Programmable Logic Controllers

Alana Amunrud
EE 367, Paper 1
February 27, 2002

Abstract

Programmable Logic Controllers (PLCs) were invented to replace relay control systems. This papers tells a little about the history of PLC development. It also talks about the components that make up PLCs. Ladder Logic is discussed briefly as a common way to program PLCs. The need, and current effort, to standardize PLCs is mentioned. Soft PLCs are brought up, along with their advantages and disadvantages as compared to regular PLCs. Practical applications of PLCs are also looked at in order to give the reader and idea of the importance of PLCs in everyday life.

Introduction: The History of PLCs

In the 1960's Programmable Logic Controllers were first developed to replace relays and relay control systems. Relays, while very useful in some applications, also have some problems. The main problem is the fact that they are mechanical. This means that they wear down and have to be replaced every so often. Also, relays take up quite a bit of space. These, along with other considerations, led to the development of PLCs.

More improvements to PLCs occurred in the 70's. In 1973 the ability to communicate between PLCs was added. This also made it possible to have the controlling circuit quite a ways away from the machine it was controlling. However, at this time the lack of standardization in PLCs created other problems. This was improved in the 1980's. The size of PLCs was also reduced then, thus using space even more efficiently.

The 90's increased the collection of ways in which a PLC could be programmed (block diagrams, instruction list, C, etc.). They also saw PLCs being replaced by PC's in some cases. However, PLCs are still very much in use in all sorts of industries, and it's likely that they will remain there for quite some time.

General Information on Programmable Logic Controllers

I. How they work:

Programmable Logic Controllers are basically relay control systems put in a very small package. This means that one PLC acts basically like a bunch of relays, counters, timers, places for data storage, and a few various other things, all in one small package (example pictures below).

Figure 1a: A Toshiba TCU111 PLC **Figure 1b:** Keyence KV Series (very small)

* There are input and output "relays" (often implemented by transistors rather than actual relays) that send and receive information from the outside world. The outputs are set according to the inputs.

* Counters can either increase (1, 2, 3, etc.), decrease, or both, depending on the manufacturer. Some PLCs also include high-speed counters in the hardware.

* Different sorts of timers are also available, depending on the manufacturer. Timers can affect when the output is changed (in relation to the change on the input), or how long the output stays changed once the input goes back to normal.

* The data storage registers can manipulate date, and can often also save data when power is removed from the PLC.

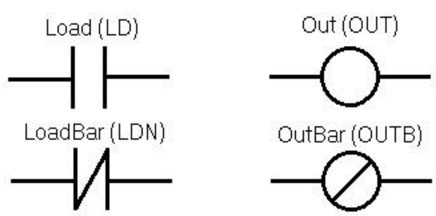
These parts do not really physically exist in the PLC. Instead they are mimicked by the parts that make up the actual chip: I/O modules and a CPU, connected to power supplies.

A PLC generally does three basic things: check the inputs, run through the program, and change the outputs. It then loops back to the top and starts again. This seems amazingly simple, though it can be made quite complex with enough different inputs and outputs and combined with various timers and counters. The scan time is the time it takes for the PLC to go through the three basic steps. This time is very important, as it affects how quickly the inputs can be read. The inputs need to be on (or off) long enough for the PLC to read them in. If they are not on that long, then problems occur. Fortunately, there are ways to fix this problem. One of the best ways is to use an interrupt whenever an input goes high. This will insure that the PLC does not miss the change.

II. Programming:

Ladder Logic is often used to program PLCs. Symbols like the ones below are used.

Figure 2: Ladder Logic Symbols



The Load is like a normally open contact, and the LoadBar a normally closed contact. The Out represents a relay coil, or a normally open output. The OutBar is not always available, but it is like a normally closed relay coil. These four symbols are used to represent almost everything in PLC logic. Things can, in fact, get rather complex with multiple inputs and outputs and "seals" (where an active output sort of acts like an input). There are also other symbols that are used for this sort of PLC coding. Programmable Logic Controllers also include counters and timers, and so there are ways to represent both in Ladder Logic.

There are also other ways of programming PLCs, such as Sequential Function Charts (SFCs), Instruction Lists (ILs), Function Block Diagrams (FBDs), and Structured Texts (STs).

III. Other Possible Operations

Besides the simple logic discussed above, there are numerous other things that PLCs can do. In fact, many of the instructions are quite similar to the assembly code used with the HC12 in EE 371. Data can be moved and stored. Two values can be added, subtracted, multiplied, or divided. Some PLCs include other algebraic and geometric operations. In some cases decimals can be used as well as integers. Boolean logic (AND, OR, etc.) is also available. All of these things can be useful in certain applications.

IV. Standardization:

One of the biggest problems in the PLC world is the lack of standardization. Different manufacturers have different sorts of counters and timers available. Different programmers program different chips in different ways, which results in some confusion. Therefore, there is a movement to standardize PLC program interfaces (with IEC 61131-3). This would allow people with different knowledge and skills to work together on the same project with minimum confusion.

V. Example Applications:

There are literally thousand, if not millions, of possible applications for PLCs. They are used in machining, packaging, material handling, and automated assembly, as well as many other areas. For example, consider an oven. A cook wants to heat the oven to a certain temperature, so she enters the desired number. A PLC takes this input and changes the output - a heater of some sort. The PLC then takes the temperature of the oven as another input. When this input reaches a certain level, the PLC again changes the output to turn the heater off. When the temperature drops below a certain level, the PLC again turns on the heater, thus keeping the temperature fairly constant. This saves the cook much trouble. Without a PLC she would have to monitor a thermometer herself and turn the heater off and on at the critical times. Cooking is difficult enough without having to also pay attention to the temperature of the oven.

VI. Soft PLCs:

Soft PLCs are basically software within a computer that acts like a PLC. This is a relatively new development in the PLC world. There are some cases in which Soft PLCs, programmed in Basic or C, are better to use than normal PLCs. However, it would be wise to test things before completely trusting the performance of a Soft PLC.

VII. Advantages and Disadvantages of Using PLCs:

On the positive side of things, PLCs are very good for controlling outputs (based on inputs). They are amazingly robust, and able to stand all sorts of difficult conditions, such as extreme temperatures or dust in the air. They last from 5 to 10 years, depending on various circumstances. They do not have contacts that wear out, like relays do. They also can switch fairly quickly without heating up much, in direct contrast to relays. This means that cooling costs are decreased. Also, if need be, they can be quite a ways away from the object they are controlling because of their communication abilities.

On the downside, PLCs are not very good at handling large amount of data, or complex data. Computers are better for that sort of thing. PLCs are also not very good with databases or displaying data. Compared to relays, PLCs are almost always a better choice. However, under certain conditions, or perhaps if the logic is very simple, a relay may be a better choice.

Conclusions

1. While PLCs may be relatively old in the electrical world, it is obvious that they are not yet outdated. It is possible that Soft PLCs, or some similar thing, would someday replace most PLCs. However, that day is most likely far off. Programmable Logic Controllers today are used all over. With all the advantages they have, as listed in this paper, it is highly unlikely that they would be replaced soon.
2. The one major disadvantage to PLCs is the lack of standardization. This causes much confusion if the PLC used for an application is replaced by one from a different manufacturer, or if a PLC programmer is replaced by a person with a different understanding of PLC programming.
3. Replacing Relay Control Systems with PLCs was a great step in the mechanical world. In general, PLCs are more robust and require less maintenance, thus making the things they control work better, inspite of the environment. Overall, PLCs appear to be an excellent solution for many different problems.

Sources Used

"History of the PLC" as told to Howard Hendricks by Dick Morley. © 199602992 R.Morley Inc. http://www.barn.org/FILES/historyofplc.html/.

"PLCs Toshiba Programmable Logic Controllers Introduction." © 1994-2001, Transtronics, Inc. http://xtronics.com/toshiba/plcnf.htm.

"Programmable Logic Controllers PLCs." © 2001 High Tech Services HTS, Cary, Raleigh, RTP, North Carolina, NC. http://www.htservices.com/Tools/PLC/.

"Programmable Logic Controllers (PLCs)." © 2002 Keyence Corporation. http://world.keyence.com/products/programmable_logic_controllers/plc.html.

"Welcome to PLCopen." © 1993-2001 PLCopen. http://www.plcopen.org/.

"Your Personal PLC Tutor Site" by Phil Melore. © 2002. http://www.plcs.net.