

8. PLC OPERATION

Topics:

- The computer structure of a PLC
- The sanity check, input, output and logic scans
- Status and memory types

Objectives:

- Understand the operation of a PLC.

8.1 INTRODUCTION

For simple programming the relay model of the PLC is sufficient. As more complex functions are used the more complex vonNeumann model of the PLC must be used. A vonNeumann computer processes one instruction at a time. Most computers operate this way, although they appear to be doing many things at once. Consider the computer components shown in Figure 89.

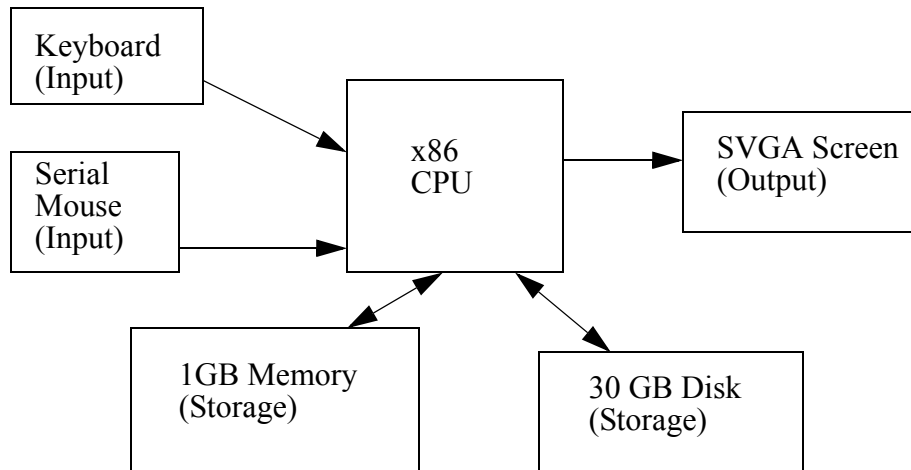


Figure 89 Simplified Personal Computer Architecture

Input is obtained from the keyboard and mouse, output is sent to the screen, and the disk and memory are used for both input and output for storage. (Note: the directions of these arrows are very important to engineers, always pay attention to indicate where information is flowing.) This figure can be redrawn as in Figure 90 to clarify the role of inputs and outputs.

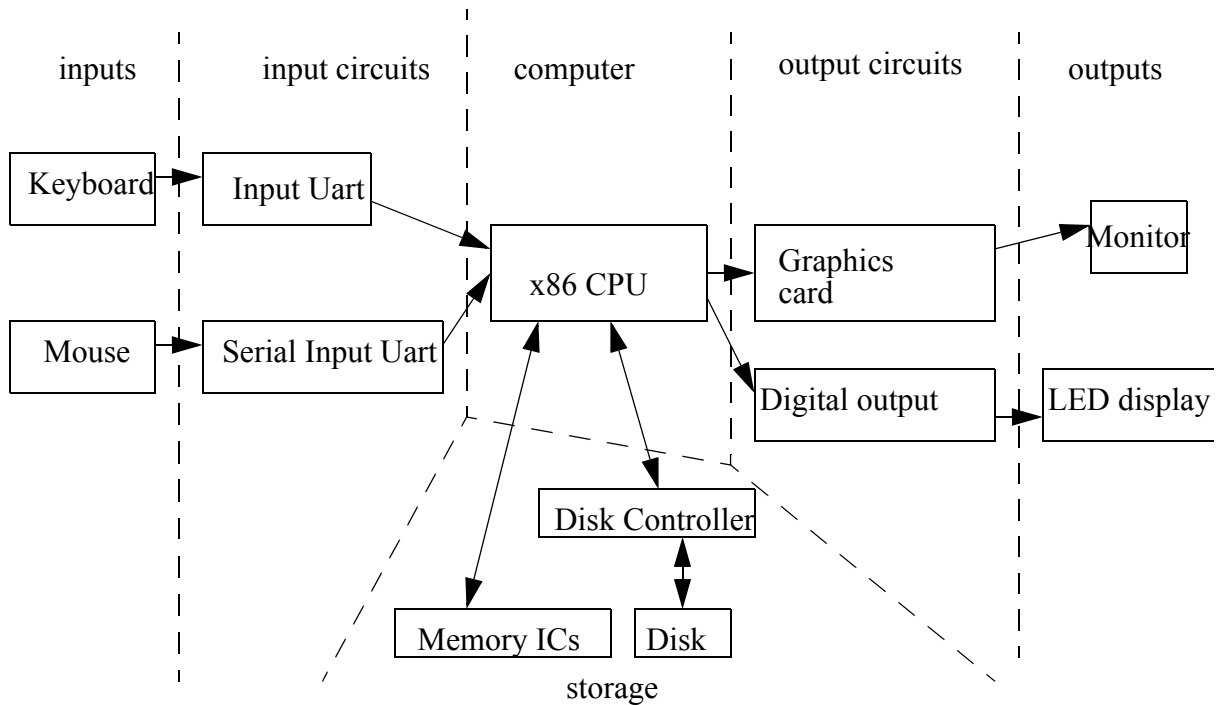


Figure 90 An Input-Output Oriented Architecture

In this figure the data enters the left side through the inputs. (Note: most engineering diagrams have inputs on the left and outputs on the right.) It travels through buffering circuits before it enters the CPU. The CPU outputs data through other circuits. Memory and disks are used for storage of data that is not destined for output. If we look at a personal computer as a controller, it is controlling the user by outputting stimuli on the screen, and inputting responses from the mouse and the keyboard.

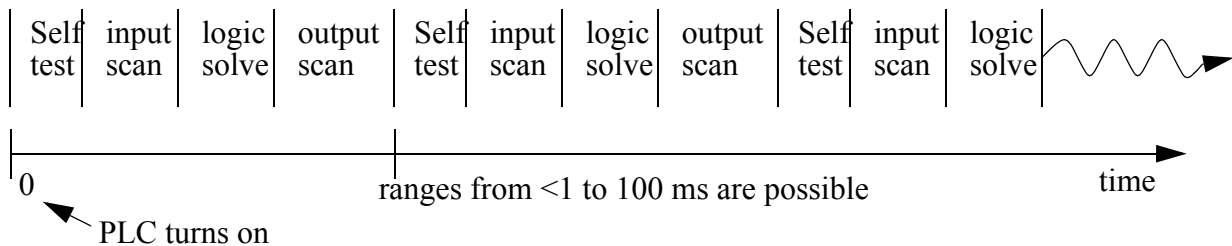
A PLC is also a computer controlling a process. When fully integrated into an application the analogies become;

- inputs - the keyboard is analogous to a proximity switch
- input circuits - the serial input uart is like a 24Vdc input card
- computer - the x86 CPU is like a PLC CPU unit
- output circuits - a graphics card is like a triac output card
- outputs - a monitor is like a light
- storage - memory in PLCs is similar to memories in personal computers

It is also possible to implement a PLC using a normal Personal Computer, although this is not advisable. In the case of a PLC the inputs and outputs are designed to be more reliable and rugged for harsh production environments.

8.2 OPERATION SEQUENCE

All PLCs have four basic stages of operations that are repeated many times per second. Initially when turned on the first time it will check it's own hardware and software for faults. If there are no problems it will copy all the input and copy their values into memory, this is called the input scan. Using only the memory copy of the inputs the ladder logic program will be solved once, this is called the logic scan. While solving the ladder logic the output values are only changed in temporary memory. When the ladder scan is done the outputs will updated using the temporary values in memory, this is called the output scan. The PLC now restarts the process by starting a self check for faults. This process typically repeats 10 to 100 times per second as is shown in Figure 91.



SELF TEST - Checks to see if all cards error free, reset watch-dog timer, etc. (A watchdog timer will cause an error, and shut down the PLC if not reset within a short period of time - this would indicate that the ladder logic is not being scanned normally).

INPUT SCAN - Reads input values from the input cards, and copies their values to memory. This makes the PLC operation faster, and avoids cases where an input changes from the start to the end of the program (e.g., an emergency stop). There are special PLC functions that read the inputs directly, and avoid the input tables.

LOGIC SOLVE/SCAN - Based on the input table in memory, the program is executed 1 step at a time, and outputs are updated. This is the focus of the later sections.

OUTPUT SCAN - The output table is copied from memory to the outputs. These then drive the output devices.

Figure 91 PLC Scan Cycle

The input and output scans often confuse the beginner, but they are important. The input scan takes a *snapshot* of the inputs, and solves the logic. This prevents potential problems that might occur if an input that is used in multiple places in the ladder logic program changed while half way through a ladder scan. Thus changing the behaviors of half of the ladder logic program. This problem could have severe effects on complex programs that are developed later in the book. One side effect of the input scan is that if a change in input is too short in duration, it might fall between input scans and be missed.

When the PLC is initially turned on the normal outputs will be turned off. This does not affect the values of the inputs.

8.2.1 The Input and Output Scans

When the inputs to the PLC are scanned the physical input values are copied into memory. When the outputs to a PLC are scanned they are copied from memory to the physical outputs. When the ladder logic is scanned it uses the values in memory, not the actual input or output values. The primary reason for doing this is so that if a program uses an input value in multiple places, a change in the input value will not invalidate the logic. Also, if output bits were changed as each bit was changed, instead of all at once at the end of the scan the PLC would operate much slower.

8.2.2 The Logic Scan

Ladder logic programs are modelled after relay logic. In relay logic each element in the ladder will switch as quickly as possible. But in a program elements can only be examined one at a time in a fixed sequence. Consider the ladder logic in Figure 92, the ladder logic will be interpreted left-to-right, top-to-bottom. In the figure the ladder logic scan begins at the top rung. At the end of the rung it interprets the top output first, then the output branched below it. On the second rung it solves branches, before moving along the ladder logic rung.

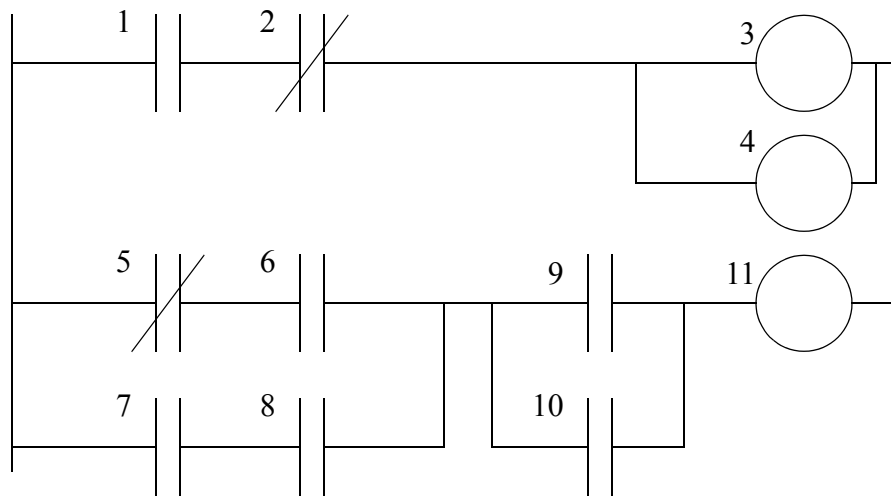
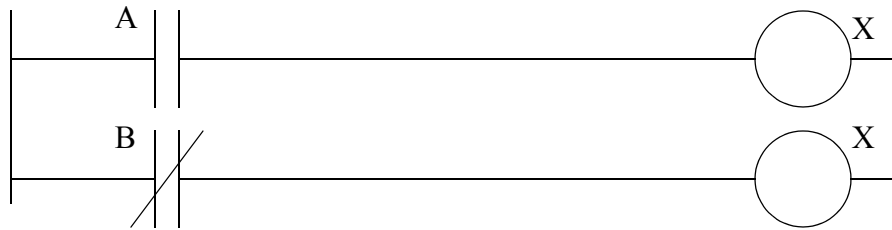


Figure 92 Ladder Logic Execution Sequence

The logic scan sequence become important when solving ladder logic programs which use outputs as inputs, as we will see in Chapter 8. It also becomes important when considering output usage. Consider Figure 93, the first line of ladder logic will examine input *A* and set output *X* to have the same value. The second line will examine input *B* and set the output *X* to have the opposite value. So the value of *X* was only equal to *A* until the second line of ladder logic was scanned. Recall that during the logic scan the outputs are only changed in memory, the actual outputs are only updated when the ladder logic scan is complete. Therefore the output scan would update the real outputs based upon the second line of ladder logic, and the first line of ladder logic would be ineffective.



Note: It is a common mistake for beginners to unintentionally repeat the same ladder logic output more than once. This will basically invalidate the first output, in this case the first line will never do anything.

Figure 93 A Duplicated Output Error

8.3 PLC STATUS

The lack of keyboard, and other input-output devices is very noticeable on a PLC. On the front of the PLC there are normally limited status lights. Common lights indicate;

power on - this will be on whenever the PLC has power

program running - this will often indicate if a program is running, or if no program is running

fault - this will indicate when the PLC has experienced a major hardware or software problem

These lights are normally used for debugging. Limited buttons will also be provided for PLC hardware. The most common will be a run/program switch that will be switched to program when maintenance is being conducted, and back to run when in production. This switch normally requires a key to keep unauthorized personnel from altering the PLC program or stopping execution. A PLC will almost never have an on-off switch or reset button on the front. This needs to be designed into the remainder of the system.

The status of the PLC can be detected by ladder logic also. It is common for programs to check to see if they are being executed for the first time, as shown in Figure 94. The 'first scan' or 'first pass' input will be true the very first time the ladder logic is scanned, but false on every other scan. In this case the address for 'first pass' in ControlLogix is 'S:FS'. With the logic in the example the first scan will seal on 'light', until 'clear' is turned on. So the light will turn on after the PLC has been turned on, but it will turn off and stay off after 'clear' is turned on. The 'first scan' bit is also referred to at the 'first

pass' bit.

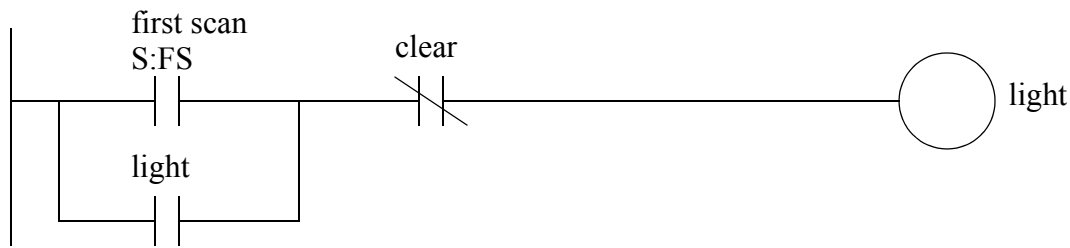


Figure 94 An program that checks for the first scan of the PLC

8.4 MEMORY TYPES

There are a few basic types of computer memory that are in use today.

RAM (Random Access Memory) - this memory is fast, but it will lose its contents when power is lost, this is known as volatile memory. Every PLC uses this memory for the central CPU when running the PLC.

ROM (Read Only Memory) - this memory is permanent and cannot be erased. It is often used for storing the operating system for the PLC.

EPROM (Erasable Programmable Read Only Memory) - this is memory that can be programmed to behave like ROM, but it can be erased with ultraviolet light and reprogrammed.

EEPROM (Electrically Erasable Programmable Read Only Memory) - This memory can store programs like ROM. It can be programmed and erased using a voltage, so it is becoming more popular than EPROMs.

Hard Disk - Software based PLCs run on top of another operating system (such as Windows) that will read and save values to a hard drive, in case power is lost.

All PLCs use RAM for the CPU and ROM to store the basic operating system for the PLC. When the power is on the contents of the RAM will be kept, but the issue is what happens when power to the memory is lost. Originally PLC vendors used RAM with a battery so that the memory contents would not be lost if the power was lost. This method is still in use, but is losing favor. EPROMs have also been a popular choice for programming PLCs. The EPROM is programmed out of the PLC, and then placed in the PLC. When the PLC is turned on the ladder logic program on the EPROM is loaded into the PLC and run. This method can be very reliable, but the erasing and programming technique can be time consuming. EEPROM memories are a permanent part of the PLC, and programs can be stored in them like EPROM. Memory costs continue to drop, and newer types (such as flash memory) are becoming available, and these changes will continue to impact PLCs.

8.5 SOFTWARE BASED PLCS

The dropping cost of personal computers is increasing their use in control, including the replacement of PLCs. Software is installed that allows the personal computer to solve ladder logic, read inputs from sensors and update outputs to actuators. These are important to mention here because they don't obey the previous timing model. For example, if the computer is running a game it may slow or

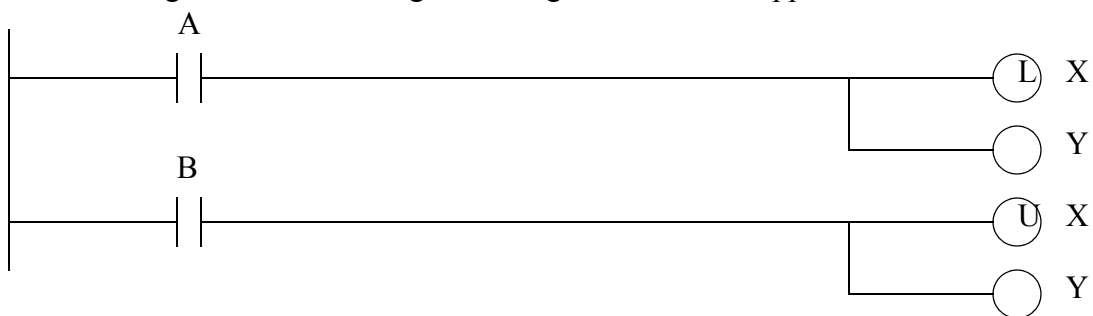
halt the computer. This issue and others are currently being investigated and good solutions should be expected soon.

8.6 SUMMARY

- A PLC and computer are similar with inputs, outputs, memory, etc.
- The PLC continuously goes through a cycle including a sanity check, input scan, logic scan, and output scan.
- While the logic is being scanned, changes in the inputs are not detected, and the outputs are not updated.
- PLCs use RAM, and sometime EPROMs are used for permanent programs.

8.7 PRACTICE PROBLEMS

1. Does a PLC normally contain RAM, ROM, EPROM and/or batteries.
2. What are the indicator lights on a PLC used for?
3. A PLC can only go through the ladder logic a few times per second. Why?
4. What will happen if the scan time for a PLC is greater than the time for an input pulse? Why?
5. What is the difference between a PLC and a desktop computer?
6. Why do PLCs do a self check every scan?
7. Will the test time for a PLC be long compared to the time required for a simple program.
8. What is wrong with the following ladder logic? What will happen if it is used?



9. What is the address for a memory location that indicates when a PLC has just been turned on?

8.8 ASSIGNMENT PROBLEMS

1. Describe the basic steps of operation for a PLC after it is turned on.
2. Repeating a normal output in ladder logic should not be done normally. Discuss why.

3. Why does removing a battery from some older PLCs clear the memory?