Logic Concepts

DR. TAREK A. TUTUNJI

Binary Concept

- In digital systems, two-state conditions can be thought of as signals that are present (activated, ON) or not present (not activated, OFF)
- This two-state concept can be the basis for making decisions; and since it is very adaptable to the binary number system, it is a fundamental building block for programmable controllers and digital computers.
- Binary 1 represents the presence of a signal (or the occurrence of some event), while binary 0 represents the absence of the signal (or the nonoccurrence of the event).
- In digital systems, these two states are actually represented by two distinct voltage levels, +V and 0V.
- Positive logic vs. negative logic.

Binary Concept

1 (+V)	0 (0V)	Example
Operating	Not operating	Limit switch
Ringing	Not ringing	Bell
On	Off	Light bulb
Blowing	Silent	Horn
Running	Stopped	Motor
Engaged	Disengaged	Clutch
Closed	Open	Valve

Table 3-1. Binary concept using positive logic.

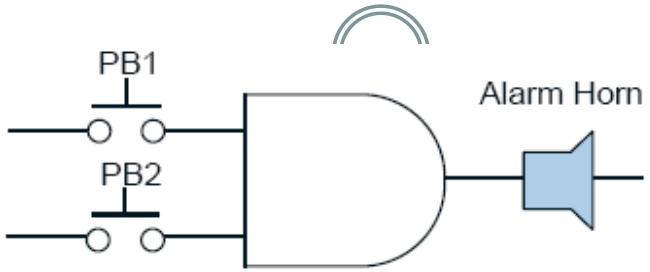
Logic Functions

- Operations performed by digital equipment, such as programmable controllers, are based on three fundamental logic functions—AND, OR, and NOT.

AND Truth Table		
Inputs		Output
A	B	Y
0	0	0
0	1	0
1	0	0
1	1	1

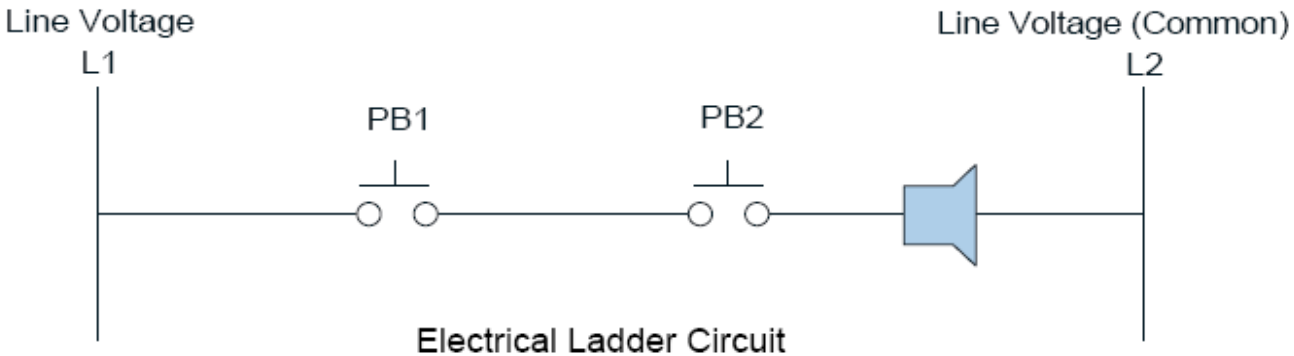
Figure 3-2. Two-input AND gate and its truth table.

AND Gate



Logic Representation

PB1	PB2	Alarm Horn
Not pushed (0)	Not pushed (0)	Silent (0)
Not pushed (0)	Pushed (1)	Silent (0)
Pushed (1)	Not pushed (0)	Silent (0)
Pushed (1)	Pushed (1)	Sounding (1)

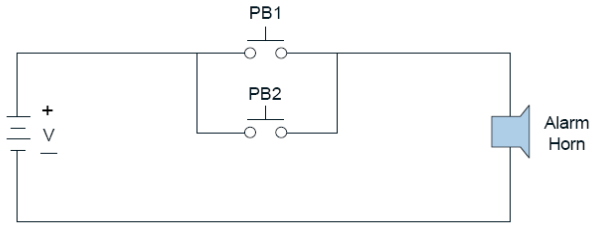


Electrical Ladder Circuit

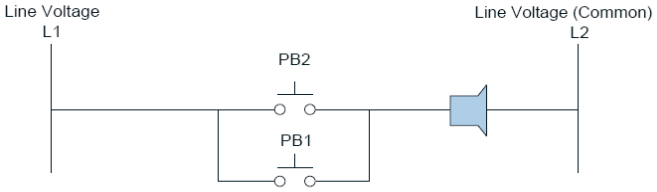
OR Gate

Logic Representation

PB1	PB2	Alarm Horn
Not pushed (0)	Not pushed (0)	Silent (0)
Not pushed (0)	Pushed (1)	Sounding (1)
Pushed (1)	Not pushed (0)	Sounding (1)
Pushed (1)	Pushed (1)	Sounding (1)



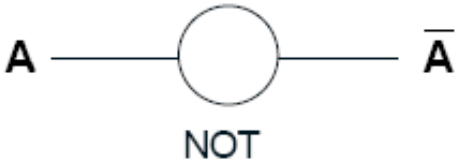
Electrical Circuit



Electrical Ladder Circuit

NOT Gate

Figure 3-5. Symbol for the NOT function.

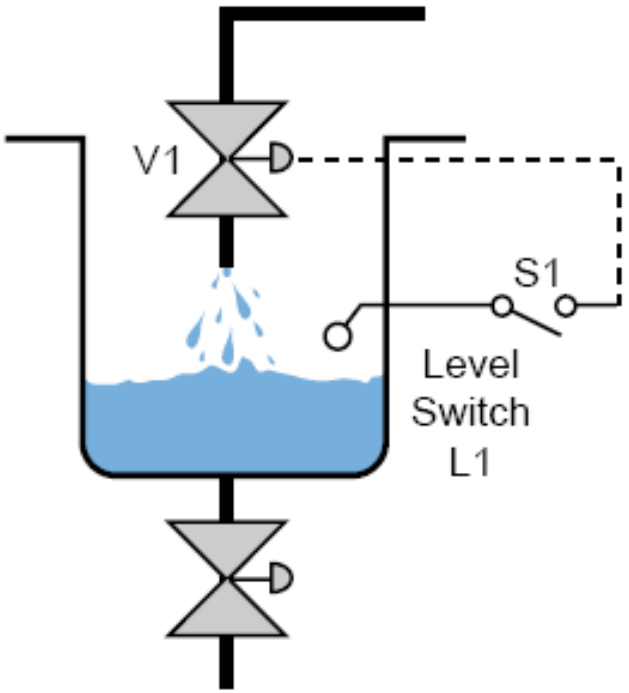


NOT Truth Table	
Input	**Output**
A	$\overline{A}$
0	1
1	0

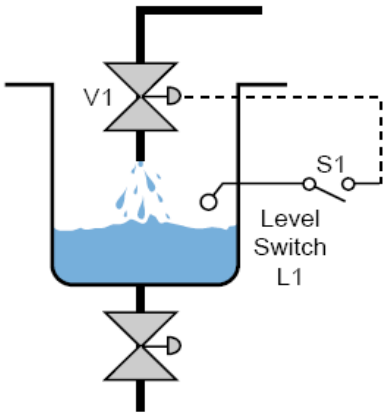
Figure 3-6. NOT gate and its truth table.

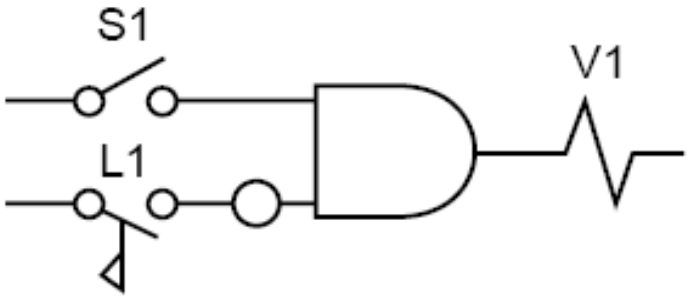
Logic Functions Example

- Show the logic gate, truth table, and circuit representation for a solenoid valve (V1) that will be open (ON) if selector switch S1 is ON and if level switch L1 is NOT ON (liquid has not reached level).



Logic Functions Example

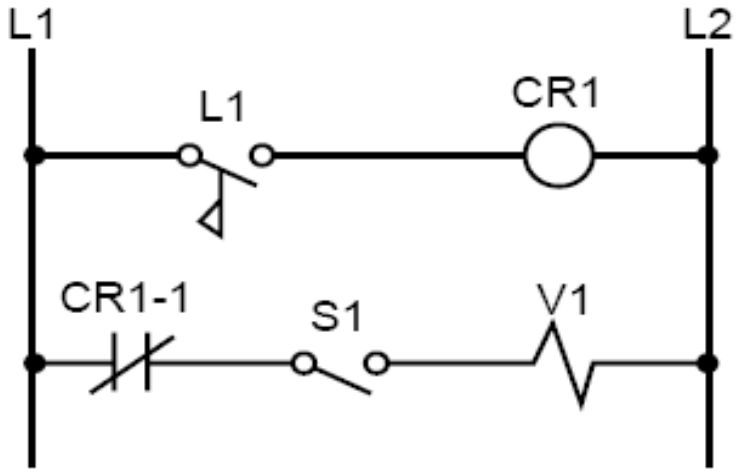




Logic Representation

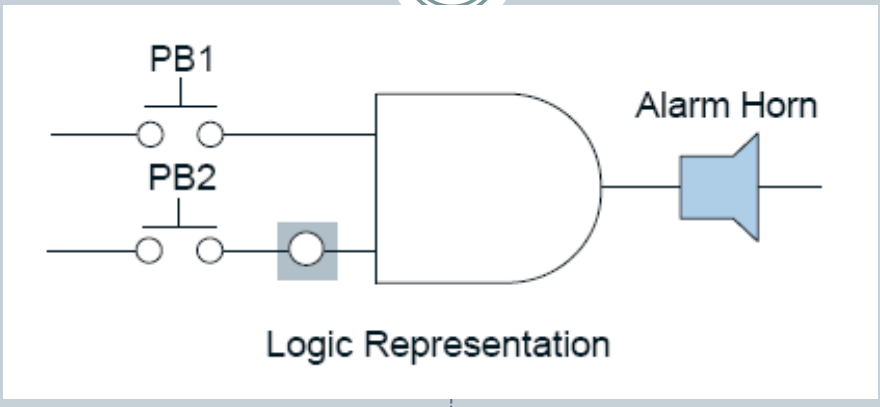
S1	L1 ($\overline{L1}$)	V1
0	0 1	0
0	1 0	0
1	0 1	1
1	1 0	0

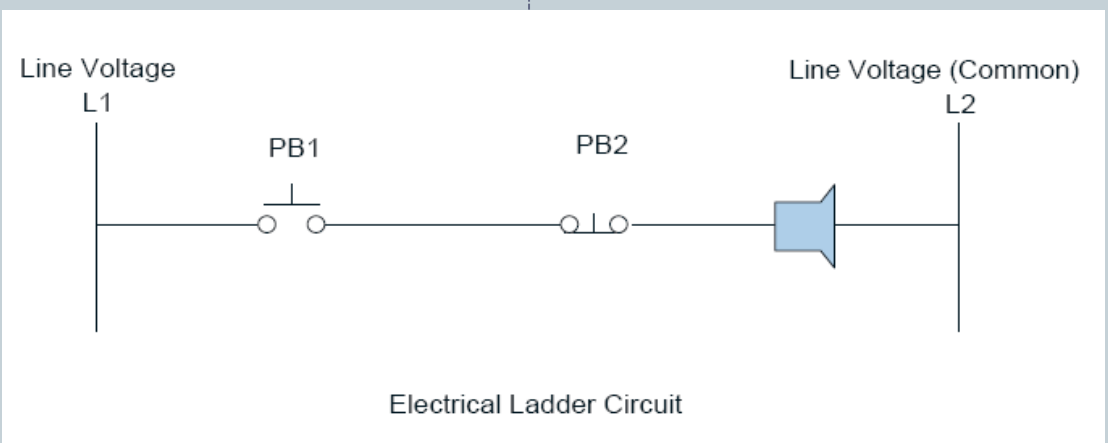
Truth Table



Electrical Ladder Circuit

Logic Functions Example





NAND / NOR Gates

Figure 3-7. Two-input NAND gate and its truth table.

NAND Truth Table	
Inputs	Output
A B	Y
0 0	1
0 1	1
1 0	1
1 1	0

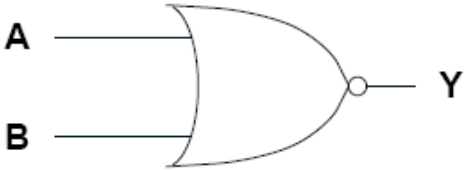


Figure 3-8. Two-input NOR gate and its truth table.

NOR Truth Table	
Inputs	Output
A B	Y
0 0	1
0 1	0
1 0	0
1 1	0

Principles of Boolean Algebra and Logic

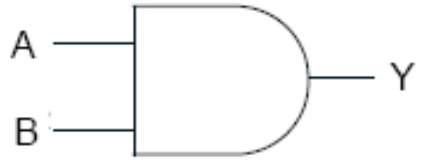
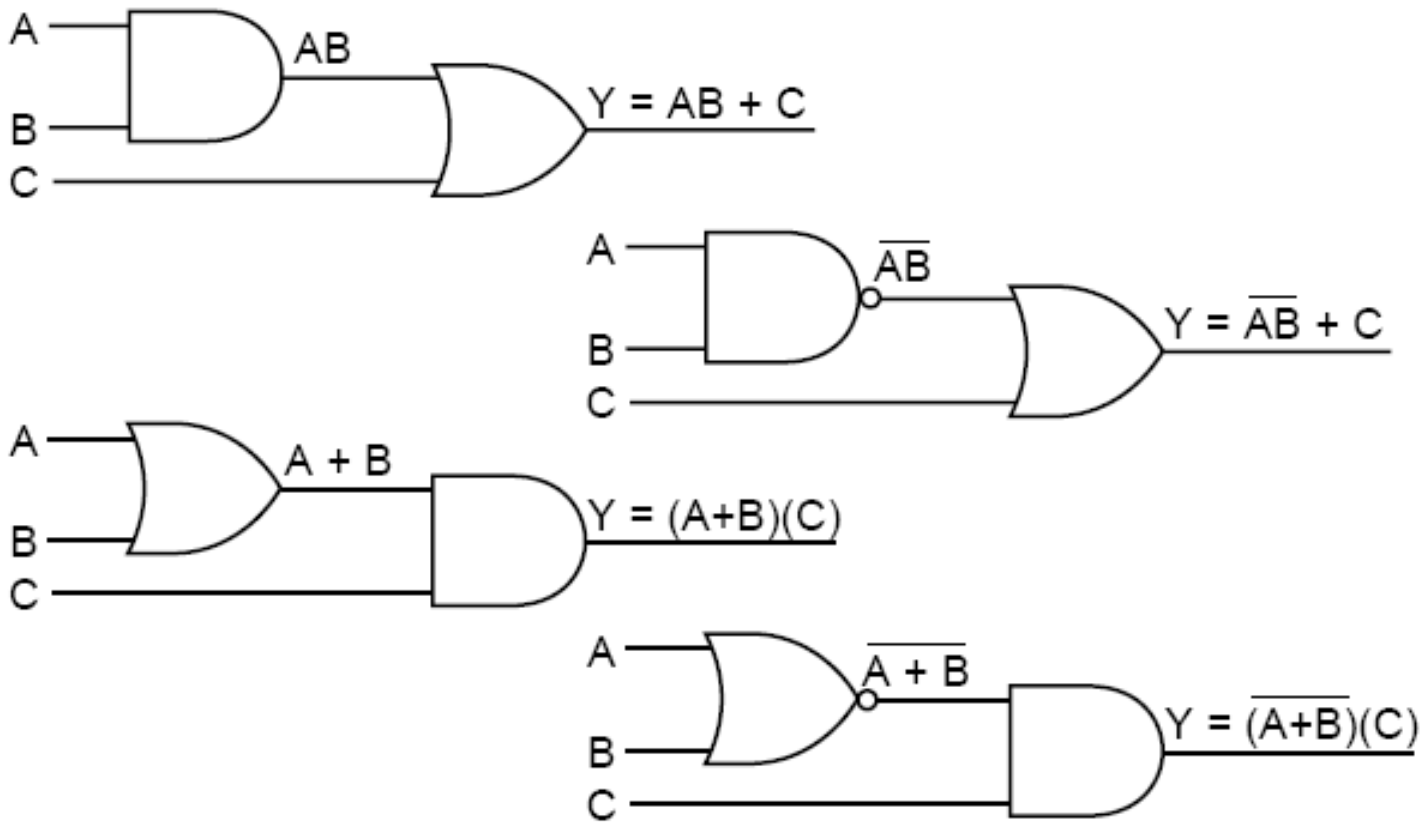
Logical Symbol	Logical Statement	Boolean Equation
	*Y* is 1 if *A* AND *B* are 1	$Y = A \cdot B$ or $Y = AB$
	*Y* is 1 if *A* OR *B* is 1	$Y = A + B$
	*Y* is 1 if *A* is 0 *Y* is 0 if *A* is 1	$Y = \overline{A}$

Figure 3-9. Boolean algebra as related to the AND, OR, and NOT functions.

Combined Gates



Boolean Algebra Rules

Commutative Laws

$A + B = B + A$

$AB = BA$

Associative Laws

$A + (B + C) = (A + B) + C$

$A(BC) = (AB)C$

De Morgan's Laws

$\overline{(A + B)} = \overline{A}\,\overline{B}$

$\overline{(AB)} = \overline{A} + \overline{B}$

$\overline{\overline{A}} = A,\ \overline{1} = 0,\ \overline{0} = 1$

$A + \overline{A}B = A + B$

$AB + AC + B\overline{C} = AC + B\overline{C}$

Distributive Laws

$A(B + C) = AB + AC$

$A + BC = (A + B)(A + C)$

Law of Absorption

$A(A + B) = A + AB = A$

PLC Circuits

- Hardwired logic refers to logic control functions (timing, sequencing, and control) that are determined by the way devices are interconnected.
- Hardwired logic is fixed and can be changed only by altering the way devices are physically connected.

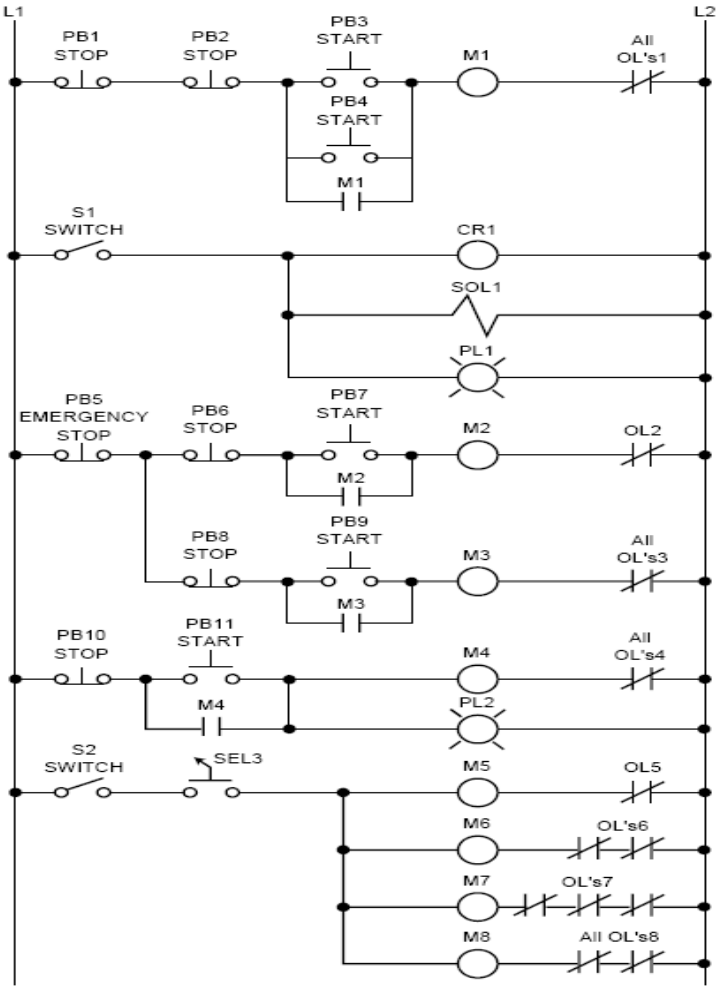


Figure 3-10a. Hardwired relay logic circuit.

PLC Circuits

- A prime function of a PLC is to replace existing hardwired control logic and to implement “programmable” control functions for new systems.
- How these instructions are implemented to convey commands to the PLC is called the **language**.
- The most conventional of the control languages is ladder diagram.
- Ladder diagrams are also called **contact symbology**, since their instructions are relay-equivalent contact symbols.

PLC Circuits

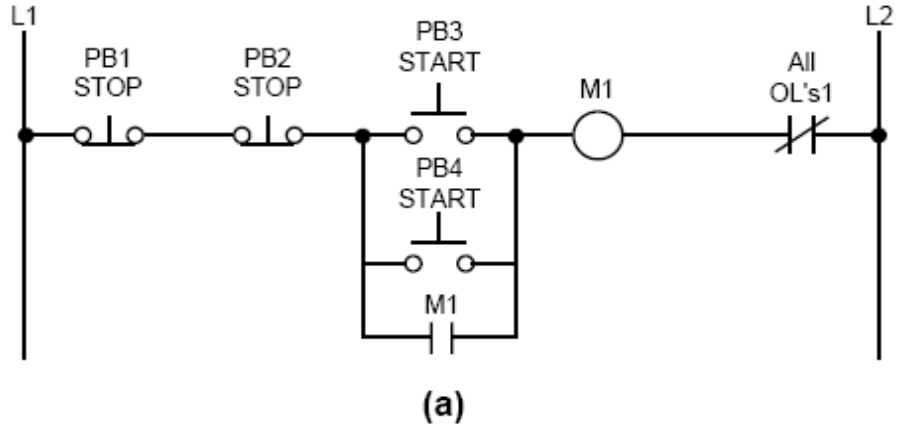
- The complete ladder circuit can be thought of individual circuits, each circuit having one output.
- Each of these circuits is known as a **rung** (or network); therefore, a rung is the contact symbology required to control an output in the PLC.
- Some controllers allow a rung to have multiple outputs, but one output per rung is the convention.

PLC Circuits

Input Devices	Output Devices
Push button	Pilot light
Selector switch	Solenoid valve
Limit switch	Horn
Proximity switch	Control relay
Timer contact	Timer

Table 3-4. ON/OFF input and output devices.

PLC Circuits



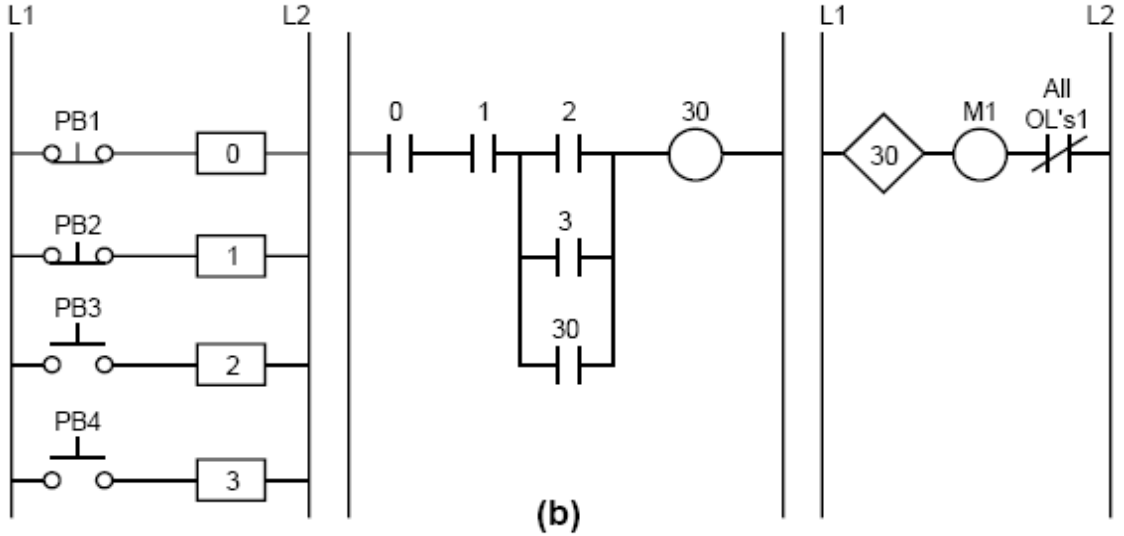


Figure 3-11. (a) Top rung of the hardwired circuit from Figure 3-10 and **(b)** its equivalent PLC circuit.

PLC Circuits and Contact Logic

- Note that the PLC diagram includes all of the field input and output devices connected to the interfaces that are used in the rung.

- A complete PLC ladder diagram program, then, consists of several rungs.

  - Each rung controls an output interface that is connected to an **output device.**

  - Each rung is a combination of input conditions (symbols) connected from left to right between two vertical lines, with the symbol that represents the output at the far right.

PLC Addressing

- Each symbol on a rung will have a *reference number*, which is the address in memory where the current status (1 or 0) for the referenced input/output is stored.

- When a field signal is connected to an input or an output interface, its address will be related to the terminal where the signal wire is connected.

PLC Addressing

- Consider a simple circuit.
- Each “real” field device (e.g., push buttons PB1 and PB2, limit switch LS1, and pilot light PL1) is connected to the PLC’s input and output modules which have a reference number—the address.
- To turn PL1 ON, one of the following conditions must occur
  - PB1 must be pressed and LS1 must be closed (or)
  - PB2 must be pressed and LS1 must be closed.

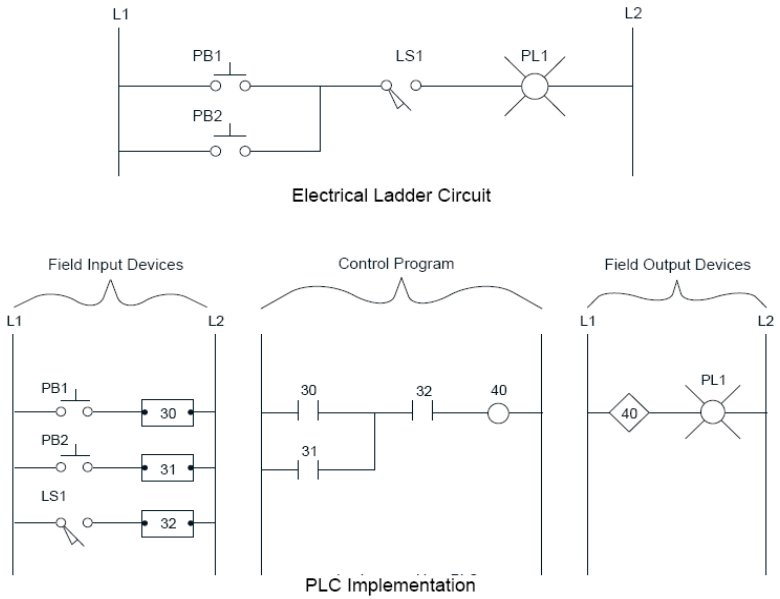


Figure 3-12. Electrical ladder circuit and its equivalent PLC implementation.

PLC Addressing

- In the PLC control program, power must flow through either addresses 30 (PB1) and 32 (LS1) or through addresses 31 (PB2) and 32 (LS1) to turn ON output 40.
- Output 40, in turn, energizes the light PL1 that is connected to the interface with address 40.
- In order to provide power to addresses 30, 31, or 32, the devices connected to the input interfaces addresses must be turned ON

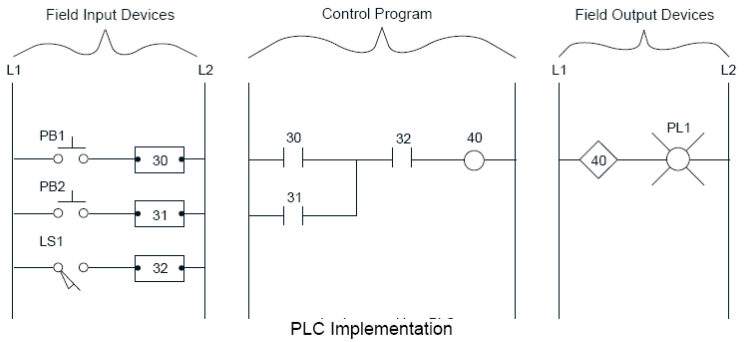


Figure 3-12. Electrical ladder circuit and its equivalent PLC implementation.

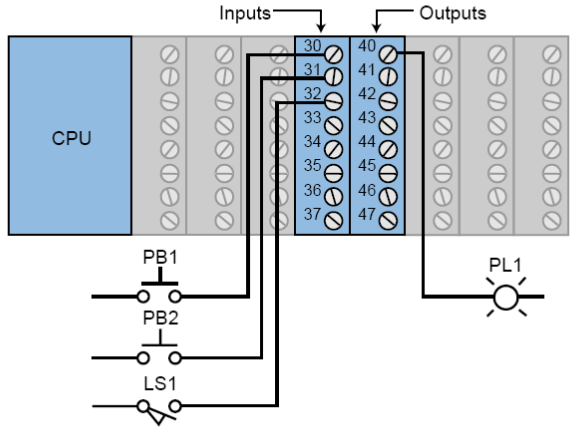
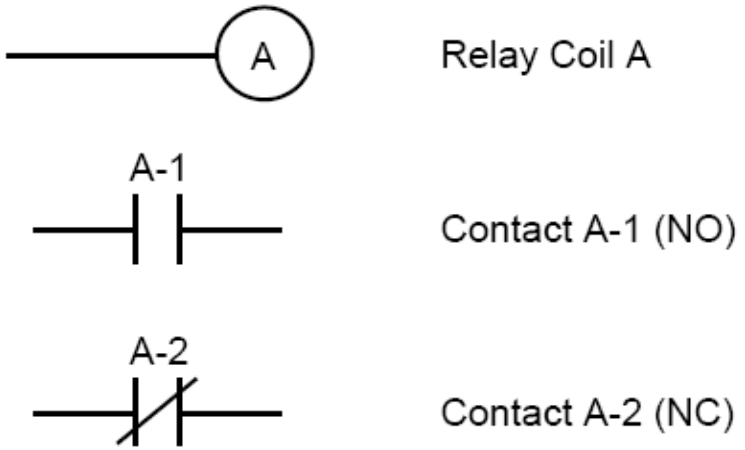


Figure 3-13. Field devices from Figure 3-12 connected to I/O module.

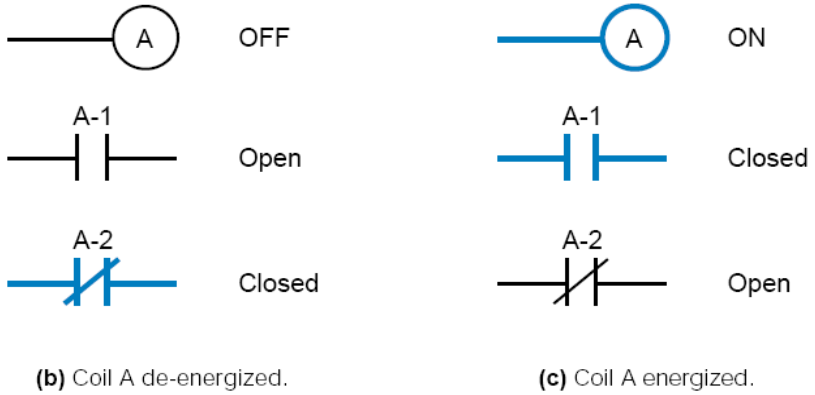
Contact Symbols used in PLC

- Consider a relay A with two sets of contacts:
  - one **normally open** contact (A-1).
  - one **normally closed** contact (A-2).



Contact Symbols used in PLC

- If relay coil A is not energized (i.e., it is OFF), contact A-1 will remain open and contact A-2 will remain closed.
- Conversely, if coil A is energized, or turned ON, contact A-1 will close and contact A-2 will open
- The blue lines highlighting the coil and contacts denote an ON, or closed, condition.



(b) Coil A de-energized. **(c)** Coil A energized.

Contact Symbols used in PLC

- Each set of available coils and its respective contacts in the PLC have a unique reference address by which they are identified.
- For instance, coil 10 will have normally open and normally closed contacts with the same address (10) as the coil.
- Note that a PLC can have as many normally open and normally closed contacts as desired.

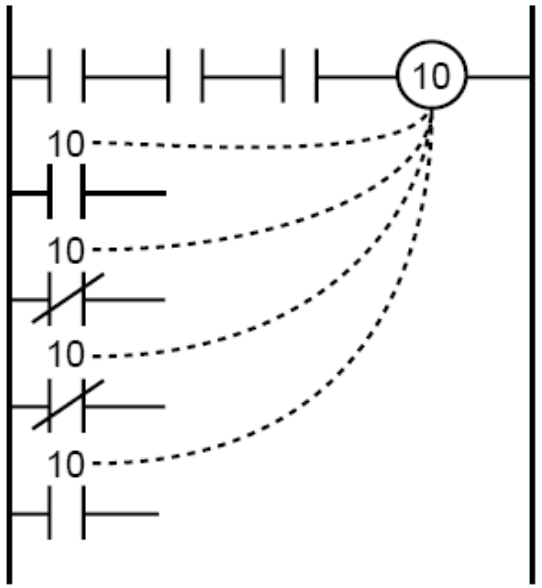


Figure 3-15. Multiple contacts from a PLC output coil.

Contact Symbols used in PLC

- A programmable controller also allows the multiple use of an input device reference.
- An example in which limit switch LS1 is connected to reference input module connection 20.
- Note that the PLC control program can have as many normally open and normally closed reference 20 contacts in as many rungs as needed.

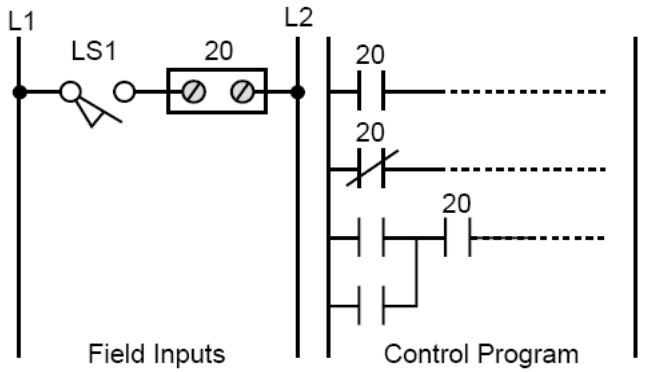


Figure 3-16. Input 20 has multiple contacts in the PLC control program.

Contact Symbols used in PLC

- **Normally open contact.**
  - When evaluated by the program, this symbol is examined for a 1 to close the contact; therefore, the signal referenced by the symbol must be ON, CLOSED, activated, etc.
- **Normally closed contact.**
  - When evaluated by the program, this symbol is examined for a 0 to keep the contact closed; thus, the signal referenced by the symbol must be OFF, OPEN, deactivated, etc.
- **Output.**
  - An output on a given rung will be energized if any left-to-right path has all contacts closed. An output can control either a connected device or an **internal output** used exclusively within the program.