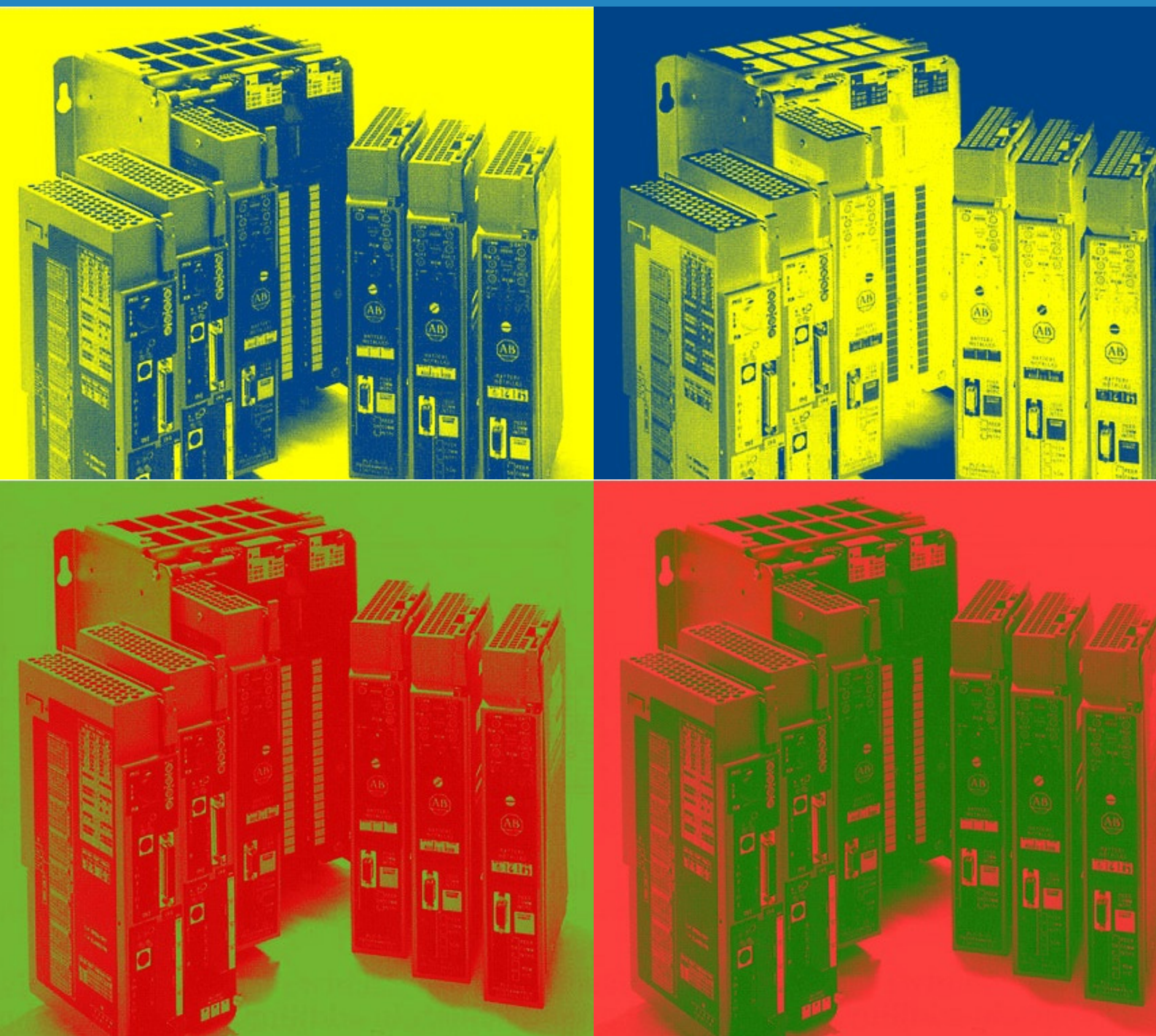
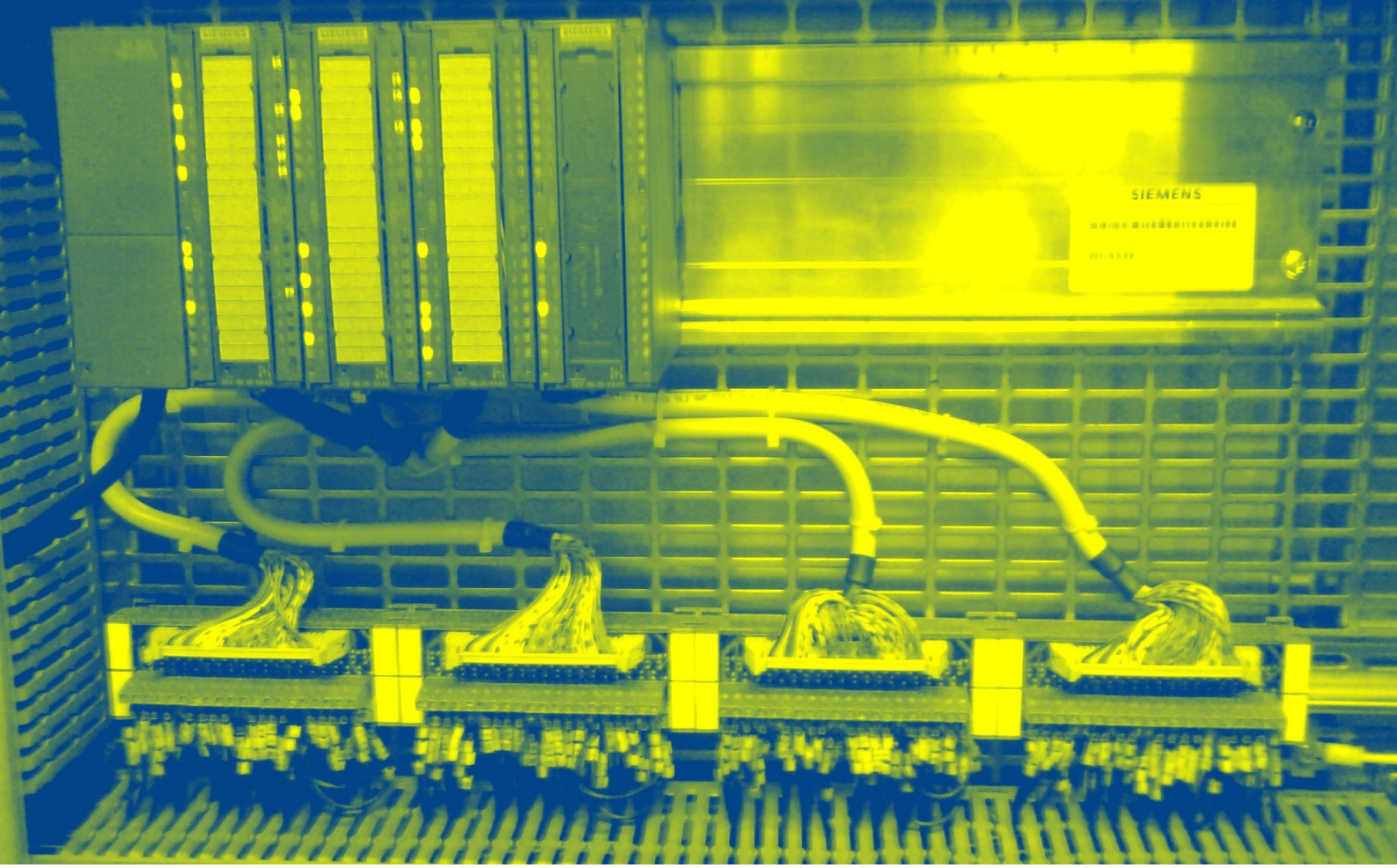
Optima

In control since 1995

The Logic behind Choosing Logic Controllers





Automated control systems are the backbone of modern production. It is hard to imagine engineering a manufacturing process without the help of a specialized computing system. This is where Programmable Logic Controllers (or PLCs) are significant. The choice of a PLC brand, therefore, is not a simple decision to make because that can affect the productivity of a whole process line.

Optima Control Solutions is an expert independent automation company and has over 16 years of experience with a wide range of PLC equipment. This report briefly compares the physical characteristics and the functional benefits of three PLC products and gives unbiased perspectives regarding the appropriate applications of each.

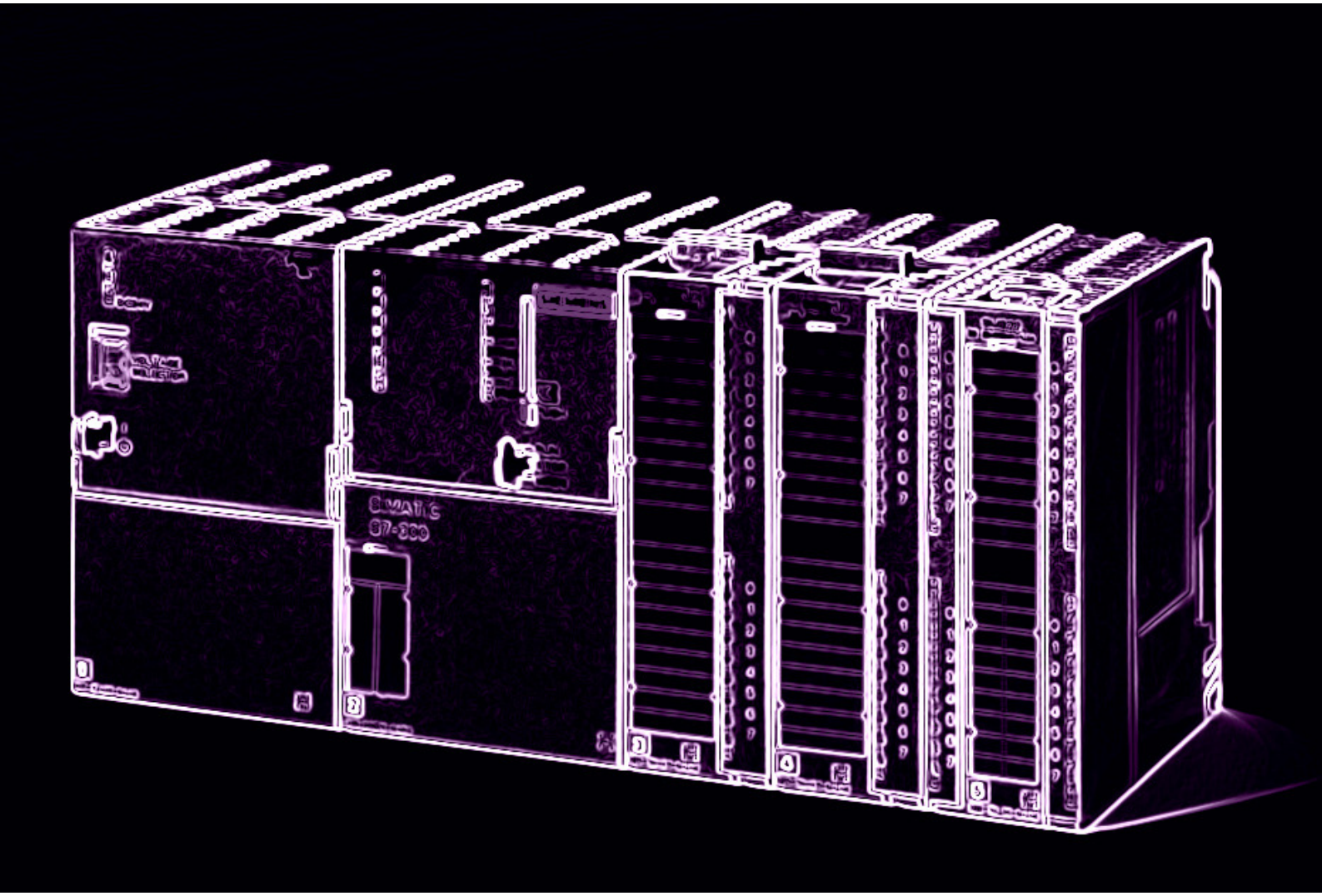
The first part of this report describes the basic principles and architecture of a programmable logic controller. The second part provides short comparisons of 3 PLC brands. The final part is a debate between Optima's senior engineers about the various benefits of each PLC brand (focusing particularly on Siemens and Allen Bradley). Ian Brady and Nick Maclean offer two different views on the pros and cons of the products of the two main PLC manufacturers.

Why PLC?

Since their inception in the 1960s in the US automotive industry, Programmable Logic Controllers have become part of virtually every mass manufacturing process across all industries and geographies. For manufacturers understanding the logic of a PLC is not a necessity. Appreciating the way it makes a difference to the production process, however, is.

These are some of the benefits of employing the latest PLC technology:

- **Efficiency** – recent PLC models have greater memory capabilities and less space requirements. Depending on the machine's specifications, the PLC size will vary from nano to large.
- **Flexibility** – A single PLC may run several machines at a time.
- **Cost efficiency** – Most PLC manufacturers offer controllers with varying capabilities. Some machines can be successfully controlled by lower range PLCs, thus eliminating the need for costly investments in high-end models.
- **Safety** – Safety PLCs can reduce the risk of industrial accidents significantly.



Logic inside the box

As with most computers, a PLC is often seen as a black box that receives, processes and sends out signals. Looking inside the box is important when trying to make the most cost effective and at the same time most benefitial for a production line decision.

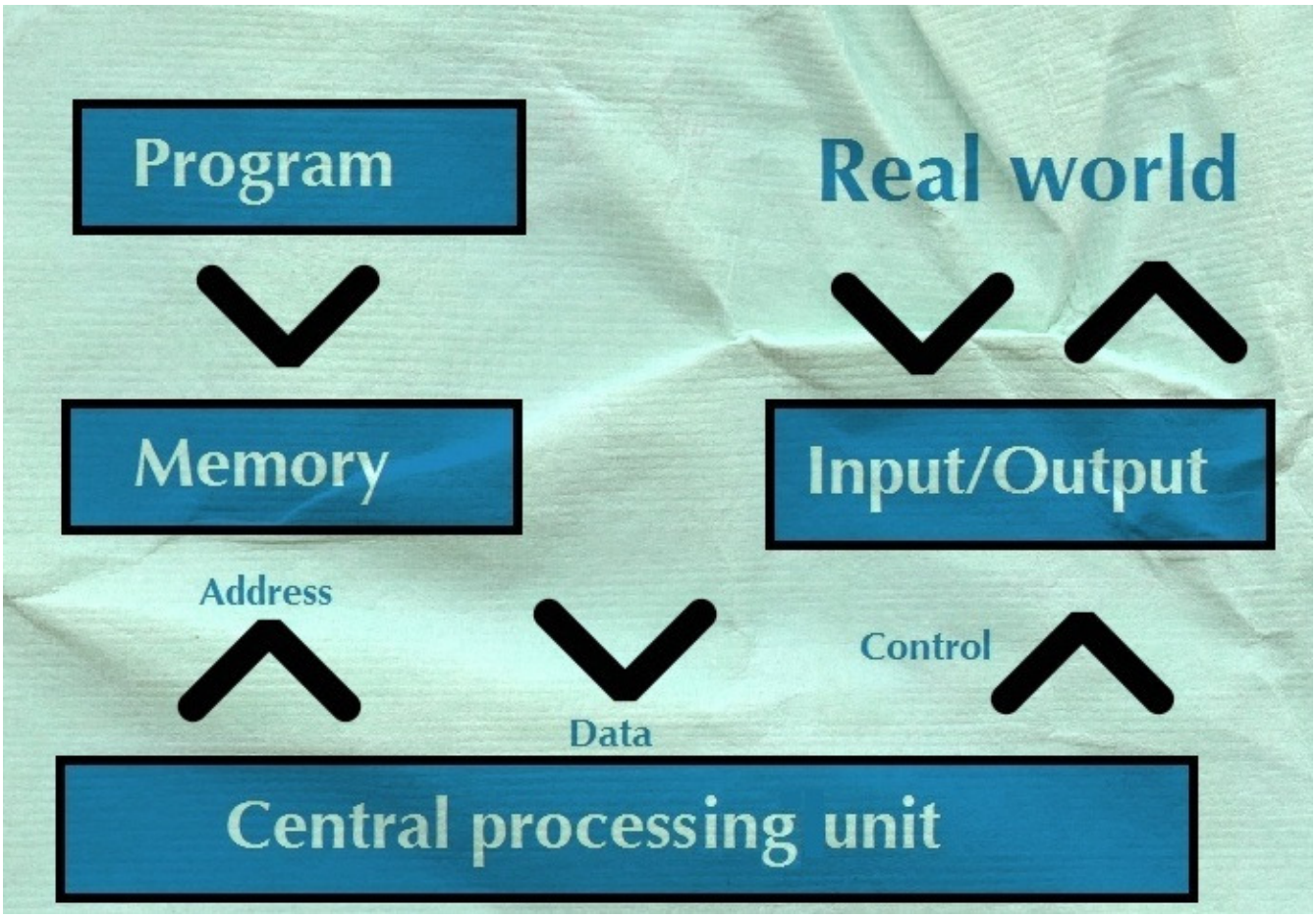
The three main components of a PLC are the input/output modules, the central processing unit(CPU) and memory.

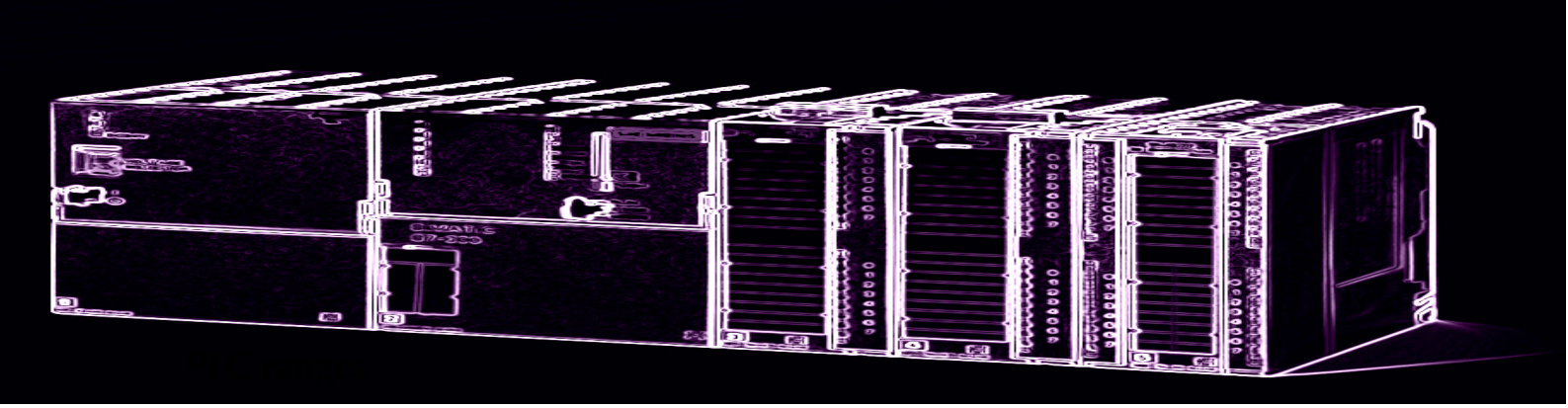
Input and output modules are converters with the function of receiving and sending analog or digital signals from and to parts of a machine. I/O modules now have remote capabilities, meaning that a single PLC can control parts of a machine at significant distance, without the need of extra wiring.

The CPU executes the program written for the particular PLC by using input signals and translating them into appropriate outputs. It is interesting that a PLC is able to receive analog signals and send out digital ones – this flexibility makes the controller applicable to almost any production line irrespective of industry and proces..

As any other type of computer the PLC needs memory to store the latest input and output data and make the appropriate translations, calculations or comparisons. The memory capabilities of a PLC, however, are highly restrained – less than 100 Mbytes memory sounds like a thing of the past for a personal computer, but it is an impressive value for a PLC.

A simple representation of what is inside 'the box'





The functionality of every PLC model is usually deciphered behind complex numerical values that may not be easily understood by everyone. Optima Control Solutions offers a simple comparison tool to help decision makers find the most effective solution for their production line problems.

The following PLC models are classified according to the complexity of tasks they are able to complete. This list is not exhaustive and certain models can be seen as niches of their own. Other PLC brands such as Omron, Schneider Electric and GE Fanuc are not listed below but tend to offer a similar range of functions. Optima is extensively using Siemens, Allen Bradley and Mitsubishi PLCs and is able to compare their functional benefits.

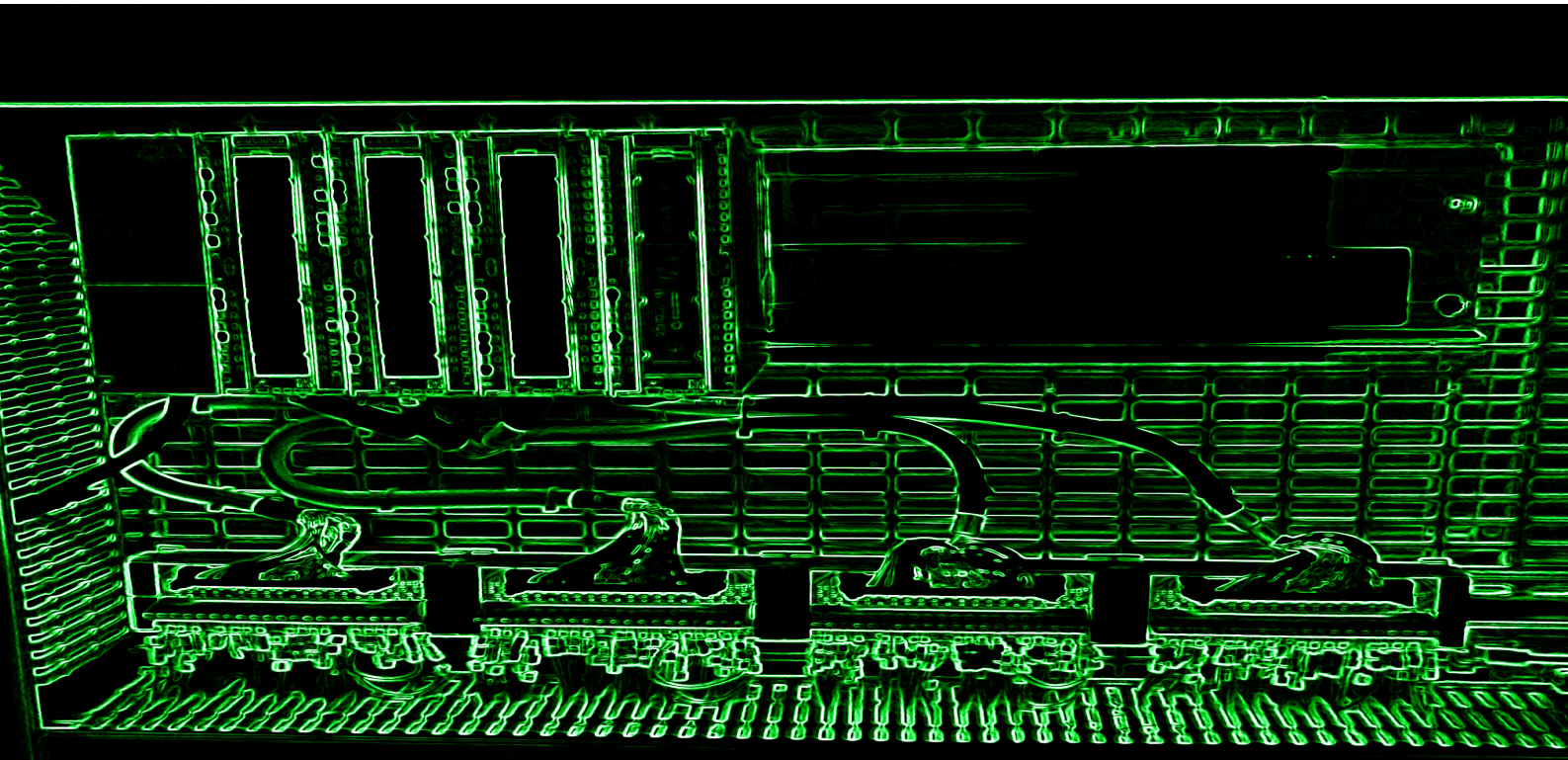
○ Low cost solutions for easy repetitive control tasks – fixed configurations, expandable

Brand	Hardware (base models)	Software	Overview	Price range
Siemens LOGO!	8 digital inputs, 4 digital outputs	Drag and drop functionality	8 basic logic functions, 31 special functions	£80-£200
Allen Bradley Pico	12 inputs, 8 outputs	PC connection capability, easy-to-use interface	Memory size – 512 instructions	£140 - £230
Mitsubishi Alpha	15 inputs, 9 outputs	Programmable with Windows AL-PCS/WIN	Wide operating temperature: -25˚C to 55˚C	£90 - £170
Toshiba T1 Micro	8 inputs, 8 outputs	Indirect addressing, 76 insertable function instructions	Pulse/PWM output for stepper motor control - can be used for continuous or high speed switching.	N/A**

Low cost solutions for open-ended control tasks – modular configurations

Brand	Hardware (expansion modules)	Software	Overview	Price range each
Siemens S7-200 Micro	Up to 32 inputs and 32 outputs	STEP 7 Micro/WIN	Tailored functionality – digital, analog, specific or communicative	CPU: £100 - £450 + I/O modules: £60 - £380
Allen Bradley MicroLogix 1100,1200,1400 series	Up to 16 inputs and 16 outputs	RSLogix 500 on Microsoft Windows	Optional memory module for program and data backup	CPU: £280 - £650 + I/O modules: £100 - £220*
Mitsubishi MELSEC FX3U-64MT	Up to 32 inputs and 32 outputs	MELSOFT suite of Automatio n software	Base models expandable up to 384 I/Os by adding expansion modules and networked I/O points.	CPU: £400 - £700 + I/O modules: £90 - £180
Toshiba V-Series S2T/S2E	Up to 8 inputs and 8 outputs	T-series compatible	Comment Storage in CPU. Device/reg. tags and comments can be stored in CPU module and be seen online.	N/A

* Allen Bradley ***Temperature*** I/O modules cost approx. £300



Mid to high performance ranges - high flexibility and high speed controllers

Brand	Hardware (expansion modules)	Software	Overview	Price range each
Siemens S7-300 Micro	Up to 32 inputs and 32 outputs; up to 8192 I/O; 256 analog I/O	Built in communications options incl. MPI, Profibus DP	CPU can support up to 32 additional cards	CPU: £300 - £ 3600 + I/O modules: £100 - £500
Allen Bradley CompactLogix series	Up to 32 inputs and 32 outputs	RSLogix 5000 Enterprise Series software	Removable CompactFlash facility; Motion control	CPU: £600 - £2300 + I/O modules: £90 - £380
Mitsubishi Q series	Up to 32 inputs and 32 outputs	EZSocket, complying with international standards	Multi processor support	CPU: £370 - £2100 + I/O modules: £120 - £240*
Toshiba V-Series S2T/S2E	N/A	N/A	N/A	N/A

* Mitsubishi Special Function Modules cost up to £700.

High-end solutions for very demanding tasks (high fail-safe requirements, high restart costs, expensive downtimes, little supervision)

Brand	Hardware (expansion modules)	Software	Overview	Price range each
Siemens S7-400 Power	Up to 32 inputs and 32 outputs; Up to 65536 I/O and 4096 analogue I/O	STEP 7	Satisfies safety requirements up to SIL 3	CPU: £780 - £8328 + I/O modules: £200 - £1200
Allen Bradley ContolLogix series	Up to 32 inputs and 32 outputs; Up to 128 000 digital I/O and 4000 analogue I/O	FactoyTalk suite from Rockwell Software; RSLOGIX5000	The Logix architecture supports motion control components that work in a wide variety of machine architectures.	CPU: £2800 - £7120 + I/O modules: £220 - £1250

Optima debate

SIEMENS \ Allen-Bradley

In the previous section a small sample of the large range of PLCs available are reported. The PLC market is dominated by two giants – Siemens and Rockwell. Optima works with other PLC brands but is most experienced with these two and is, therefore, able to provide a detailed expert opinion about both Siemens and Allen Bradley controllers.

Two contrasting opinions prevail among Optima's engineers. Some say Allen Bradley PLCs are simple, hassle-free and that these benefits save both time and money to the customer. Others believe that Siemens PLCs are very advanced in many aspects and that this improves production processes. We asked Ian Brady and Nick Maclean, Optima's senior engineers, to give their own arguments on the topic.

Allen-Bradley

By Ian Brady

I shall start like this - you cannot say that an Allen Bradley (**AB**) PLC is faster or better than a Siemens one. In certain categories they are very similar. Now, why would you choose one over the other? Simple question: first, what does the company already use – if they have Siemens equipment they will carry on using it! It is usually the biggest criteria, not whether one is better than the other. Also, their maintenance people have got some experience with it so rather than learning about a new piece, they will carry on using the old one.

So if your question is "why Allen Bradley is better than Siemens in certain categories?" my answer is – it is not. I have an opinion, however, that if you are to give two maintenance men who have not seen or not used these things before a problem on a machine at 2 a.m. and they have not seen it for 6 months, **they will find the fault quicker on an Allen Bradley**, Toshiba or Mitsubishi **PLC than they would do on Siemens**. I think it is easier to understand how it works and fix it! At the end of the day it is not whether one is better than the other, it is just that the customer might find it **easier to use**.

Also, when I am commissioning, I find **debugging a lot quicker and a lot easier with Allen Bradley than with Siemens.** That may be just down to the fact that I am more used to AB than Siemens, it does not mean that one is better than the other. **AB is easy to fault find, easier to modify, easier to understand.** That, for me, is **a big plus especially for the customer** who would find it easier to work with this piece of kit than with Siemens.

[H: Do you think that certain PLCs are more appropriate for, say, extruders, than other PLC brands?]

Absolutely not! It depends again – if you have a customer who says 'I want to use only Siemens' you cannot put an AB PLC.

Sometimes we have choice; we say 'Hold on, we have a choice here so let's **design the best mix!**' We have had both cases. **We have used Parker drives with AB and with Siemens**. One machine I am doing in Scotland has Siemens PLCs and an SSD drive. **The easiest mix for every machine is different.**

Why would I use an AB PLC? Sometimes, **because of its functionality*!* If you want** to do servo opposition control, **a very accurate system, it may be better to use AB PLCs because their function blocks are more appropriate for that particular application.**

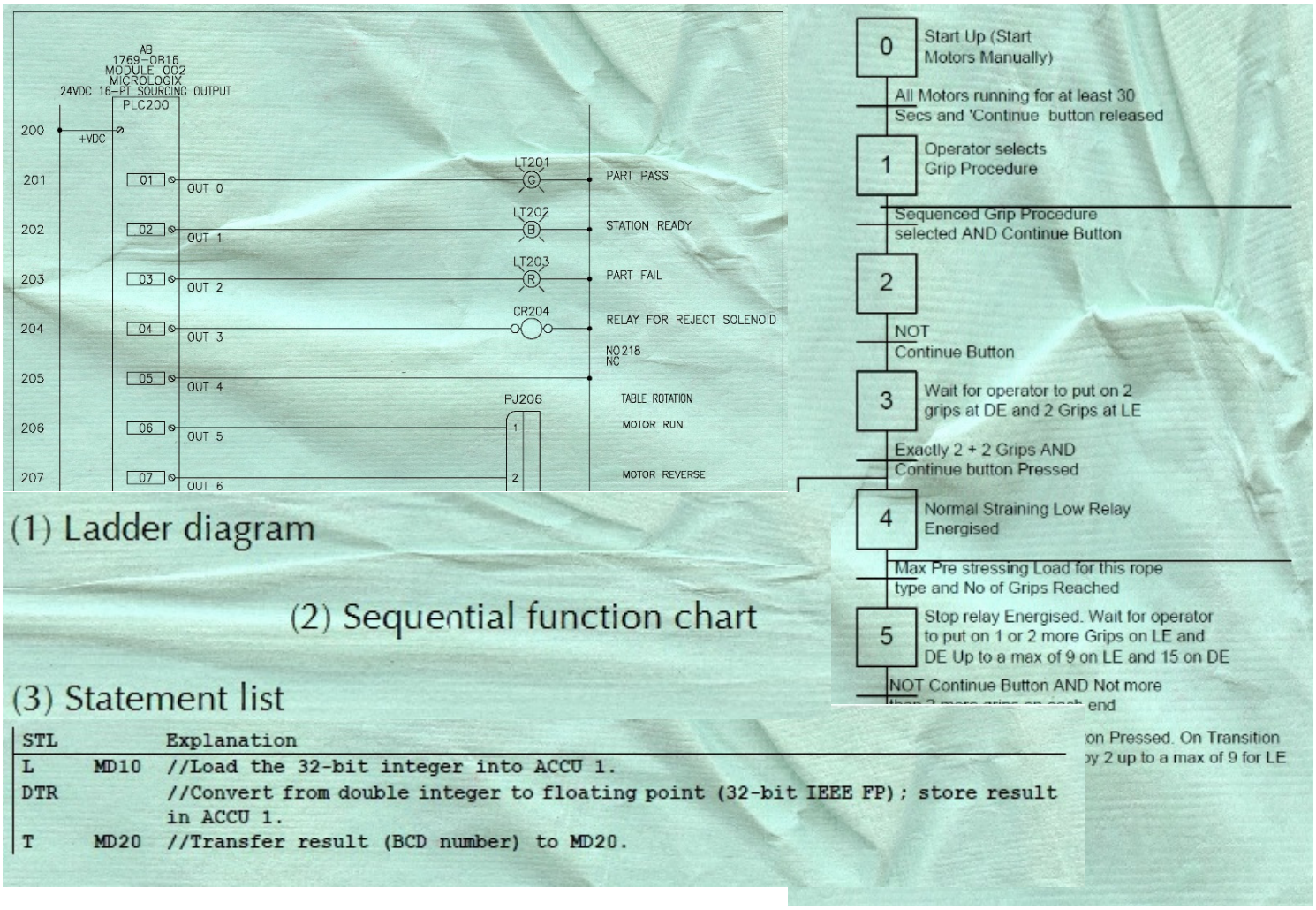
[H: Can you explain what a function block is in more detail?]

Here is one example. When you start a machine, you start at 0 meters/minute but after one minute you want it to run at, say, 500 meters/min. To get from 0 to 500, we call that a ramp and what we do is ramping the reference value. That ramp we can put in a block, so what we have in a block is just input and output. What goes on in the block really does not matter. All you need to know about input is: there is a start (or stop), there is a maximum value and a minimum value and there is rate – for example, ramp time per minute. Finally, there is output also known as reference. This is called a function. **In Siemens terms that would be called a FC** (function call) – once it is written, it can be used as many times as needed. But it will not do anything else – everything it will do is to produce a ramp output every single time. **In AB words, these function blocks would be called subroutines**.

Next thing, software can be written in many different ways – **ladder diagrams** (1), SFC – **sequential function charts** (2) or in language terms known as **statement lists** (3). Now, (3)

is where it all gets complicated and harder! Siemens writes a lot of the PLC programs like (3). When you are trying to figure out how things work, when you look at the machine and ask “show me what is happening?” the charts change colours, for example. If the inputs are on, the chart will go green so that one can see how the line works. In (3) you will have a couple of columns and there will be numbers all over the place. If the code is written in this way, **it is really hard for the maintenance engineer to understand what all of these column values mean** – it means nothing to him. **Siemens was written for programmers, AB and all the rest of them were written for electricians and maintenance people.** (1) and (2) are easy to understand, (3) is not. It is sometimes frightening – the problem is if the processor you’ve got is not too powerful. What it does is **to turn a statement list into a ladder diagram** in its own little brain**. If the ladder is too complex and gets translated into a code, the processor may never translate it back so, all you see on the screen is code!** This might be completely staggering for maintenance people!

That is the difference when using a simple PLC or simple and understandable way of writing. **The problem with machines is that nobody looks at them when they are working.** When they are not working, people lose money! **The maintenance guy** goes to the machine when it is not working. He **will try to find out what is wrong with it and the quicker he does that, the sooner it gets back into production. So if you use a simple PLC such as AB, you may find the problem faster and make less loss.**



[H: Can you explain me the difference between fixed and modular configuration?]

The fixed configuration has a certain number of inputs and outputs, it is totally fixed, and you cannot add bits to it! Basically, you get what you get. Some can be **extended but rarely**. These are used for fairly simple control tasks. Here **(4)** is one example – Siemens Logo. I can show you four different makes and they all look identical. There are a few models with some additions but more or less **their functionality is the same.** When you go up the range, PLCs are also pretty much fixed. Top of range PLCs are modular, where you can add many more modules with many more inputs and outputs. Quite obviously, **the prices reflect how advanced a PLC is – and prices will not vary too much across brands as much as they will vary across the individual product ranges of every PLC manufacturer!** If you buy an expensive PLC you will be sure that its control capabilities are better. But again unless you own a nuclear power plant and must have the most advanced control systems, you do not need to spend over £5000 for a single PLC from the high end of the line.

[H: What about scan time?]

If you **look at (2)**, the system works from top to bottom before it repeats the process and starts over.
In programming there is something called parallel processing or let us call it another way – multitasking. A PLC tends not to do that – it tends **to do one thing at a time**. It runs one thing and then goes on to the next task. So a **scan is the time it takes a PLC to do a job from top to bottom**. It depends on how big the program is, how fast the processor is and so on. Well**, it all depends on how much you are going to spend,** really. **Sometimes you can find that smaller PLCs such as the Siemens 300 series are faster than bigger ones (400 series).** The 400 are higher in price but the 300 will be able to do total a lot more. The **big AB Contrologix PLCs**, for example, can take up to 10000 inputs and outputs; **the smaller ones are more limited so they perform faster.** It is a matter of what you need – big PLCs have the option to add more than one processor in a rack so if one fails the other takes over – these are perfect **for safety critical tasks**.

(4) Siemens LOGO!

SIEMENS

By Nick Maclean

I do not have enough experience with Allen Bradley **(AB)** and I have always been a Siemens programmer. It suits the style of programming for very complex tasks, very mathematical, analytical tasks. In my view, **Siemens supports these tasks better than an AB**. This might be because I do not know the Rockwell stuff very well.

[H: What about ease of use and programming?]

If you are a trained programmer, you will find the Siemens one more intuitive. If you are not a programmer by nature then you want to do everything with ladder logic. And you can go only so far with ladder logic**, to go beyond that you need to use a low language that obviously gives more flexibility and more efficiency of programming.**

[H: Do you think Siemens is more

appropriate for particular production

processes then?]

I think for most of the work we do, you could realistically use either. In terms of speed of development, I would say **Siemens is a better one once you get used to it.** It is **easier to make reusable components and transfer them between projects.** It has a **very integrated environment.** For example, with HMI screens –reading the text directly out of the project. Probably the AB does the same??

[H: What are the PLC components that you consider being superior in a Siemens piece?]

The programming language as an integration is more superior. There are some shortcomings in using statement lists, **so an AB PLC would be easier to program**. But **for complex tasks I must say Siemens is the better one.** At the bottom level they are very similar. When you get to the more **in-depth features is where they differ**. There is a number of languages available for Siemens PLCs. Some of them we use, some – we do not; some languages will become more important in the future. For example, they have a language called **SCL. It is very similar to Pascal and compiles to a low level language with native code in the newer processors.** Older ones use interpretative code. So the new versions of SCL compile native code **that will allow processors to execute much more efficiently and will allow people to use programmers from the industrial marketplace** – those who are experienced Pascal programmers, they will not need to be assembly code co-programmers.

There are different levels of people's understanding of PLCs. **Those who prefer ladder logic or sequential charts will tend to use AB PLCs**. Siemens has an advantage in statement lists programming. **For very heavily mathematical stuff and highly complex tasks I would choose Siemens.** Array handling is one of its weaknesses but then the SCL code makes that a lot more possible.

[H: Would you recommend a certain PLC

brand for, say, coating lines control? Or

this decision should be project specific?]

It really depends on what the customer's support network is for any given product. If they ask us for a recommendation we will look at their machine and design the best mix. **In certain industries it is more common to find certain PLC brands. For example, in the steel industry Bosch Rexroth or Modicon are very popular.** A mixture of either brands(Siemens and AB) is typical, too.

[H: What about function blocks? Can you compare AB and Siemens in terms of function blocks?]

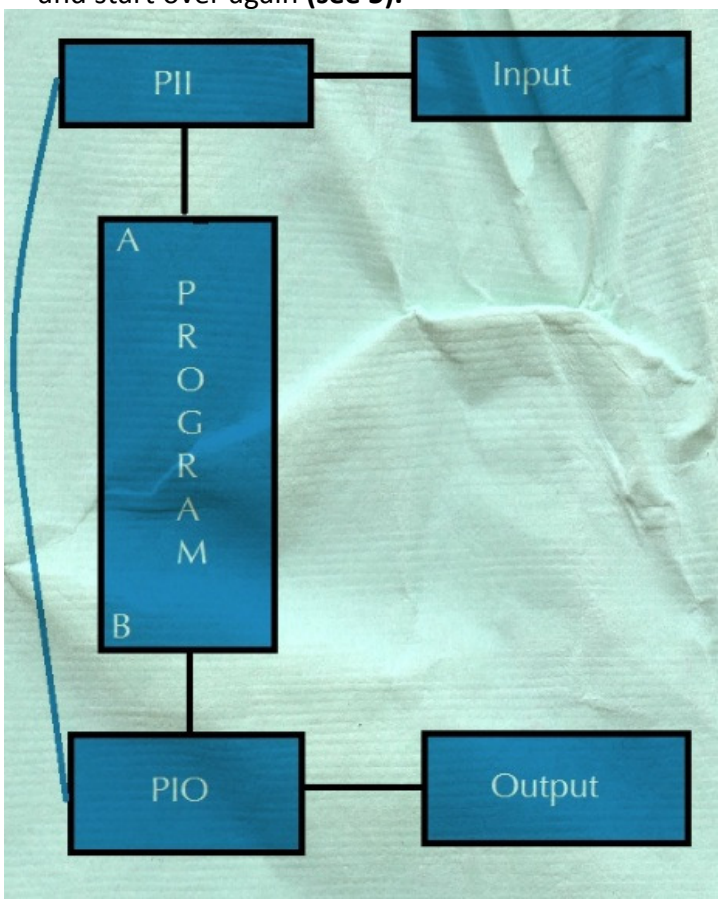
Function block programming in a Rockwell is easier for some engineers. **With an AB you can produce reusable blocks but I know people who have tried using them more than once and it did not work very well**. I suspect that is a limitation in their knowledge rather than the function blocks themselves. *[laughs*] Function blocks are very easy to use with Siemens and reuse them from one project to the next one. It is a question of how they are written – it is a matter of experience of using the tool, really.

[H: Can you tell me more about the types of PLC configurations and how do they differ?]

Fixed configurations tend to be the very small PLCs, we do not use small ones – they are very standardised and easy. We work with more modular configurations, as our projects are bigger. **For a machine with the size of a table, you can get away with a simple fixed configuration**. We are normally dealing with much larger pieces of equipment; they have specific requirements that outstrip what a fixed PLC piece offers as a capability. All PLC manufacturers fall in both categories – not only Siemens or AB. We have worked with Mitsubishi and GE Fanuc, for example – all of them have a wide range – from basic to high-end.
I think **both Siemens and AB are more specialized in modular configurations**. They do fulfil the fixed configuration criteria just to compete in this niche and create brand presence. To be honest, **other manufacturers such as Mitsubishi and Omron dominate the fixed PLCs market.**

[H: Scan time. What can you say about it for both Siemens and AB?]

AB processors, the high-end ones, are very very fast. This is because they are slightly different – they have **the acyclic I/O update** which is…a bit difficult to understand. **The way most PLCs work is cyclically**: they run to the end of their program and start over again **(see 5).**



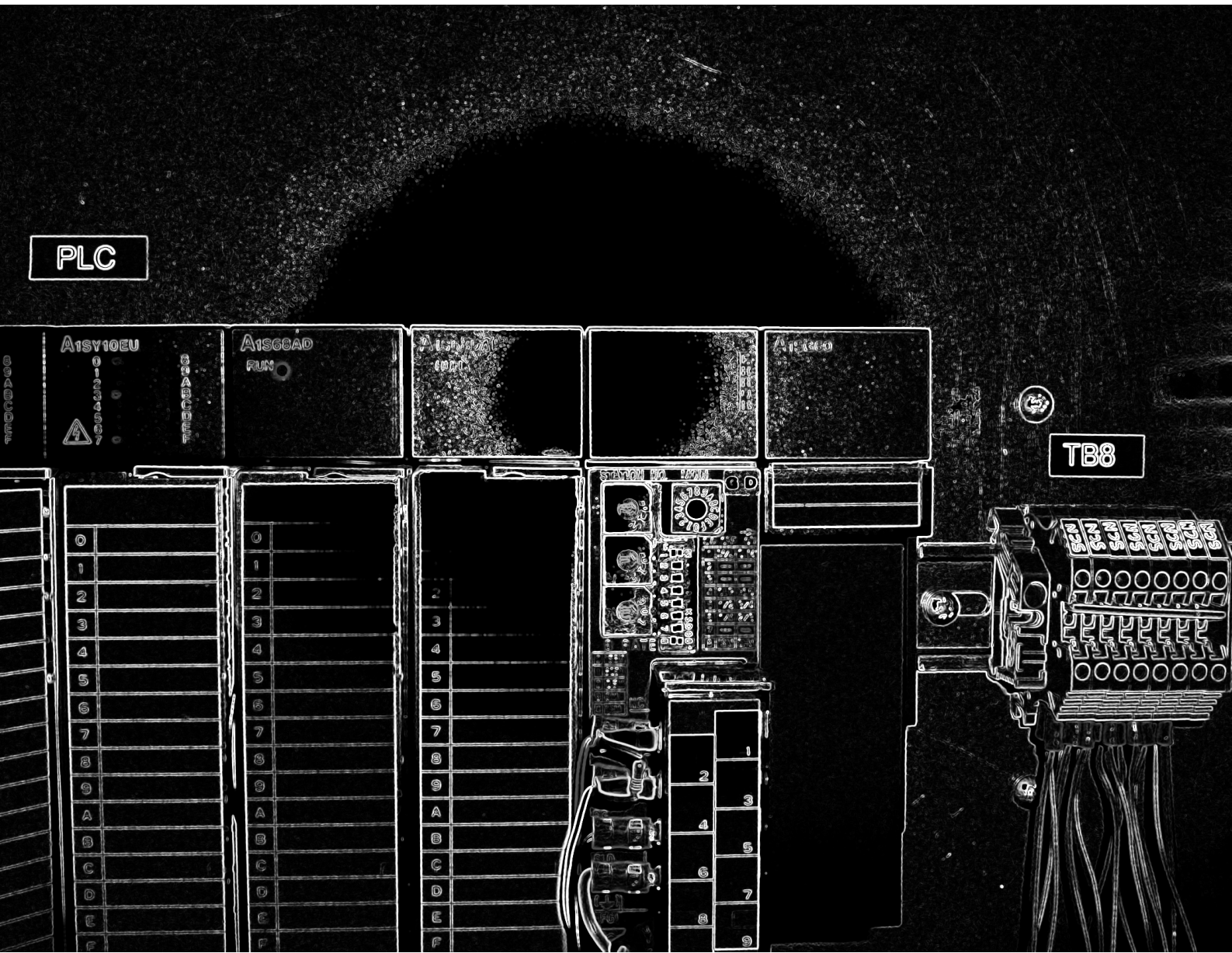
(5) I/O update cycle

At the start of every program execution, they read the input values into what is called 'process image' (or PII). Then the program executes and at the end of the program there is a bit called PIO – process image output. At the end of the process the PIOs are transferred into actual outputs. That is a cyclic operation or a cyclic I/O update. With an AB processor, it is actually not executed cyclically. In a Siemens program, for example, you know that if an input is true in point A (beginning of the program), it will also be true in point B (end of the program). If that input is updated

halfway through, it may not be true at the end of the program execution – that can influence how you design the software. With an Allen Bradley processor **there are complexities that are not necessarily obvious to people who only understand ladder logic**. So, it is an acyclic I/O update that you most often need on Allen Bradley. I do not see it as an advantage; I see it as a disadvantage. In terms of cycle time, the AB processor is much quicker but it is a question of being **quick and** being **quick enough** against a certain cost. In terms of price, they tend to be quite competitive with each other based on performance. **Total cost of ownership is another thing people consider and it comes down to what resources are available in-house**.

[H: Does memory size matter?]

One good argument is 'why would you want to put massive amounts of memory in a PLC?' That would imply that a program is very inefficient. If you could get the same program running in a smaller amount of memory then it is a lot **more efficient.** There is some disadvantage to that as well – you cannot download the source code into the PLC. In some of the newer ones, you can – they are all moving in that direction.



Disclaimer: All data provided in this report is for informational purposes only. Opinions expressed in this report are those of the relevant contributors. These do not reflect recommendations of one brand over another.

Images sourced from:

1. Optima® gallery – all copyrights reserved.
2. Siemens® logo – www.siemens.com
3. Allen Bradley® logo – www.rockwellautomation.com

Design :

www.picnic.com

Written by:

Hristina Stefanova

Revised by:

Michael Hill