

Программирование
автоматики

В терминалах РЗ

АПВ. АВР. АЧР

elec.ru



Программирование автоматики В ПЛК – программируемые логические контроллеры

АПИ



Программирование автоматики В ПЛК – программируемые логические контроллеры АПВ, АВР, АЧР

Входные логические, аналоговые
сигналы



Выходные логические сигналы
– контакты реле («сухой
контакт»)

Программирование автоматики В терминалах РЗ, В ПЛК – программируемые логические контроллеры

Входные переменные

- Логические входы
- Функции защит:
 - Выходы с выдержкой времени
 - Выходы без выдержки времени
 - Другие выходы защит (фаза, направление...)

Редактор логических уравнений

- 31 локальная переменная
- Логические операции: НЕ, И, ИЛИ, Искл. ИЛИ
- Таймеры
- Триггеры и защелки
- Программируемые таймеры
- Максимум 100 операций
- Время цикла = 14мс

Выходные переменные

- 10 переменных, используемых с матрицей управления
- 4 стандартных переменных:
 - 3 переменные управления выключателем
 - 1 переменная управления осциллографом
- TS 177 - TS 186
- Входы защит (блокировка, квитирование)

Программирование
автоматики

В терминалах РЗ

АПВ, АВР, АУР

В 1993 г. Международная электротехническая комиссия выпустила в свет **стандарт МЭК 61131-3**. Этот международный стандарт входит в группу МЭК 61131 стандартов, которые охватывают различные аспекты использования ПЛК.

Декларируемые цели МЭК 61131-3

– **стандартизация существующих языков ПЛК**

IEC 65B/373/CD, Committee Draft – IEC 61131-3.

Programmable controllers. Part 3: Programming languages, 2nd Ed // International Electrotechnic Commission. 1998.

Международная электротехническая комиссия (МЭК)

(IEC – International Electrotechnical Commission ,
www.iec.ch)

основана в 1906

Основателем и первым президентом МЭК был известный английский физик **лорд Кельвин** (Уильям Томсон).

Основная цель МЭК, определенная его Уставом, – содействие международному сотрудничеству по **стандартизации в области электротехники,**

Программирование
автоматики

В терминалах РЗ

АПВ, АВР, АЧР
• *ST (Structured Text)*

– *текстовый высокоуровневый язык* общего назначения,

по синтаксису ориентированный на **Паскаль**.

В терминалах РЗ

АПВ, АВР, АЧР

Вычисление максимума из массива

```
VAR_CONSTANT
```

```
    Array_Sz:BYTE:=4;
```

```
END_VAR
```

```
VAR
```

```
    Iter:BYTE;
```

```
    arr:ARRAY [1..Array_Sz] of real:=3.2,4.2,1.4,7.8;
```

```
fnd_max:REAL:=-1.2E38;
```

```
END_VAR
```

```
FOR Iter:=1 TO Array_Sz
```

```
DO
```

```
    fnd_max:=MAX(fnd_max,arr[Iter]);
```

```
END_FOR
```


Программирование автоматики

В терминалах РЗ

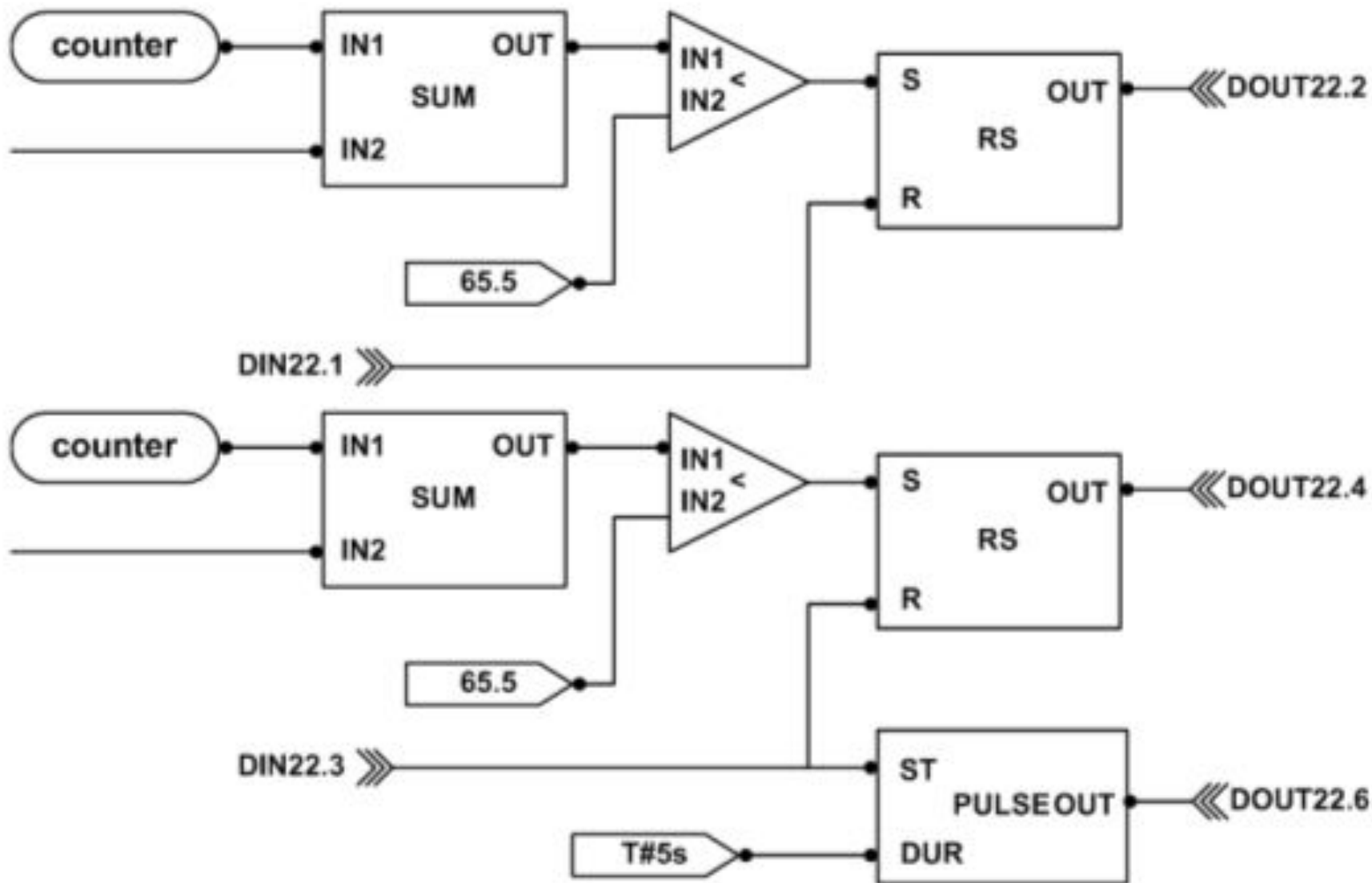
• **FBD (Functional Block Diagram)** – графический язык, по своей сути похожий на LD: вместо реле в этом языке используются функциональные блоки.

Алгоритм работы некоторого устройства, выраженный средствами этого языка, напоминает функциональную схему электронного устройства:

Элементы типа “логическое И”, “логическое ИЛИ” и т.п., соединенные линиями.

Программирование автоматики В терминалах РЗ

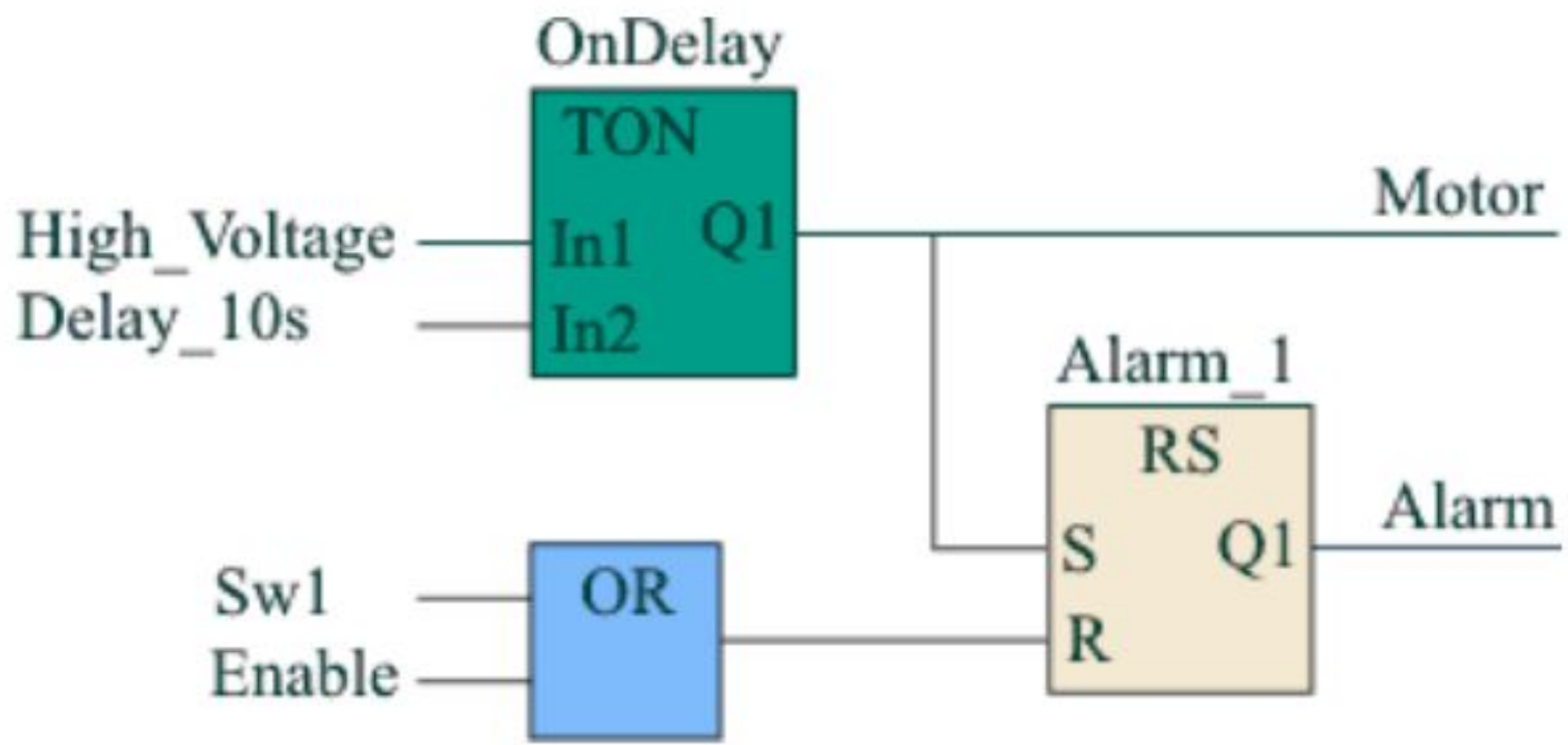
FBD (Functional Block Diagram)



Программирование
автоматики
В терминалах РЗ

FBD (Functional Block Diagram)

АЛВ, АВР, АУР
FBD обладает характерным для метафорических языков преимуществом: легкостью начального



Программирование автоматики

В терминалах РЗ

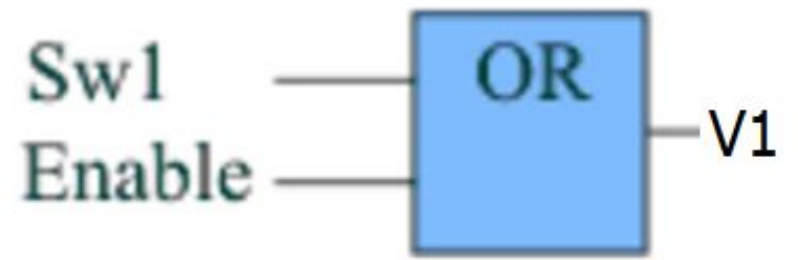
Логические уравнения:

АПВ АВР АЧР
Логические уравнения:

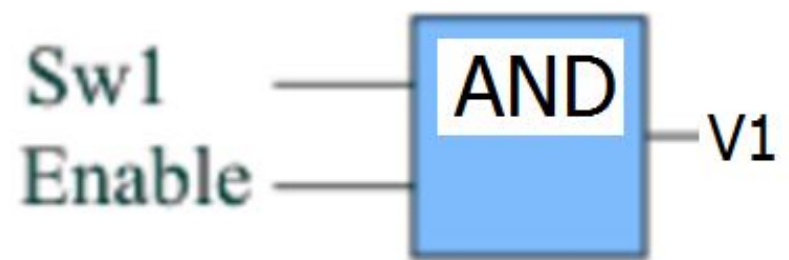
- *FBD (Functional Block Diagram)*, где каждый блок записывается в текстовом виде.

• Входные, промежуточные сигналы – справа от =

• Выходные сигналы – слева от =



$$V1 = Sw1 \text{ OR } Enable$$

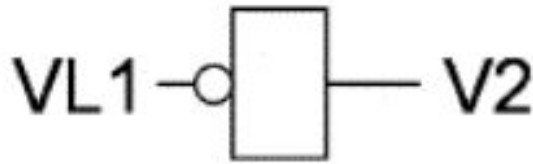


$$V1 = Sw1 \text{ AND } Enable$$

Логические уравнения:

НЕ: инверсия

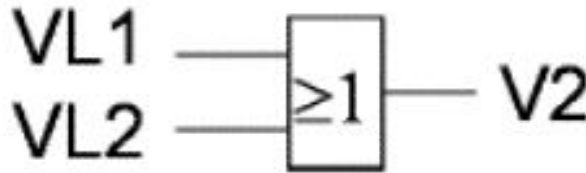
0 => 1
1 => 0



$V2 = \text{NOT } VL1$

ИЛИ: Логическое ИЛИ

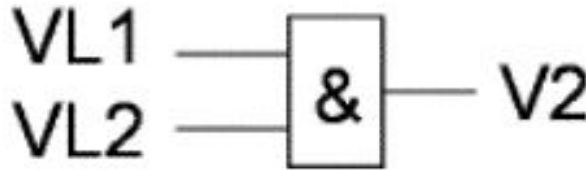
0, 0 => 0
0, 1 => 1
1, 0 => 1
1, 1 => 1



$V2 = VL1 \text{ OR } VL2$

И: Логическое И

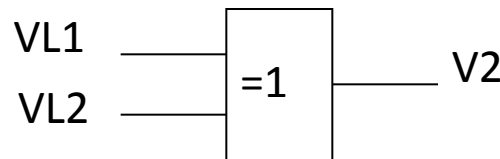
0, 0 => 0
0, 1 => 0
1, 0 => 0
1, 1 => 1



$V2 = VL1 \text{ AND } VL2$

Искл. ИЛИ: Исключающее ИЛИ. $V1 \text{ XOR } V2$ равно $(V1 \text{ AND } (\text{NOT } V2)) \text{ OR } (V2 \text{ AND } (\text{NOT } V1))$

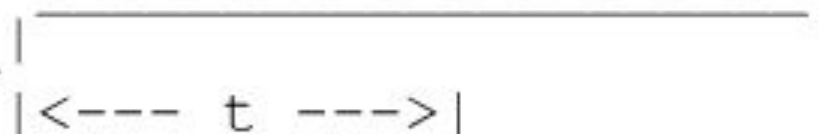
0, 0 => 0
0, 1 => 1
1, 0 => 1
1, 1 => 0



$V2 = VL1 \text{ XOR } VL2$

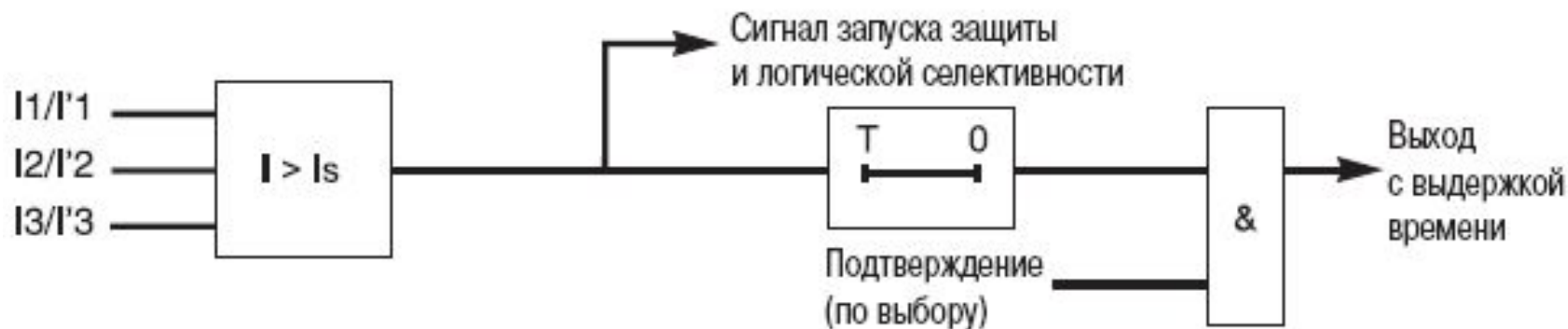
- $x = \text{TON}(y, t)$: таймер на срабатывание
(задержка выходного сигнала относительно переднего фронта входного сигнала) переменная x отслеживает переход из 0 в 1 переменной y с задержкой в t миллисекунд

I12



VL1

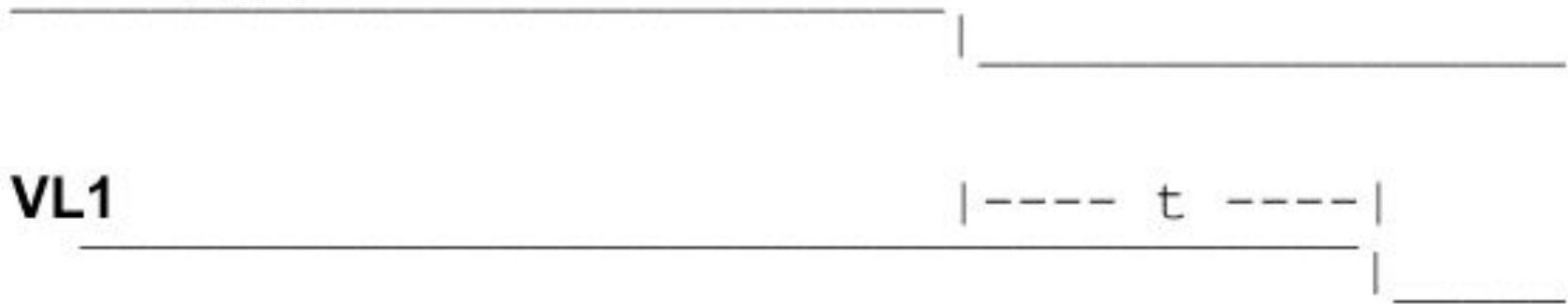


$$\text{VL1} = \text{TON}(\text{I12}, 300)$$


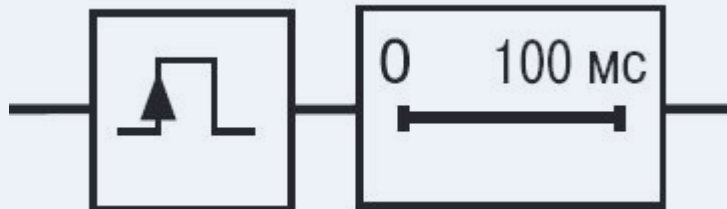
- $x = \text{TOF}(y, t)$: таймер на возврат
(задержка выходного сигнала относительно заднего фронта входного сигнала) переменная x отслеживает переход из 1 в 0 переменной y с задержкой в t миллисекунд.

P50/51_1_1

VL1



VL1 = TOF (P50/51_1_1 , 300)



Удлинение импульса на 100мс

Программирование автоматики

В терминалах РЗ

- ***LD (Ladder Diagram)*** – *графический язык, стандартизованный вариант класса языков релейно-контактных схем.*

- Входные сигналы – контакты реле,
- Элементы (таймеры, счетчики) – катушки реле (имеющие свои контакты),
- Промежуточные реле, имеющие свои контакты,
- Выходные сигналы – катушки реле.

Из-за своих ограниченных возможностей язык дополнен привнесенными средствами: таймерами, счетчиками и т.п.

Программирование
автоматики

В терминалах РЗ

LD (Ladder Diagram)

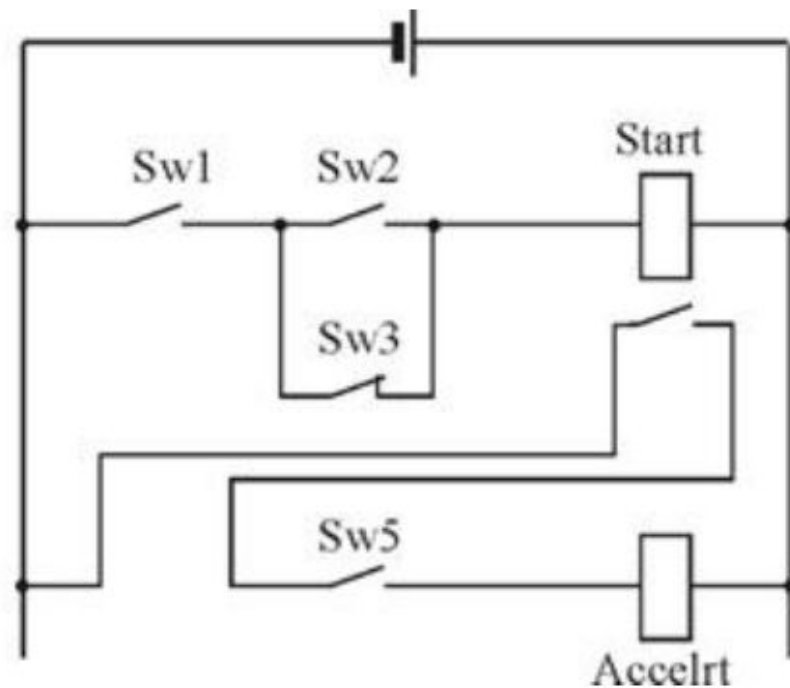
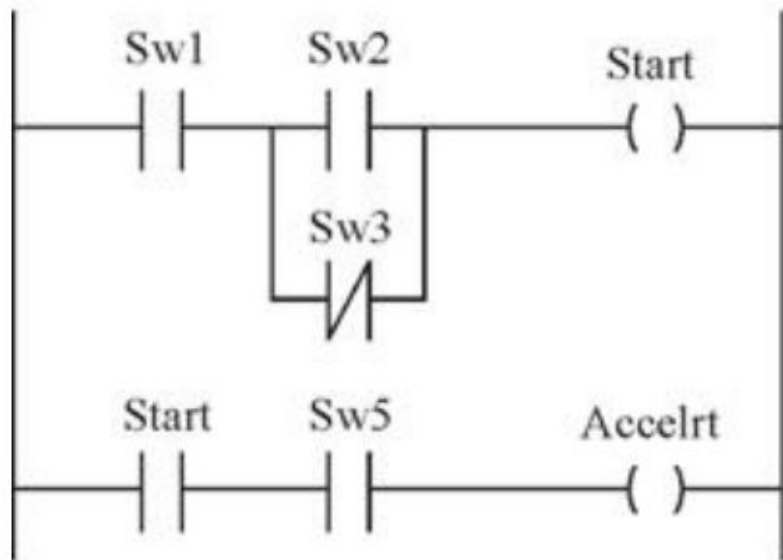
АПВ, АВР, АЦР

LD — метафора реле, лежащая в основе концепции, графическая форма описания алгоритма, позволяет легко освоить язык непрофессионалу.

При переходе на ПЛК язык обладал вполне объяснимыми преимуществами, т.к. снимал психологические проблемы переучивания персонала.

Программирование автоматики В терминалах РЗ

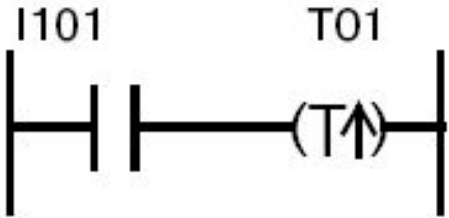
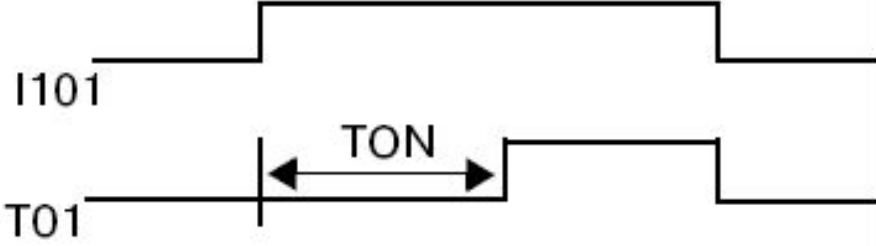
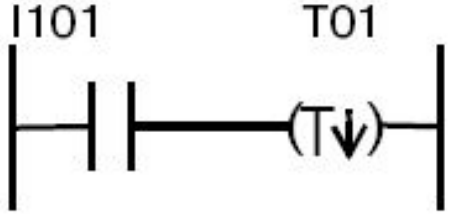

LD (Ladder Diagram)



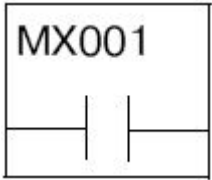
Пример программы на языке LD (слева) – слева питание, справа – земля.

и ее эквивалент в виде электрической цепи с реле и выключателями (справа)

LD (Ladder Diagram)

Программа Logigram	Эквивалентная временная диаграмма
 <p>A ladder diagram showing a normally open contact labeled I101 connected to a timer coil labeled T01 with a (T↑) symbol, indicating a start-on-delay timer.</p>	 <p>A timing diagram with two traces: I101 and T01. I101 transitions from low to high. T01 remains low until a delay period labeled TON has elapsed, then transitions to high. When I101 returns to low, T01 also returns to low.</p>
 <p>A ladder diagram showing a normally open contact labeled I101 connected to a timer coil labeled T01 with a (T↓) symbol, indicating a stop-on-delay timer.</p>	 <p>A timing diagram with two traces: I101 and T01. I101 transitions from low to high. T01 transitions to high immediately. When I101 returns to low, T01 remains high for a delay period labeled TOF before returning to low.</p>

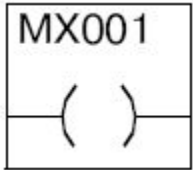
LD (Ladder Diagram)



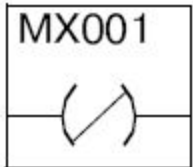
Контакт является замкнутым, когда управляющая им переменная находится в состоянии 1 (*нормально разомкнутый контакт*)



Контакт является замкнутым, когда управляющая им переменная находится в состоянии 0 (*нормально замкнутый контакт*)



Катушка «на срабатывание» - *реле срабатывает при подаче напряжения на катушку*



Катушка "на возврат" – *реле срабатывает при отсутствии напряжения на катушке*

Составить пример на данных языках

- *ST (Structured Text) (типа Паскаль)*
- *FBD (Functional Block Diagram) (И, ИЛИ, НЕ, ...)*
- **Логические уравнения**
- *LD (Ladder Diagram) – реле, контакты*

1) Выходной сигнал **V1** равен **1** при входных сигналах **I11** и **I12** равными **1**

2)) Выходной сигнал **V1** равен **1** через **1** секунду, после того, как и **I12** стал равными **1**