Optima

In control since 1995

Drives Software Programming – using PLC or Drive bespoke programming?

A variable-speed drive's purpose in life is to control the speed and torque of a motor. In essence, it takes AC electricity from the mains supply and converts it to drive a motor in a controlled way. To accurately control the torque and speed of a motor the drive requires an appropriate "speed demand" signal (from an external source usually) together with feedback signals from motor mounted transducers; e.g. In the case of a speed loop, a drive receives its speed demand signal (or set point) from an analogue source (potentiometer) or a digital source (over a field bus comms network). This speed demand sets the drive the target speed. A corresponding "actual speed" signal (from a motor mounted tachometer or encoder) then feeds back to the drive to say how fast the motor is turning and where the rotor is in its $360^{0\ \text{radial}}$ position. Today's microprocessor controlled drives run complex control algorithms to accurately control the speed and torque of the load motor.

Variable speed drives are tuned in two ways: 1) to match the drive to the electrical characteristics of the motor by auto tune routines (standard drive functionality these days), and secondly to match the physical load conditions (by drive commissioning engineers) to match the response of the drive to varying load conditions. These tuning mechanisms will ensure the drive-motor combination is optimised to maintain the desired speed.

Applications vary in their degree of difficulty, sometimes complex and often time-consuming drive software programming must take place.

Optima Control Solutions Ltd. specialises in drive applications. The systems they engineer nearly always include PLC, HMI, SCADA and VSD equipment.

Many drive manufacturers such as Siemens and Parker SSD have developed application software (often based on Macro type solutions) that are designed to meet most common drive applications and control requirements. However, it's not unusual for certain projects to require bespoke software to be written.

One frequently asked question in applications of this nature is, “Should one use the drive’s internal programming blocks or write one’s own blocks in the PLC to control the machine?” The answer is not a straight forward one.
When building up a drive configuration engineers might need to employ many mathematical functions of varying complexity. Ranging from simple AND and OR gates to more complicated diameter calculation, tension profiles and PID tuning functions (Proportional Integral and Derivative terms).

Given that there are two possible solutions, some experience is necessary if the best decision is to be made.

Option 1 – using the Drive internal function block programming

Assuming the application can be controlled using the function blocks available, this option is often the quickest solution. Arranging the connectivity of prewritten function blocks with all their internal maths taken care means the engineer must only select and connect the right blocks in an appropriate configuration. Naturally, this requires good working knowledge of machine control and the control requirements of the application at hand. Nevertheless, it has definite advantages. We list these in the table below.

As always, there are tradeoffs to consider, e.g. different drive manufacturers employ different ways of programming the drives using their own software tools. For example, Siemens alone offers two programming options - 'Starter' (the simpler of their two offers) or 'Scout' (designed to engineer highly complex applications) . Parker SSD offers DSE Lite or DSE with similar characteristics (**see figure 3**). So familiarity (or expertise) with the programming software is a pre-requisite in order to build effective drive configurations.

Advantages	**Disadvantages**
Drive programming software written and supported by drive manufacturer.	Each manufacturer has its own software to learn. Some quite complex!
Quick to build a configuration.	Integration to HMI and PLC is more difficult as more data needs to be transferred to the PLC
Some software packages have built in trace features like an oscilloscope. This is useful when testing.	--

Table 1. Internal Drives Blocks – pros and cons

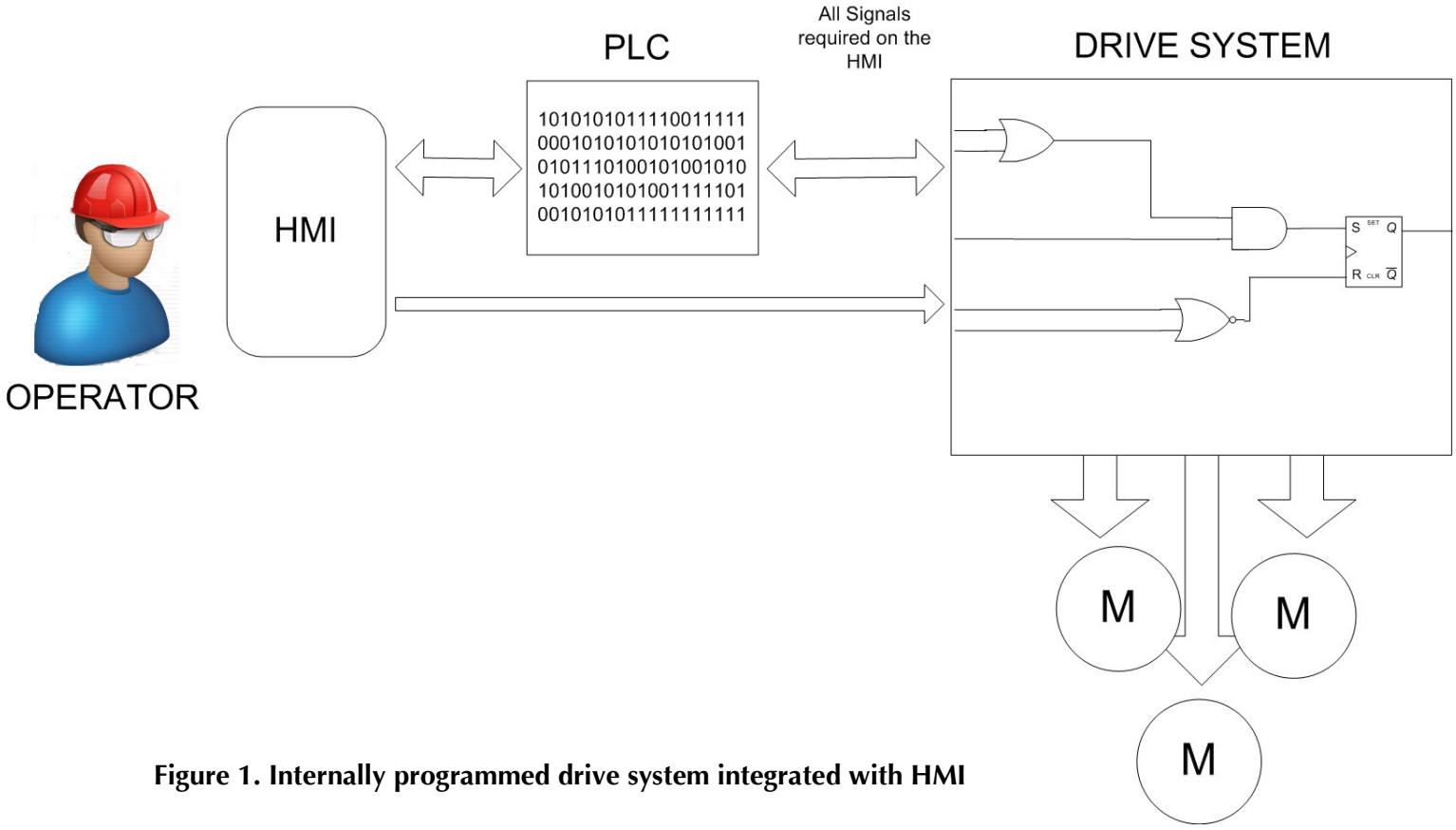

Figure 1. Internally programmed drive system integrated with HMI

Option 2 - Drive programming using PLC resident functions

Learning to expertly use different drive programming packages is a complicated task that takes time. This reason often justifies an engineer's choice of hosting the drive functionality software in the PLC(**see figure 4**). Often, control data can be passed to and from most manufacturers' drives from the environment of the PLC with which the engineer is already familiar with. Thus limiting the range of software tools with which the engineer must be aware of (***table 2***).

This does mean however that much more complicated mathematic functions will have to be coded by the engineer, often proving time consuming and difficult. In the main, PLCs are best suited to sequential control needs, they do this very easily and reliably. However, when higher level mathematics is involved much more powerful processors should be specified and the structure of the functional software is considerably more complex.

Advantages	**Disadvantages**
Common control for any drive manufacturer .	Complex maths requires significantly more engineering.
Once written it is quick to implement as all systems are the same	Complex maths requires much more processing power than simple code.This pushes the PLC cost up.
Only simple signals are sent to the drive (start, stop and speed).	--
HMI Integration is easy as all important numbers already exist in the PLC	--

Table 2. Using PLC programming blocks – pros and cons

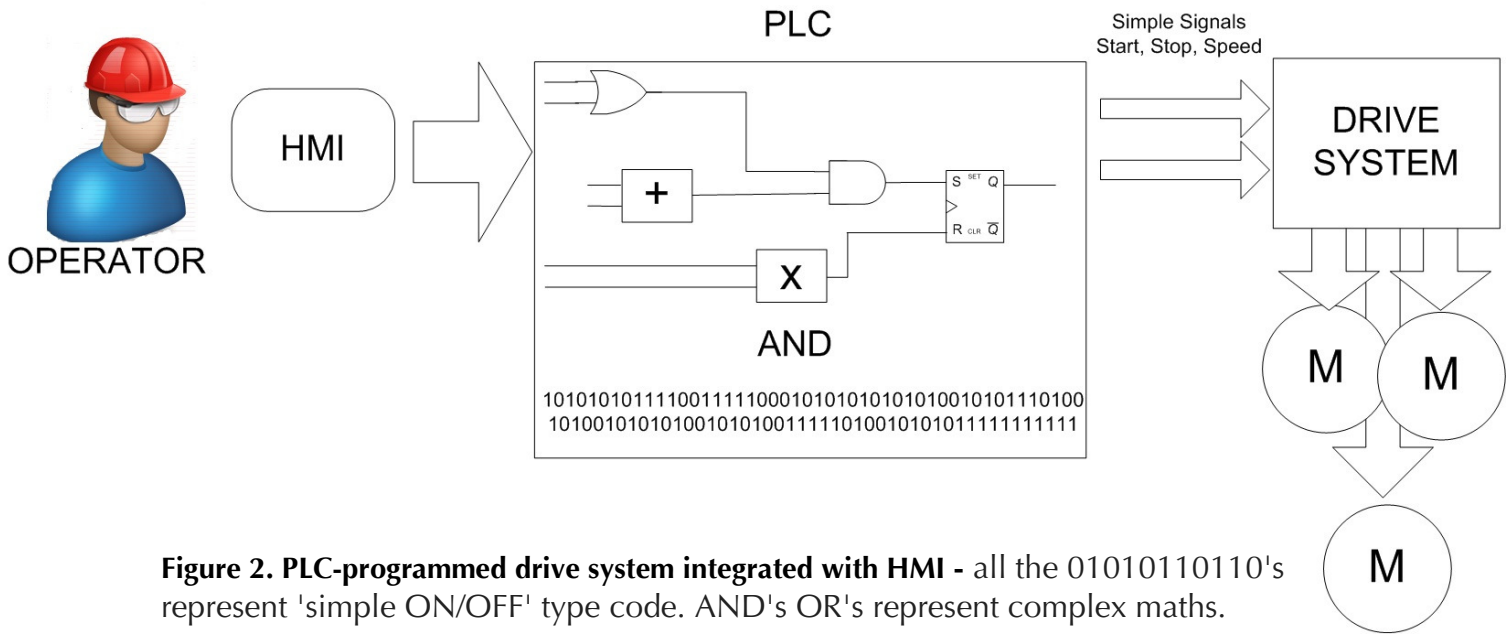

Figure 2. PLC-programmed drive system integrated with HMI - all the 01010110110's represent 'simple ON/OFF' type code. AND's OR's represent complex maths.

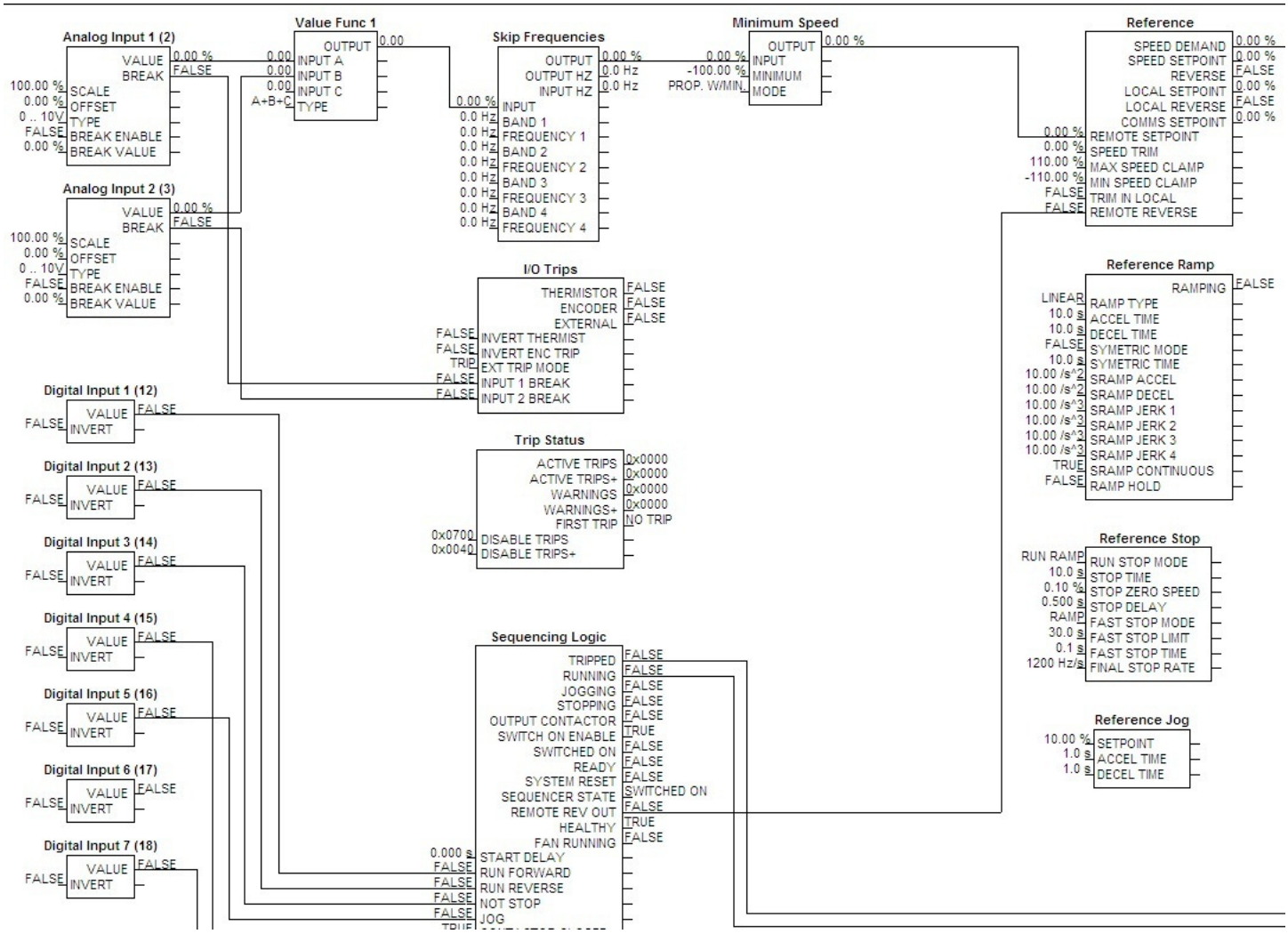

Figure 3. Typical drive programming language – the example depicts Parker SSD config. code

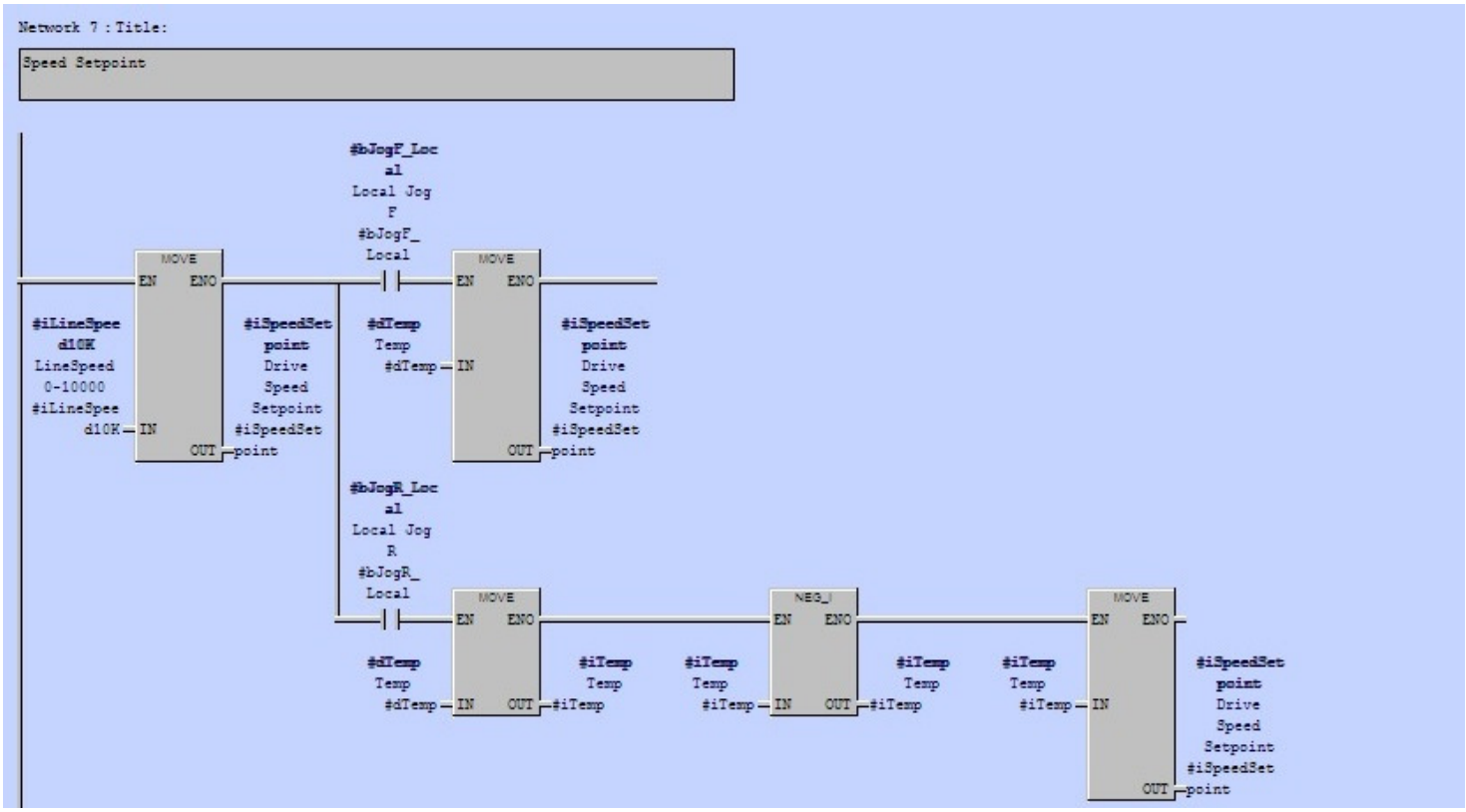

Figure 4. One 'line' of code from a Siemens PLC programming job. This network is calculating either a positive (forward) or negative (reverse) speed setpoint as a jog speed.

Which programming method to use?

The best control method is circumstantial.

OEMs can benefit from using the PLC resident software option as they can have a common PLC with the flexibility to offer any type of drive to their customers.

For **maintenance departments** it would depend on the expertise of their engineers. Having well documented and common function blocks (as in the drive resident software option) would offer standard, documented software solutions and the comfort of the drive manufacturers service and support resources. A "site standard drive" is required here, if different types of drives and PLCs are used on a site then effective maintenance is a more difficult task.

As an independent systems integrator, at Optima Control Solutions we have no drive/PLC manufacturer obligations to meet, we design systems and solutions for our clients and fully consider our customers' circumstances when determining which option to recommend.

-- END--

Images sourced from:
Optima™ gallery – all copyrights reserved.
Internet

Design:
Gimp, Picasa

Written by:
Colin Keating, Hristina Stefanova

Revised by:
Michael Hill