совокупность функциональных блоков (functional blocks, FBs), входа и выхода которых соединены линиями связи (connections).
34	Язык CFC (Continuous Flow Chart)	Язык CFC (Continuous Flow Chart) – еще один высокоуровневый язык визуального программирования. По сути, CFC – это дальнейшее развития языка FBD. Этот язык был специально создан для проектирования систем управления непрерывными технологическими процессами.
35	Язык ST (Structured Text, Структурированный Текст)	Язык ST (Structured Text, Структурированный Текст) представляет собой язык высокого уровня, имеющий черты языков Pascal и Basic.
36	Что такое программный компонент?	Программный компонент (program organization unit, POU) – это функция, функциональный блок, класс или программа.
37	Что такое комментарий в языках программирования?	Комментарий (comment) – это конструкция для включения текста, не влияющего на выполнение программы.
38	Как объявляются однострочные комментарии?	<p>Однострочные комментарии в строковых языках программирования объявляются с помощью символов:</p> <p>//...</p>
39	Приведите пример к объявленной однострочной комментария.	X:= 13; // комментарий для одной строки
40	Как объявляются многострочные комментарии?	<p>Многострочные комментарии в строковых языках программирования объявляются с помощью символов:</p> <p>(*...*) или /*...*/</p>
41	Приведите пример к объявленной многострочной комментария.	X:= 10; (* текст комментария *) /* комментарий на одной или более строк */

42	Что такое константа в языках программирования?	Константа (constant) – это элемент языка, указывающий на элемент данных с фиксированным значением.
43	Элементарные типы данных.	BOOL, INT, REAL, STRING и т.д.
44	Объявление входных переменных. VAR_INPUT ... END_VAR	VAR_INPUT IN: BOOL; T1: TIME; END_VAR
45	Объявление выходных переменных VAR_OUTPUT ... END_VAR	VAR_OUTPUT OUT: BOOL; ET_OFF: TIME; END_VAR
46	Определение статических переменных VAR ... END_VAR	VAR B: REAL; END_VAR
47	Где объявляются переменные?	<p>Переменные могут объявляться в различных текстовых конструкциях VAR ... END_VAR</p>  <p>Переменные могут быть объявлены либо как локальные в разделе определений POU, либо как глобальные - в списке глобальных переменных.</p>
48	Объявите переменную X1 с типом данных BOOL.	<pre>PROGRAM PLC_PRG VAR X1:BOOL; END_VAR</pre> 
49	Объявите входные переменные X1 и X2 и выходную переменную Y1 с типом данных BOOL.	<pre>PROGRAM PLC_PRG VAR_INPUT X1: BOOL; X2: BOOL; END_VAR VAR_OUTPUT Y1: BOOL; END_VAR</pre>

		
50	Стандартные операторы и функции МЭК.	<ul style="list-style-type: none"> •Арифметические операторы •Битовые операторы •Операторы выборки •Операторы сравнения •Математические функции
51	Приведите пример арифметического оператора сложение на языке реализации ST. (строчный язык)	<pre>VAR Y1: INT; END_VAR Y1:=1+2+3+4;</pre>
52	Приведите пример арифметического оператора сложение на языке реализации CFC. (блочный графический язык)	<pre>VAR Y1,X1,X2,X3: INT; END_VAR</pre> 
53	<p>В языке реализации CFC к чему будет равен значение переменного Y2 если X1:=8; и X2:=2; ?</p> 	<pre>X1:=8; X2:=2; Y2:=X1/X2; Y2=4</pre> 
54	Перечислите операторы выборки.	<p>Операторы выборки:</p> <pre>SEL MAX MIN LIMIT MUX</pre>
55	Оператор выборки SEL.	<pre>SEL</pre> <p>Бинарный выбор.</p> <p>OUT := SEL(G, IN0, IN1) означает:</p> <p>OUT := IN0 если G=FALSE;</p> <p>OUT := IN1 если G=TRUE.</p>

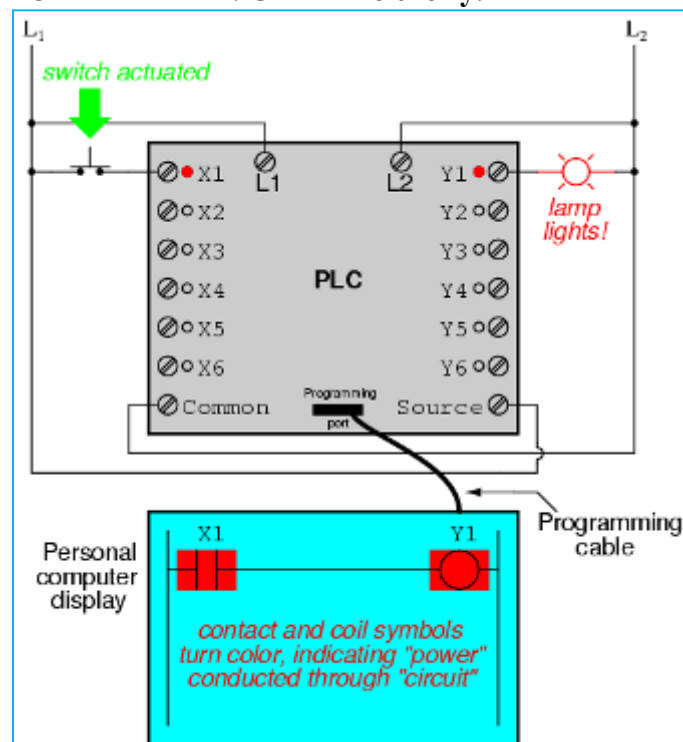
продолжение

56	<p>К чему будет равен значение переменного var1, если к функциональному входу поступает сигнал логическое «TRUE» ?</p> 	<p>Значение переменного var1 будет равен var1=4.</p>
57	<p>К чему будет равен значение переменного var1, если к функциональному входу поступает сигнал логическое «FALSE» ?</p> 	<p>Значение переменного var1 будет равен OUT=3.</p>
58	<p>Оператор выборки MAX.</p>	<p>MAX</p> <p>Функция максимум возвращает наибольшее из двух значений</p> <p>OUT := MAX(IN0, IN1)</p>
59	<p>К чему будет равен значение переменного var1?</p> 	<p>Значение переменного var1 будет равен var1=90.</p>
60	<p>Оператор выборки LIMIT.</p>	<p>LIMIT</p> <p>Ограничитель</p> <p>OUT := LIMIT(Min, IN, Max).</p> <p>Max задает верхнюю и Min нижнюю границы ограничителя. Если IN больше верхнего или меньше нижнего пределов, результат “обрезается” соответственно до Max или Min.</p>
61	<p>Приведите пример работы оператора выборки LIMIT.</p>	<p>Var1:=LIMIT(30,90,80); (* Результат 80 *);</p>
62	<p>Оператор выборки MUX.</p>	<p>MUX</p> <p>Мультиплексор. Возвращает K-е значение из входных переменных.</p> <p>OUT := MUX(K, IN0,...,INn) означает:</p> <p>OUT := INK.</p> 

63	<p>В каком случае загорится лампа?</p> 	<p>Лампа загорится только в том случае если pitaniya и vklyuchatel будут «TRUE»</p> 
64	<p>Масштабирование (работа преобразователей). Какое значение примет выходное переменное OUT если во входное переменное INPUT поступает токовый сигнал 12 мА ?</p> 	<p>Если во входное переменное INPUT поступает токовый сигнал 12 мА, то выходное переменное OUT примет значение равное 830. Так как функциональным блоком LIN_TRAFO осуществляется масштабирование поступающего сигнала на вход INPUT.</p>
65	<p>Язык LADDER. Опишите схему.</p> 	<p>Когда кнопочный переключатель не задействован (находится в не нажатом состоянии), сигнал не посылается на вход X1. В соответствии с программой, которая показывает «открытый» вход X1, сигнал не будет посылаться и на выход Y1. Таким образом, выход Y1 останется обесточенным, а индикатор, подключенный к нему, погасшим.</p>

66

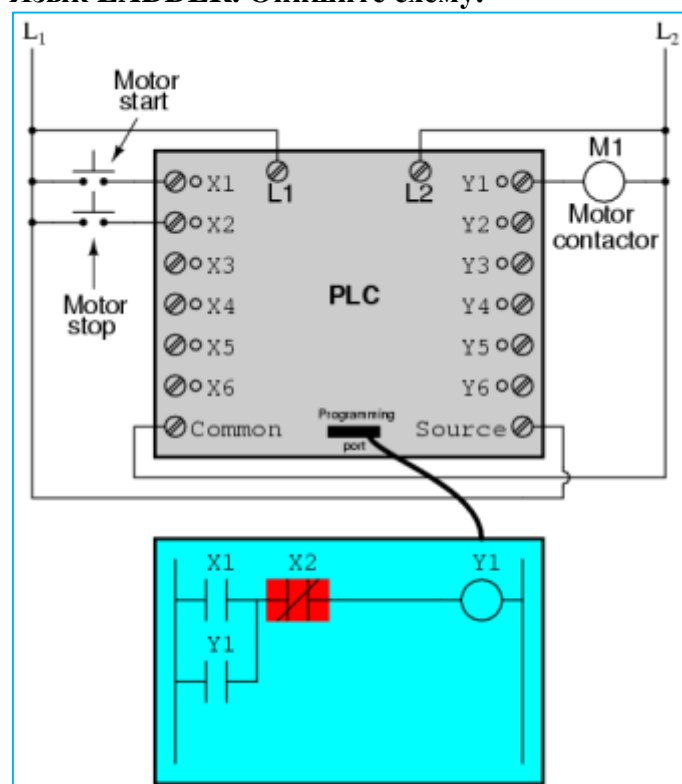
Язык LADDER. Опишите схему.



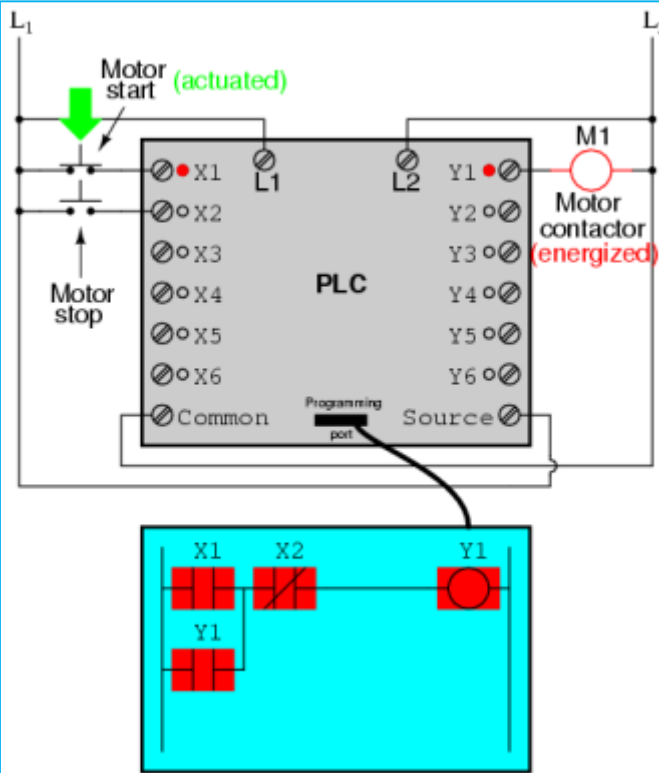
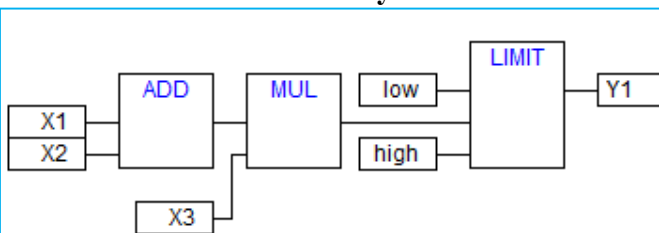
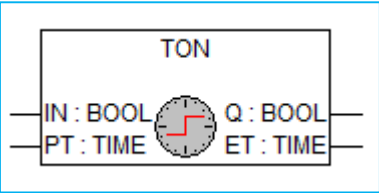
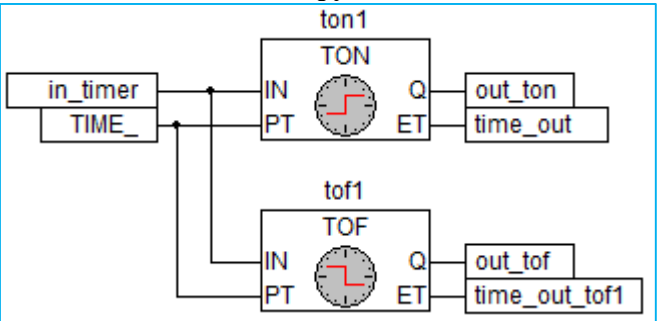
Если кнопочный переключатель нажат, сигнал будет отправлен к входу X1. Все контакты X1 в программе примут активированное состояние, как будто они являются контактами реле, активированными посредством подачи напряжения катушке реле, названной X1. В этом случае открытый контакт X1 будет «закрыт» и отправит сигнал к катушке Y1. Когда катушка Y1 будет находиться под напряжением, выход Y1 осветится лампочкой, подключенной к нему.

67

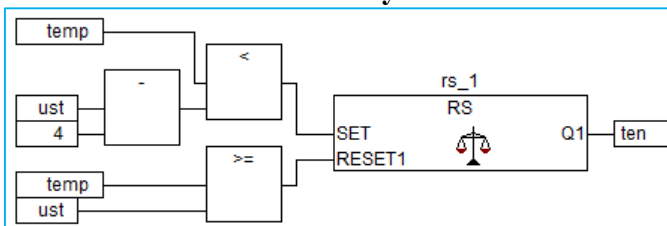
Язык LADDER. Опишите схему.



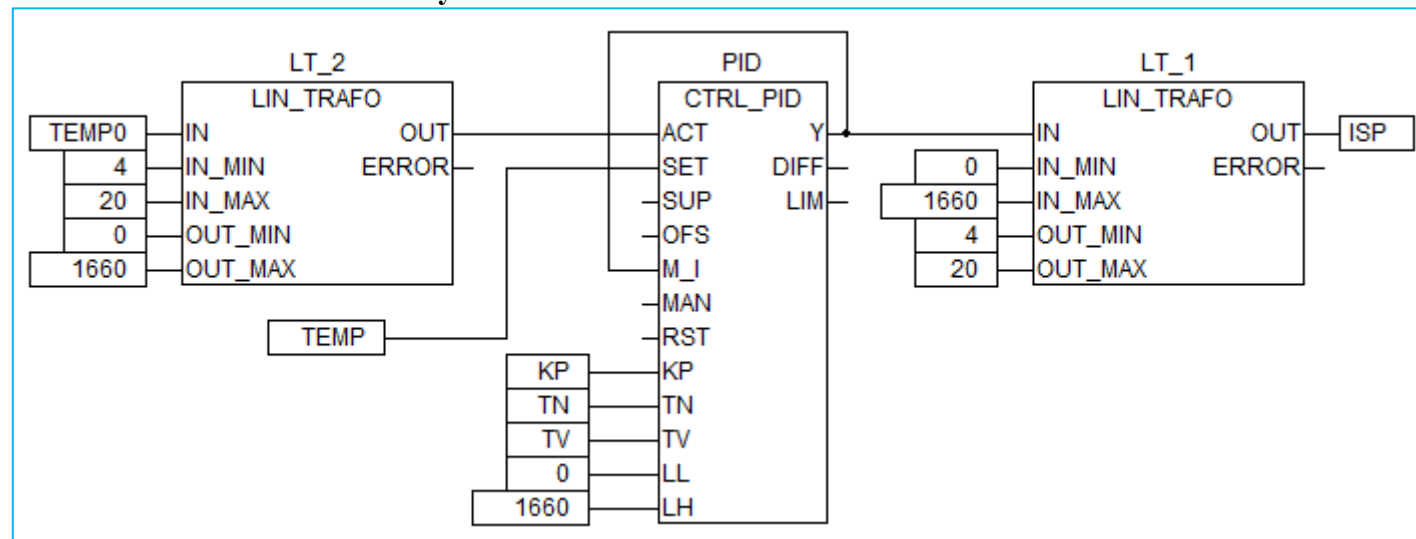
Переключатель, подключенный к входу X1, служит кнопкой «Старт», в то время как переключатель, подключенный к входу X2 - кнопкой «Стоп». Другой контакт, названный Y1, подобно печати в контакте, позволяет контактору двигателя оставаться под напряжением, даже если отпустить кнопку «Старт». При этом вы можете увидеть, как контакт X2, «закрытый» при нормальных условиях, появится в цветном блоке, показывая тем самым, что он находится в «закрытом» («электропроводящем») состоянии.

68	<p>Язык LADDER. Опишите схему.</p> 	<p>Если нажать кнопку «Старт», то по «закрытому» контакту X1 пройдет ток и он отправит 120 В переменного тока к контактору двигателя Y1. Параллельный контакт Y1 также «закроется», тем самым замкнув цепь.</p>
69	<p>Язык CFC. Опишите схему.</p> 	
70	<p>Язык CFC. Опишите функциональный блок.</p> 	<p>Функциональный блок TON «таймер с задержкой включения». Пока IN равен FALSE, выход Q = FALSE, выход ET = 0. Как только IN становится TRUE, начинается отсчет времени (в миллисекундах) на выходе ET до значения, равного PT. Далее счетчик не увеличивается. Q равен TRUE, когда IN равен TRUE и ET равен PT, иначе FALSE. Таким образом, выход Q устанавливается с задержкой PT от фронта входа IN.</p>
71	<p>Язык CFC. Опишите функциональный блок.</p> 	

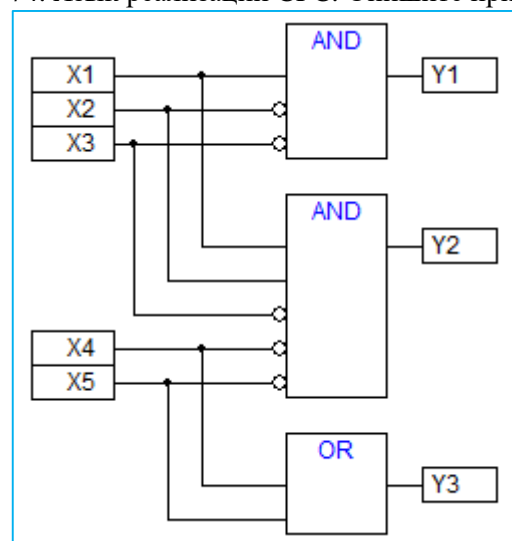
продолжение

72	<p>Язык CFC. Опишите схему.</p> 	<p>Двухпозиционный регулятор импульсное регулирование.</p>
----	--	---

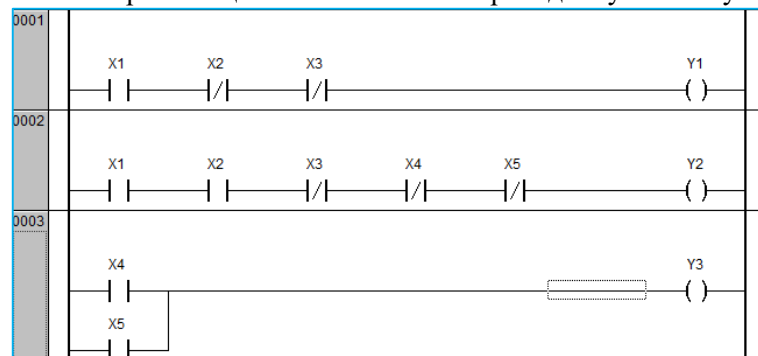
73. Язык CFC. Опишите схему.



74. Язык реализации CFC. Опишите приведенную логику.



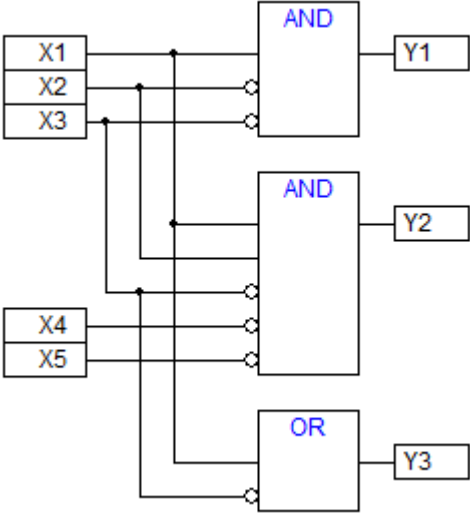
75. Язык реализации LAD. Опишите приведенную логику.



76. Язык реализации ST. Опишите приведенную логику.

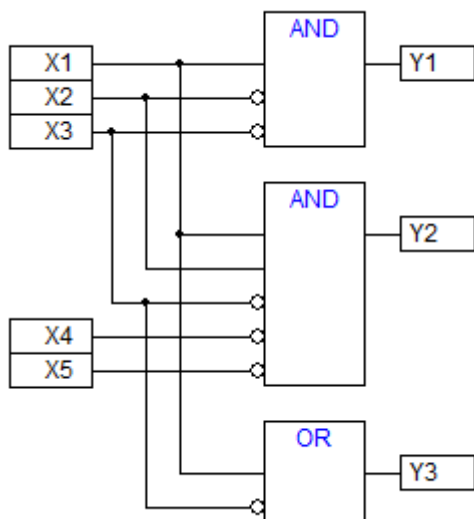
```
0001 IF X1 AND (NOT X2) AND (NOT X3)
0002 THEN
0003 Y1 := TRUE;
0004 ELSE
0005 Y1:=FALSE;
0006 END_IF
0007
0008 IF X1 AND X2 AND (NOT X3) AND (NOT X4) AND (NOT X5)
0009 THEN
0010 Y2:=TRUE;
0011 ELSE
0012 Y2:=FALSE;
0013 END_IF
0014
0015 IF X4 OR X5
0016 THEN
0017 Y3:=TRUE;
0018 ELSE
0019 Y3:=FALSE;
0020 END_IF
```

77. Язык реализации CFC, ST. Напишите ST реализацию логику разработанную в CFC.

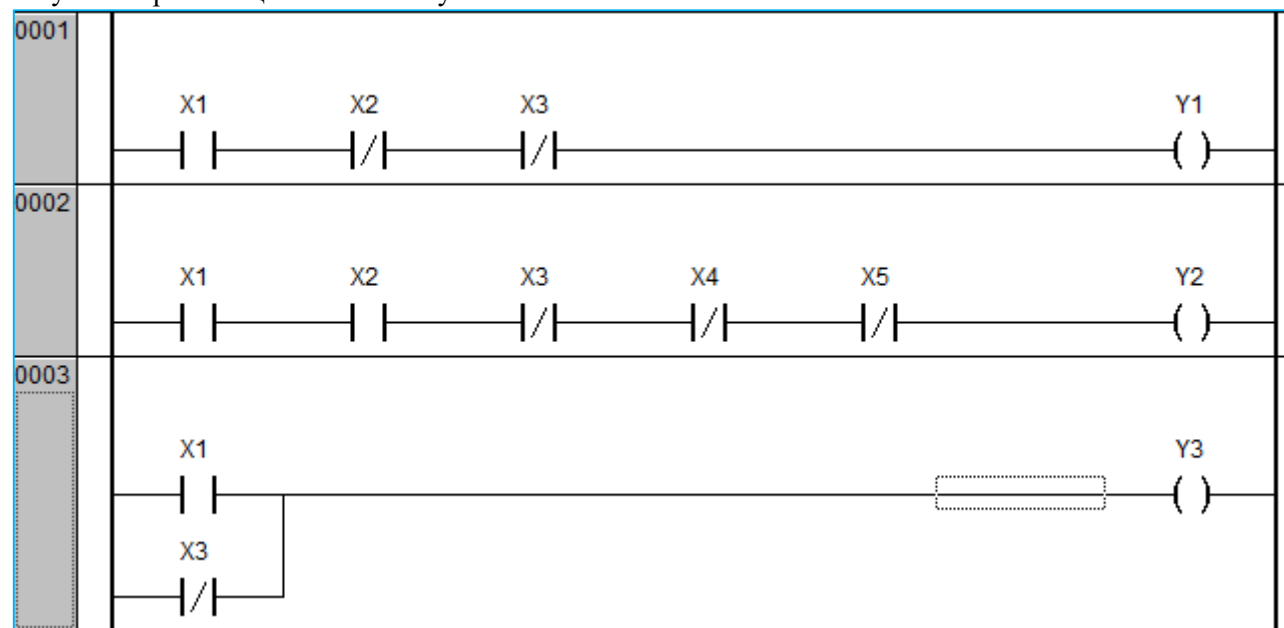
Реализация логики в CFC	Результата реализации CFC логику на языке ST
	<pre>0001 IF X1 AND (NOT X2) AND (NOT X3) 0002 THEN 0003 Y1 := TRUE; 0004 ELSE 0005 Y1:=FALSE; 0006 END_IF 0007 0008 IF X1 AND X2 AND (NOT X3) AND (NOT X4) AND (NOT X5) 0009 THEN 0010 Y2:=TRUE; 0011 ELSE 0012 Y2:=FALSE; 0013 END_IF 0014 0015 IF X1 OR (NOT X3) 0016 THEN 0017 Y3:=TRUE; 0018 ELSE 0019 Y3:=FALSE; 0020 END_IF</pre>

78. Язык реализации CFC, LAD. Напишите LAD реализацию логику разработанную в CFC.

--	--



Результата реализации CFC логику на языке LAD

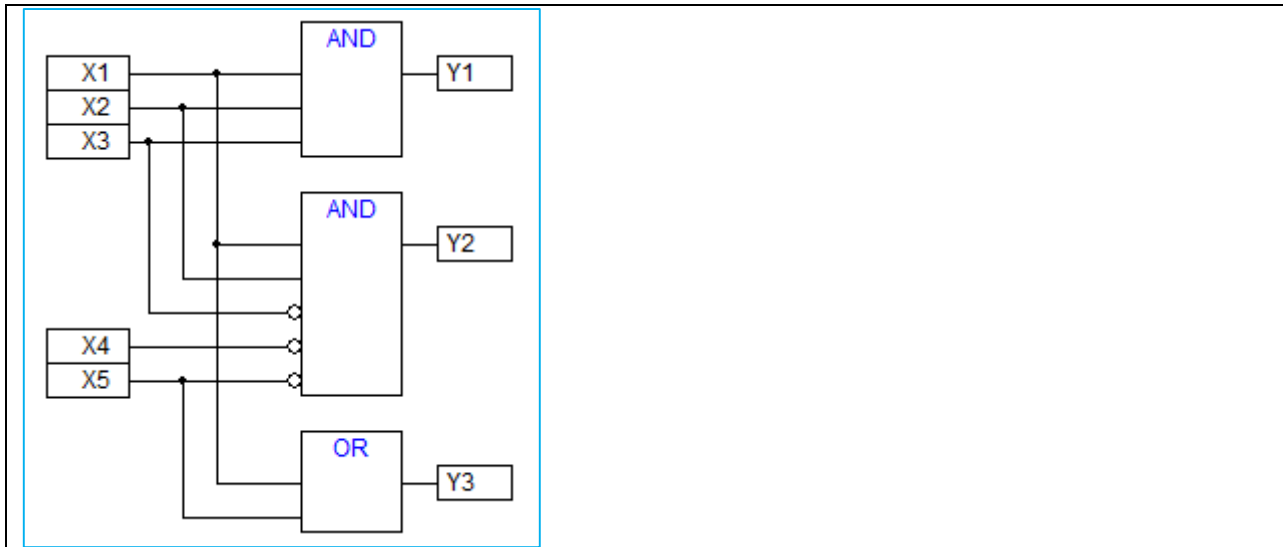


79. Язык реализации ST, CFC. Разработайте CFC реализацию логику написанную в ST.

```

0001 IF X1 AND X2 AND X3
0002 THEN
0003 Y1 := TRUE;
0004 ELSE
0005 Y1 := FALSE;
0006 END_IF
0007
0008 IF X1 AND X2 AND (NOT X3) AND (NOT X4) AND (NOT X5)
0009 THEN
0010 Y2 := TRUE;
0011 ELSE
0012 Y2 := FALSE;
0013 END_IF
0014
0015 IF X1 OR X5
0016 THEN
0017 Y3 := TRUE;
0018 ELSE
0019 Y3 := FALSE;
0020 END_IF
  
```

Результата реализации ST логику на языке CFC



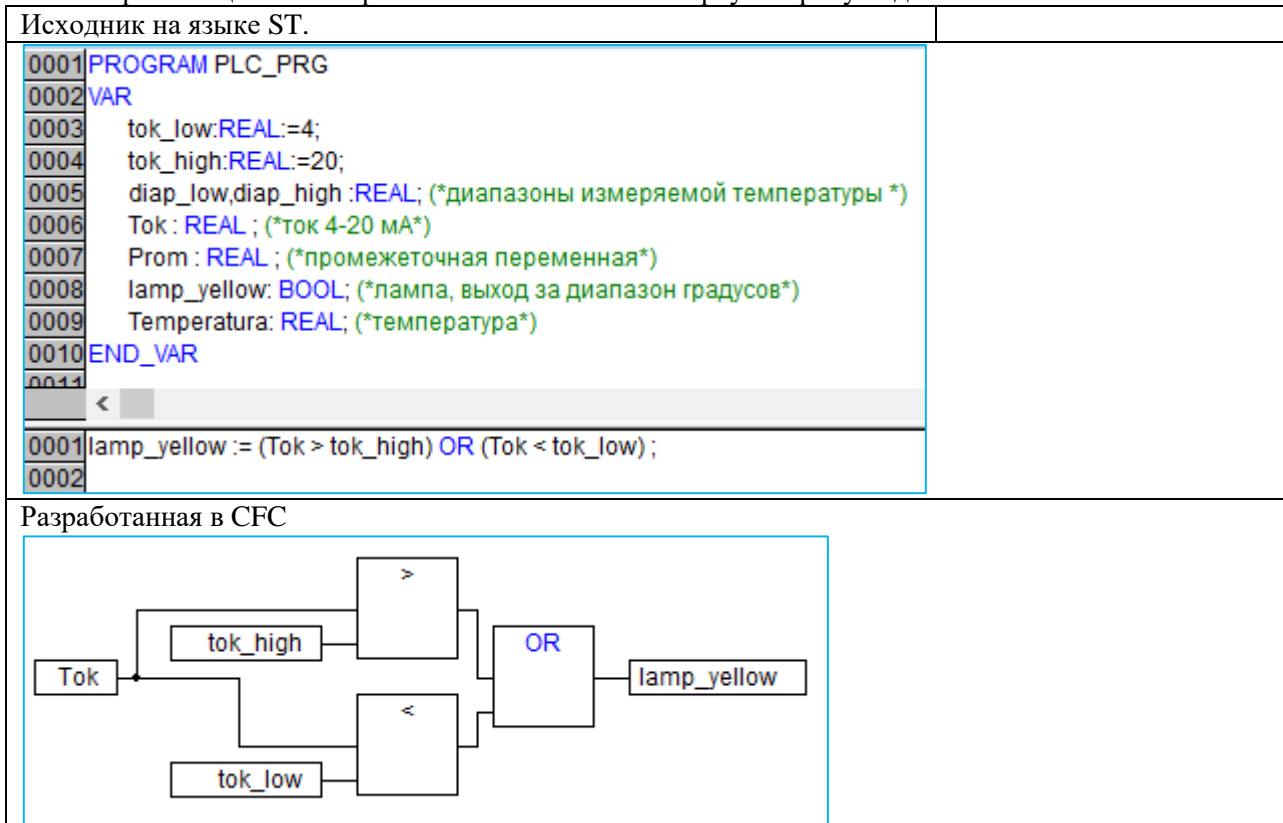
80. Язык реализации ST. Опишите написанную программу на языке ST для преобразователя.

```

0001 PROGRAM PLC_PRG
0002 VAR
0003   tok_low:REAL:=4;
0004   tok_high:REAL:=20;
0005   diap_low,diap_high :REAL; (*диапазоны измеряемой температуры *)
0006   Tok : REAL ; (*ток 4-20 мА*)
0007   Prom : REAL ; (*промежуточная переменная*)
0008   lamp_yellow: BOOL; (*лампа, выход за диапазон градусов*)
0009   Temperatura: REAL; (*температура*)
0010 END_VAR
0011
0001 lamp_yellow := (Tok > tok_high) OR (Tok < tok_low);
0002
0003 Prom:=LIMIT(tok_low, Tok, tok_high);
0004
0005 Temperatura:=((diap_high*(Prom-4))/16);

```

81. Язык реализации ST. Разработайте на языке CFC первую строку кода языка ST.



82.

0001	PROGRAM PLC_PRG	
0002	VAR	
0003	tok_low:REAL:=4;	
0004	tok_high:REAL:=20;	
0005	diap_low,diap_high:REAL; (*диапазоны измеряемой температуры *)	
0006	Tok: REAL; (*ток 4-20 мА*)	
0007	Prom: REAL; (*промежуточная переменная*)	
0008	lamp_yellow: BOOL; (*лампа, выход за диапазон градусов*)	
0009	Temperatura: REAL; (*температура*)	
0010	END_VAR	
0011	<	
0001	lamp_yellow := (Tok > tok_high) OR (Tok < tok_low);	
0002		
0003	Prom:=LIMIT(tok_low, Tok, tok_high);	
0004		
0005	Temperatura:=((diap_high*(Prom-4))/16);	
0006		

83.

0001	PROGRAM PLC_PRG	
0002	VAR	
0003	tok_low:REAL:=4;	
0004	tok_high:REAL:=20;	
0005	diap_low,diap_high:REAL; (*диапазоны измеряемой температуры *)	
0006	Tok: REAL; (*ток 4-20 мА*)	
0007	Prom: REAL; (*промежуточная переменная*)	
0008	lamp_yellow: BOOL; (*лампа, выход за диапазон градусов*)	
0009	Temperatura: REAL; (*температура*)	
0010	END_VAR	
0011	<	
0001	lamp_yellow := (Tok > tok_high) OR (Tok < tok_low);	
0002		
0003	Prom:=LIMIT(tok_low, Tok, tok_high);	
0004		
0005	Temperatura:=((diap_high*(Prom-4))/16);	
0006		

```

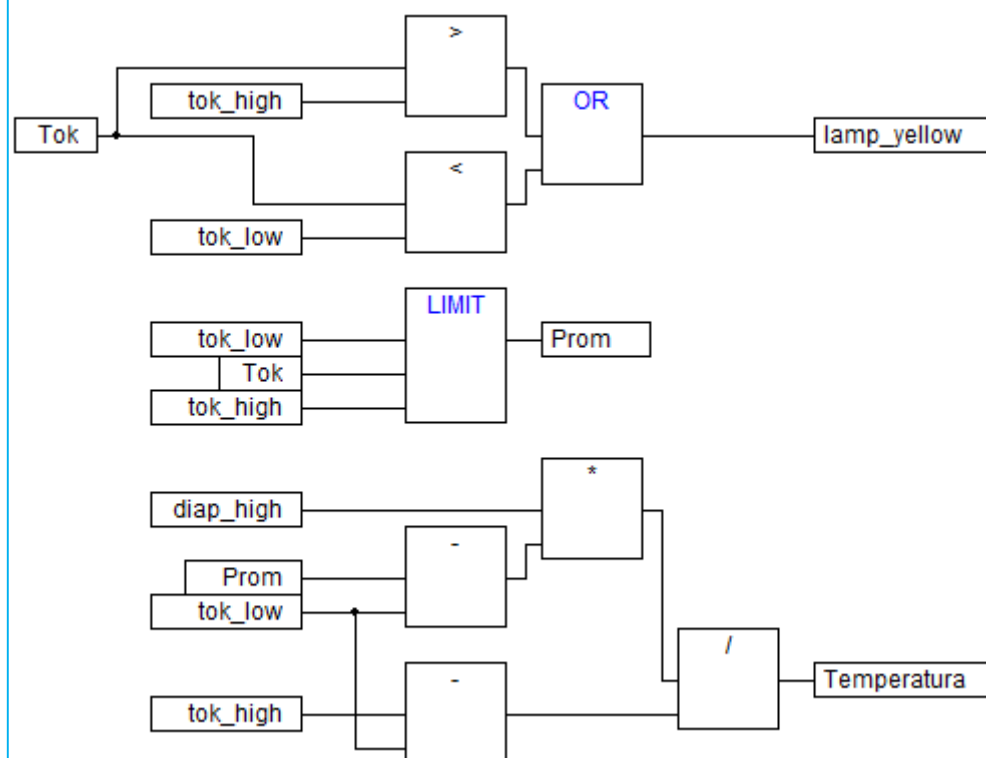
0001 PROGRAM PLC_PRG
0002 VAR
0003   tok_low:REAL:=4;
0004   tok_high:REAL:=20;
0005   diap_low,diap_high:REAL; (*диапазоны измеряемой температуры *)
0006   Tok: REAL; (*ток 4-20 мА*)
0007   Prom: REAL; (*промежуточная переменная*)
0008   lamp_yellow: BOOL; (*лампа, выход за диапазон градусов*)
0009   Temperatura: REAL; (*температура*)
0010 END_VAR

```

```

0001 lamp_yellow := (Tok > tok_high) OR (Tok < tok_low);
0002
0003 Prom:=LIMIT(tok_low, Tok, tok_high);
0004
0005 Temperatura:=((diap_high*(Prom-4))/16);
0006

```



85. Язык реализации ST. Опишите код.

```
0001(*первый этап*)
0002IF start=TRUE THEN
0003enter:=FALSE;
0004status:=0;
0005END_IF
0006
0007(*второй этап*)
0008CASE status OF
00090: IF s2 = TRUE THEN (*проверка-если кнопка №2 нажата, то status равен 1*)
0010status:=1;
0011ELSIF (s1 OR s3 OR s4 OR s5 OR s6 OR s7 OR s8 OR s9) = TRUE THEN (*если кнопки были нажаты ошибочно, то обнулить status*)
0012status:=0;
0013END_IF
00141: IF s9 = TRUE THEN
0015status:=2;
0016ELSIF (s1 OR s3 OR s4 OR s5 OR s6 OR s7 OR s8) = TRUE THEN
0017status:=0;
0018END_IF
00192: IF s1 = TRUE THEN
0020status:=3;
0021ELSIF (s1 OR s4 OR s5 OR s6 OR s7 OR s8) = TRUE THEN
0022status:=0;
0023END_IF
00243: IF s6 = TRUE THEN
0025status:=4;
0026ELSIF (s4 OR s5 OR s7 OR s8) = TRUE THEN
0027status:=0;
0028END_IF
00294: IF s7 = TRUE THEN
0030status:=5;
0031ELSIF (s4 OR s5 OR s8) = TRUE THEN
0032status:=0;
0033END_IF
0034END_CASE
0035
0036(*заключение-вывод сообщение "замок открыт")
0037IF status=5 THEN
0038enter:=TRUE;
0039mes:="замок открыт";
0040END_IF
0041
```

86. Внесите изменения в листинг кода программы «Кодовый замок», чтобы при закрытом состоянии замка в индикаторе выводилось сообщение: «Замок закрыт».

Ответ:

```
0001(*первый этап*)
0002IF start=TRUE THEN
0003enter:=FALSE;
0004status:=0;
0005END_IF
0006
0007(*второй этап*)
0008CASE status OF
00090: IF s2 = TRUE THEN (*проверка-если кнопка №2 нажата, то status равен 1*)
0010status:=1;
0011ELSIF (s1 OR s3 OR s4 OR s5 OR s6 OR s7 OR s8 OR s9) = TRUE THEN (*если кнопки были нажаты ошибочно, то обнулить status*)
0012status:=0;
0013END_IF
00141: IF s9 = TRUE THEN
0015status:=2;
0016ELSIF (s1 OR s3 OR s4 OR s5 OR s6 OR s7 OR s8) = TRUE THEN
0017status:=0;
0018END_IF
00192: IF s1 = TRUE THEN
0020status:=3;
0021ELSIF (s1 OR s4 OR s5 OR s6 OR s7 OR s8) = TRUE THEN
0022status:=0;
0023END_IF
00243: IF s6 = TRUE THEN
0025status:=4;
0026ELSIF (s4 OR s5 OR s7 OR s8) = TRUE THEN
0027status:=0;
0028END_IF
00294: IF s7 = TRUE THEN
0030status:=5;
0031ELSIF (s4 OR s5 OR s8) = TRUE THEN
0032status:=0;
0033END_IF
0034END_CASE
0035
```

87. Язык реализации ST. Опишите листинг.

```
0001(*первый этап*)
0002IF start=TRUE THEN
0003enter:=FALSE;
0004status:=0;
0005END_IF
```

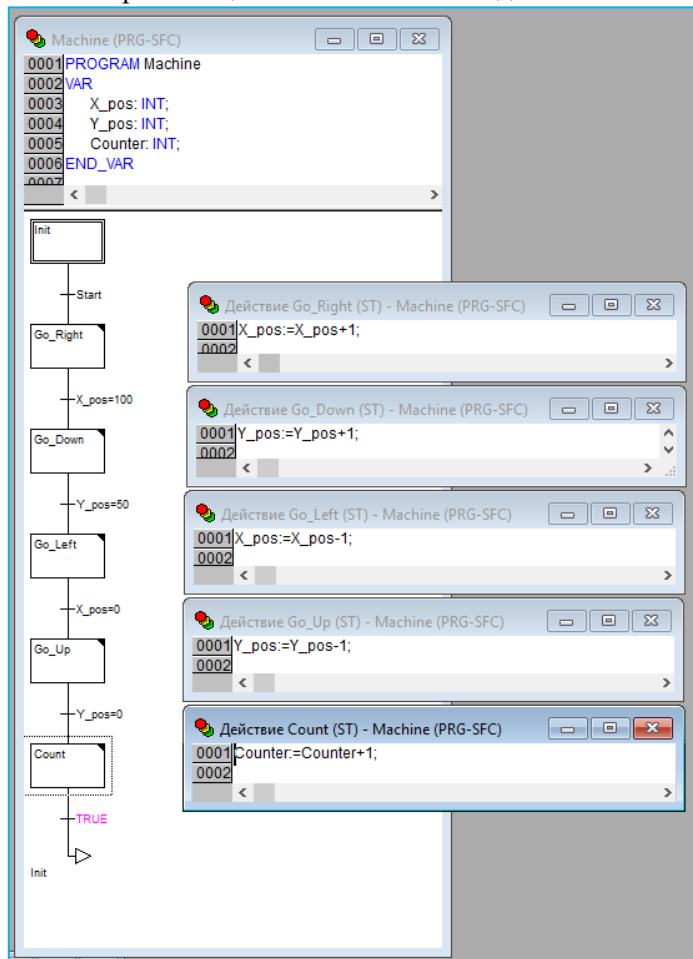
88. Язык реализации ST. Опишите листинг.

```
0006
0007 (*второй этап*)
0008 CASE status OF
0009 0: IF s2 = TRUE THEN (*проверка-если кнопка №2 нажата, то status равен 1*)
0010 status:=1;
0011 ELSIF (s1 OR s3 OR s4 OR s5 OR s6 OR s7 OR s8 OR s9) = TRUE THEN (*если кнопки были нажаты ошибочно, то обнулять status*)
0012 status:=0;
0013 END_IF
0014 1: IF s9 = TRUE THEN
0015 status:=2;
0016 ELSIF (s1 OR s3 OR s4 OR s5 OR s6 OR s7 OR s8) = TRUE THEN
0017 status:=0;
0018 END_IF
0019 2: IF s1 = TRUE THEN
0020 status:=3;
0021 ELSIF (s1 OR s4 OR s5 OR s6 OR s7 OR s8) = TRUE THEN
0022 status:=0;
0023 END_IF
0024 3: IF s6 = TRUE THEN
0025 status:=4;
0026 ELSIF (s4 OR s5 OR s7 OR s8) = TRUE THEN
0027 status:=0;
0028 END_IF
0029 4: IF s7 = TRUE THEN
0030 status:=5;
0031 ELSIF (s4 OR s5 OR s8) = TRUE THEN
0032 status:=0;
0033 END_IF
0034 END_CASE
0035
```

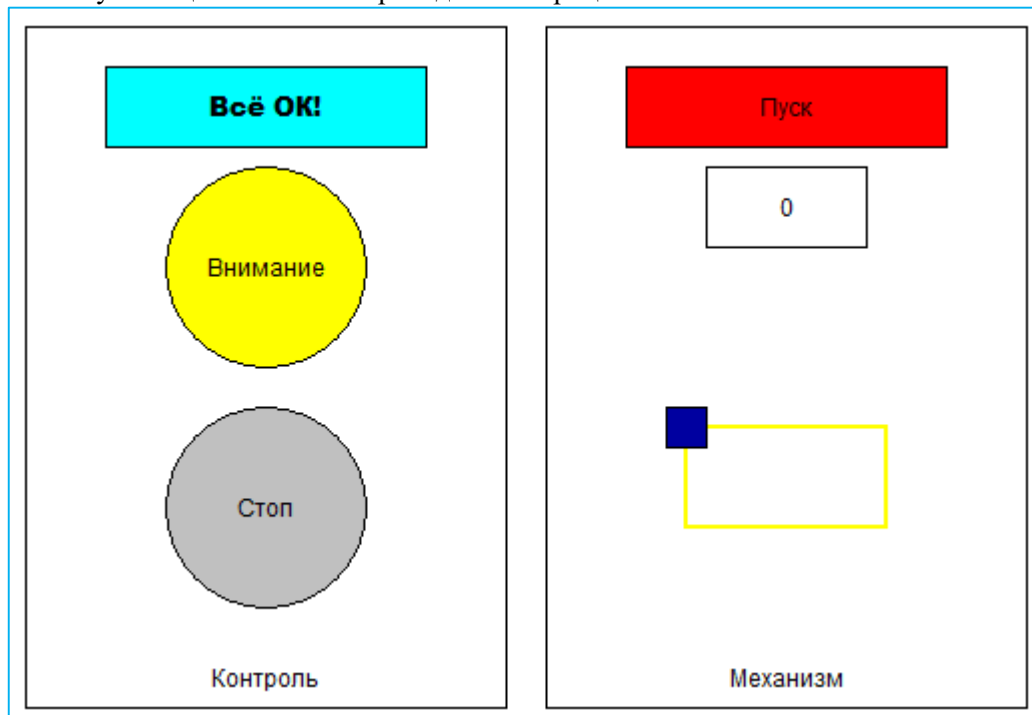
89. Язык реализации ST. Опишите листинг.

```
0035
0036 (*заключение-вывод сообщение "замок открыт"*)
0037 IF status=5 THEN
0038 enter:=TRUE;
0039 mes:='замок открыт';
0040 ELSE
0041 mes:='замок закрыт';
0042 END_IF
0043
```

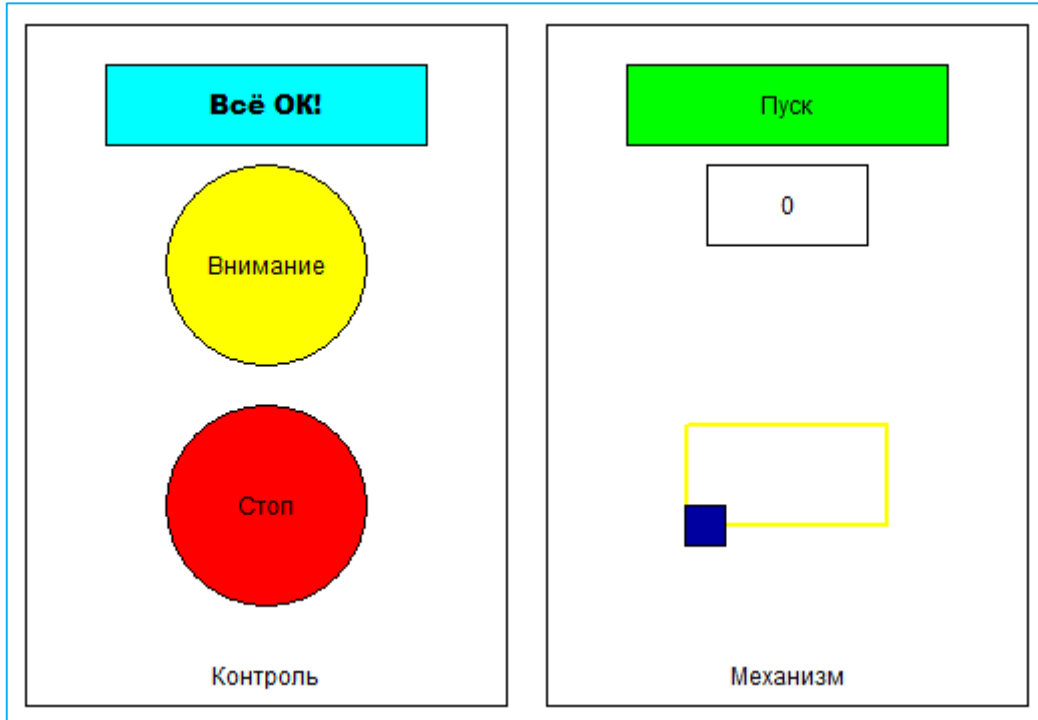
90. Язык реализации SFC. Опишите модель



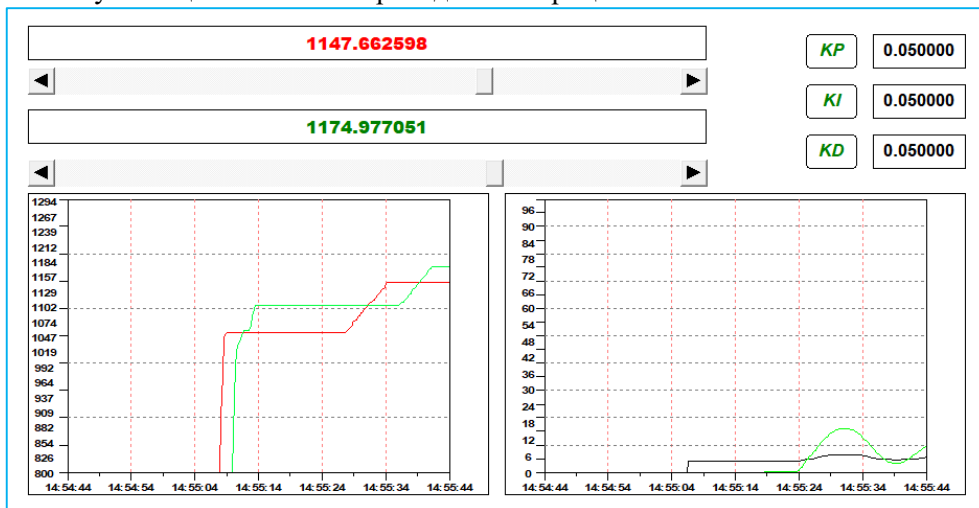
91. Визуализация. Опишите приведенный процесс



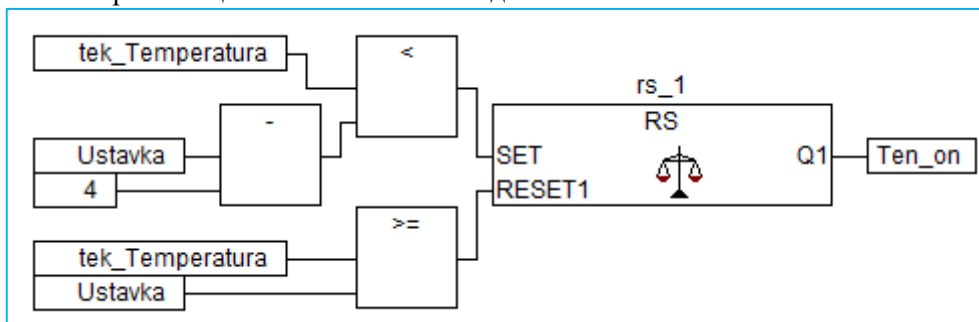
92. Визуализация. Опишите приведенный процесс



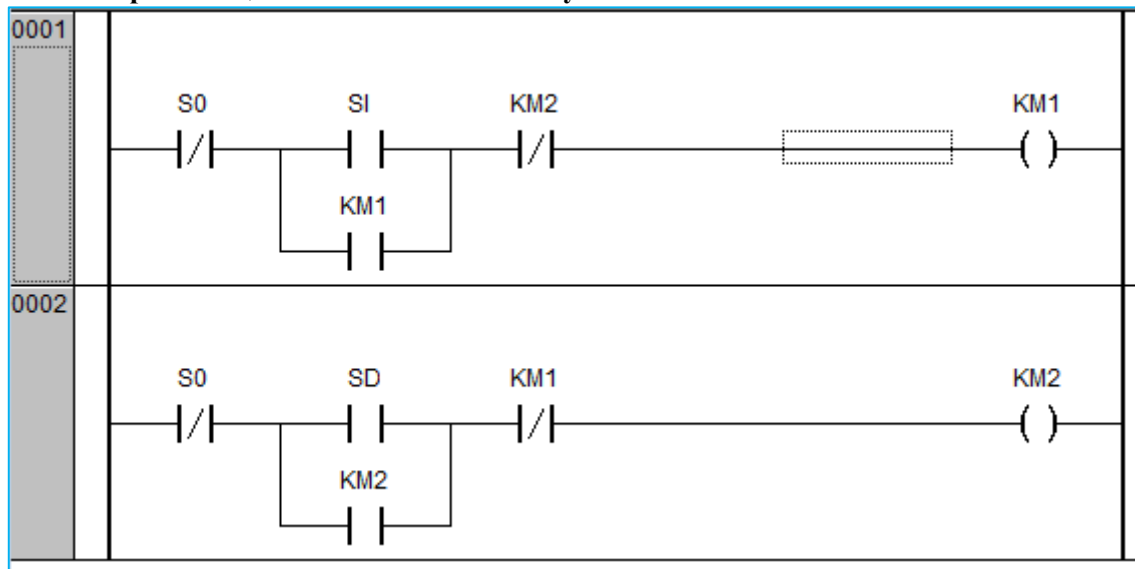
93. Визуализация. Опишите приведенный процесс



94. Язык реализации CFC. Опишите модель



95. Язык реализации LAD. Опишите схему



96. Язык реализации LAD. Опишите схему

