New approach for high performance motion control

Dipl-Ing. (FH), MBA, Ari Almqvist
Moog GmbH
EM-BB Division
Hanns-Klemm_Straße 28
71034, Böblingen
Phone +49 7031 622-0, Fax +49 7031 622-250
E-mail: aalmqvist@moog.de

ABSTRACT

Closed loop control of hydraulic or electric systems offers an excellent means of achieving precision movement with big loads and high forces. However, hydraulic servo drives in particular have a lot of anomalies, e.g. non linearities and low damping, which in some applications makes these systems demanding to control. In such cases advanced control features are required in order to get the specified performance out of the drive.

The new approach described in this paper is aimed at providing the industry with a standard tool for tackling such applications. Easy and fast application development, together with excellent performance has been the key factors in the described solution.

The end result is a servo control platform versatile enough to handle the simplest through to the most complex problems and which is applicable to electric drives as well as hydraulic drives.

KEYWORDS: Closed loop control, hydraulic and electric systems, motion control, development tool, advanced controls

1 INTRODUCTION

1.1 High Performance Motion Control

High performance motion control is taking on an increasingly important role in the development of modern production machines. Using motion control, usually one expects to control force or position (and in some cases derivatives of these measures) with a goal to achieve higher or the highest possible standards in the following areas of the machine or its end product:

- precision
- minimization of time used for the process
- flexibility
- controllability and monitoring

Additional features could be to save energy, material, etc.

Whatever the individual goal is, any investments made in closed loop or motion control needs either to save money or to improve the machine value. In fact many of the improvements to be found in modern machines is not possible without closed loop or motion control. This has lead to servo controls being accepted as a norm in many cases. A direct consequence of the increasing servo use is the need for machine builders to become familiar with servo technology. The new approach described here is strongly driven by this need in the industry.

1.2 Background

High performance motion control has been around for a while. Before the 80’s mainly analog systems were used. These offered already reasonable performance, but had some limitations in what kind of control algorithms could be used, as well as a limited flexibility to make changes. The switch of product delivered from a given machine often required totally new tuning of the system by a qualified systems engineer. Consequently significant losses in time and production were inevitable.

Digital systems were a significant step forward in this respect, but at the same time increased the complexity of system development. In may cases, application engineers needed to learn programming languages and understand the quantization and timing issues as well as digital communications. In most cases, the benefits of digital control were offset in part by the increased development time and cost of the system.

To marry these two worlds – ease of use and high flexibility - has been a key motivation during development of the new approach.

1.3 What is new?

The use of standard programming languages and libraries, like PLCOpen for motion commands and IEC61131 for system configuration are new features in the programming of high performance closed loop controls. Also being part of an automation alliance, Figure 1, is a new approach, making it possible for customers to choose devices from different sources whilst having a single programming (automation) environment. It is becoming increasingly important for machine builders, (especially of large machines) to reduce the number of development environments. This requirement been taken into account in the new concept design.

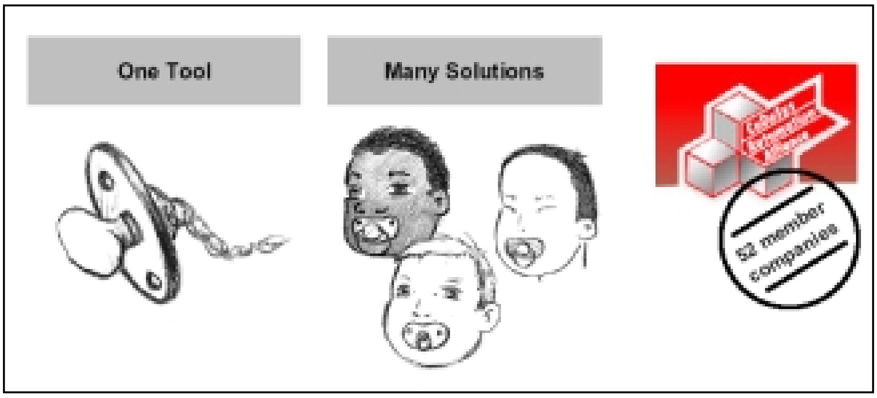

Figure 1: One tool for automation

2 HIGH PERFORMANCE MOTION CONTROL - CONCEPT

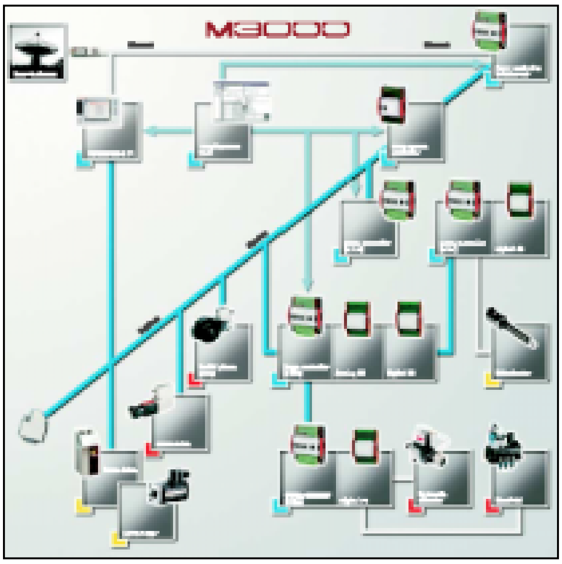

Figure 2: Automation concept M3000

A solution that takes care of all elements of motion and closed loop controls is the new concept, Figure 2. Whilst focused on high performance motion control and closed loop control, the concept includes the full functionality needed for machine control.

Starting from the actuation elements; it covers hydraulic drives/actuators, hydraulic pumps and electric drives/actuators. Next comes sequencing and PLC functionality. Extendable I/O via internal or external buses fulfills the requirements needed for medium sized machine controls. The next level of functionality is between devices, which can be based on fieldbus standards (like CAN, Profibus or ethernet). It is possible to go up to the factory communication level to get production data for the machine or diagnostics from the device and component level. Here ethernet is normally used.

One system development tool covers all of the above-mentioned functions from hardware configuration to system visualization.

3 SOFTWARE FUNCTIONALITY

3.1 Development of solution

A single development suite covers the development of solutions from the beginning and opening of a new project through to visualization of the running system. In between these beginning and end steps is a course of development, documentation and testing, which will shortly be described in following sections.

3.1.1 Project tool

A project environment is nowadays an important part of development. It includes all functionalities for project monitoring including;

- Opening
- saving
- version control
- documentation

This functionality helps development/application engineers to make complete projects faster and more efficiently. The approach described here in this paper has all of these features covered.

3.1.2 Hardware configuration

Tool shown in Figure 3 is used to configure the controller, as is typically done by an applications engineer, who knows the detailed functionality of the controller. With this part of the tool, inputs and outputs, sensor interfaces, fail safe options and additions/extensions to hardware will be defined. It offers all the functional options of the hardware without any physical changes (e.g. link settings).

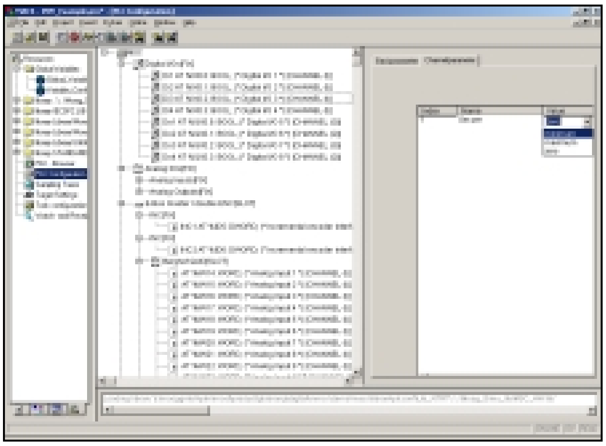

Figure 3: Hardware configuration tool

3.1.3 Developing closed loop controls

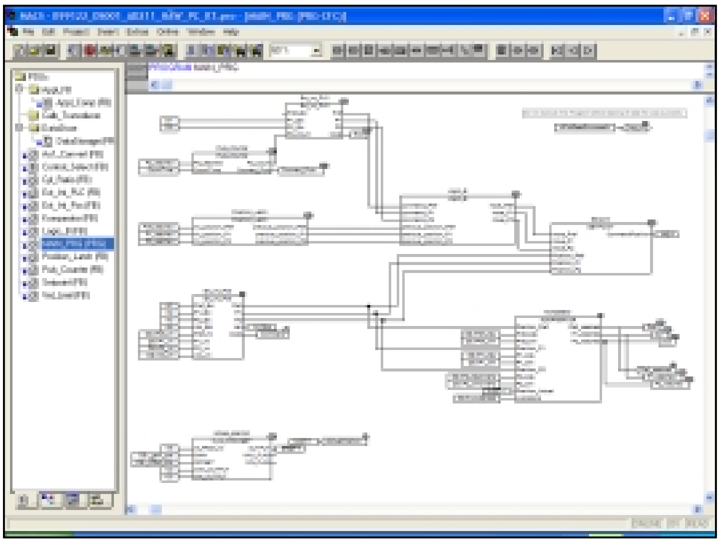

Figure 4: Graphical programming tool for control structures

Closed loop control is developed by using a graphical (object oriented) editor, Figure 4. This allows the developer to take function blocks, which are represented by related icons, and place them in the workspace. Here the functionality will be defined and interconnections wired. The interconnection of the blocks will define the data-flow in the embedded side of the controller. The control scheme can be changed any time and downloaded to the hardware. This makes it possible to make rapid changes, if needed, even at the customer site.

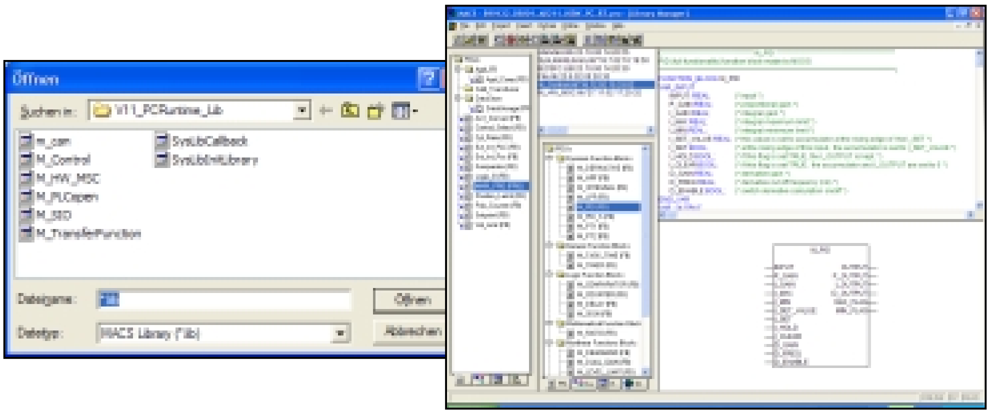

Figure 5: Libraries of pre tested function blocks

Solutions worked out this way in the field can be relied upon because all of the function blocks are pre-tested software modules, Figure 5. This leaves the application engineer to focus specifically on the task at hand rather than becoming embroiled in software debugging.

It is also possible to upload systems in the field and make changes. This is useful if the applications engineer is working with an old system, where the documentation is not always readily available. The hardware also has a storage system, where part of the critical documentation can be stored and retrieved.

3.1.4 Developing of machine sequencing

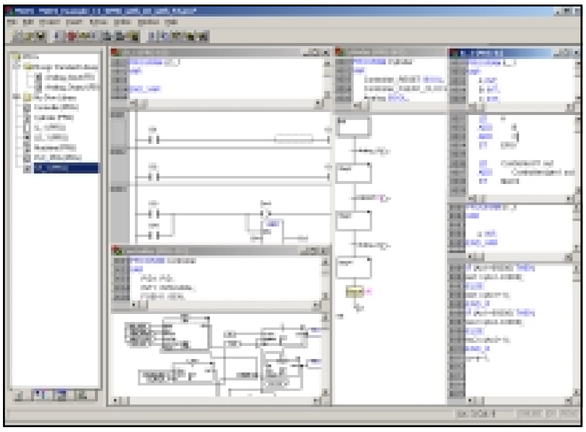

Figure 6: Selection of programming languages

Five languages, Figure 6, are available for programming machine sequences. One can select to use the ladder diagram, instruction list, structured text, function block diagram or the more structured (state machine) sequence flow chart, Figure 7, to develop sequencing solutions. Any mixture of these languages and all variables used in any other language can be made available for the entire development environment. All these IEC 61131 defined languages are very widely used in the automation industry. This high flexibility makes it possible for application engineers to use a preferred environment, which in turn leads to improved productivity.

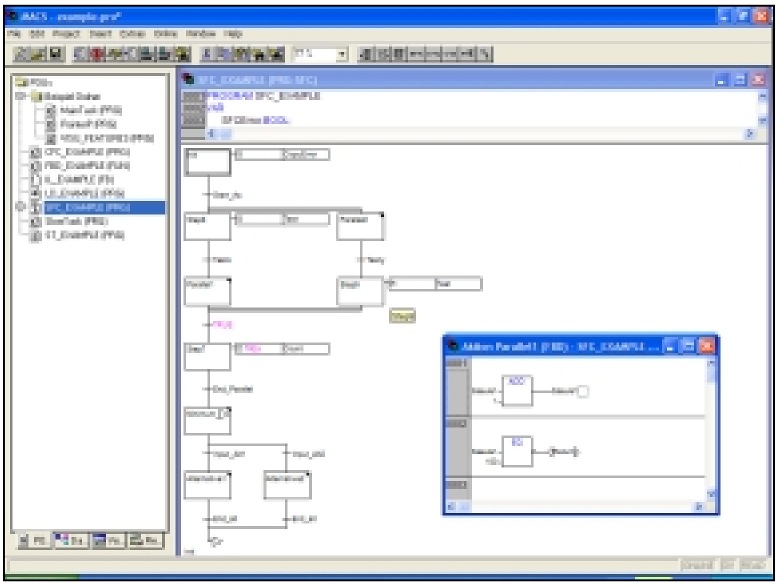

Figure 7: Sequencing flow chart (SFC)

3.1.5 Creating visualization

The development suite includes integrated visualization development tool and visualization software, which runs on a PC. This environment offers the functionality needed to operate typical industrial machines where visualizations will allow the project variables to be viewed, Figure 8. Using visualization functionality, the developer can draw geometrical elements offline, which can then change (e.g. their form or color) online in response to specified values or variables. For example it is possible to display the increase in a variable value with a bar chart.

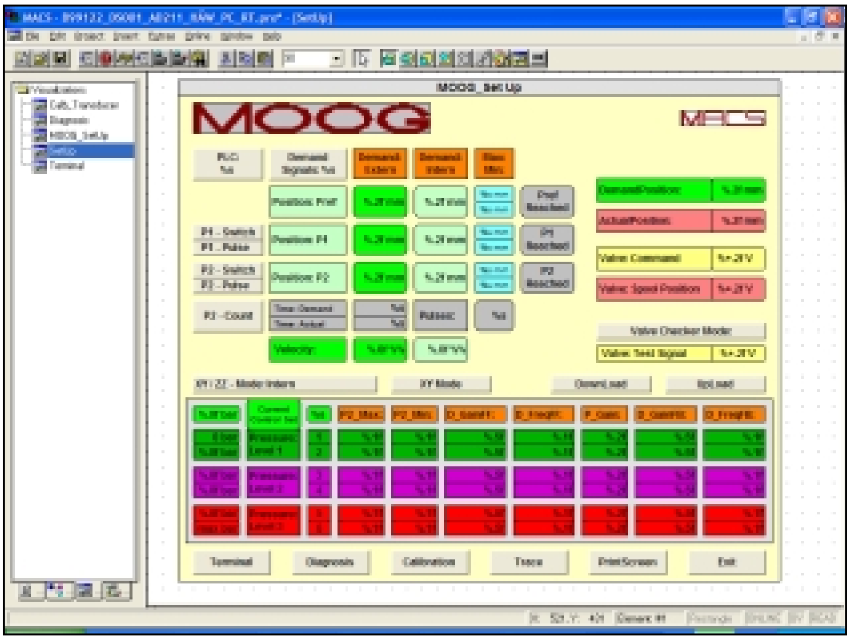

Figure 8: Integrated visualization

In Online mode, the operator can see and input variables for the system via a mouse and keyboard. In this mode the operator is able to change the parameter set, for example to change the product to be produced.

3.1.6 System testing and debugging

When a solution has been developed it is important to undergo system testing prior to full implementation. The new approach offers quite extensive possibilities to test the system, even without hardware. The first possibility is to use simulation of the hardware, where the development software emulates real hardware and all the functionality and software integrity can be tested off line. When offline testing is finished the developer can download software to the hardware and start testing it step by step. Single step, breakpoints, visualization of active functionality and more are available to offer easy to use and effective system testing and debugging.

3.1.7 Documentation

The development suite also includes project documentation features. For example the project compare functionality. Here it is possible to compare two projects to see what changes have been made between the current and previous versions. Another feature is to print a document according to the selection of the developer, where all the key information of the project can be printed and used as final documentation for the project.

3.2 Embedded software

The Embedded/controller side consists of a number of different hardware drivers, a boot-loader, and a simple set-up communication link in its firmware. In addition, a real time OS, hardware function blocks, and a communication protocol can be downloaded from the PC as EMB software. This downloaded code is viewed as part of the embedded software since that is where it will execute. Significant time has been given over to the embedded software design, as this base is considered as a foundation that will be used for many years to come.

3.3 Interfaces

3.3.1 Standardized Windows interfaces

Standard software interfaces, like OPC and DDE are available to support seamless integration to other automation software, like SCADA tools, which can be used for visualization of the machine, instead of the integrated visualization. These interfaces make any parameter exchange easy and safe to implement.

3.3.2 Serial interfaces

Two serial interfaces, based on RS232 are available for system development and any communication to a PC or other devices.

3.3.3 Fieldbus interface

CAN is used as the standard interface between devices or remote I/O's or as a real time link between host and motion controller. Profibus will be an option in the near future for fieldbus communications.

3.3.4 Ethernet

Higher-level communications are divided between fieldbus communication as well as factory and device level communication, which are based on ethernet. The first version of the hardware has two CAN busses, two RS232 serial interfaces and 10 Mbit/s ethernet with TCP/IP protocol.

4 HARDWARE

4.1 Modularity

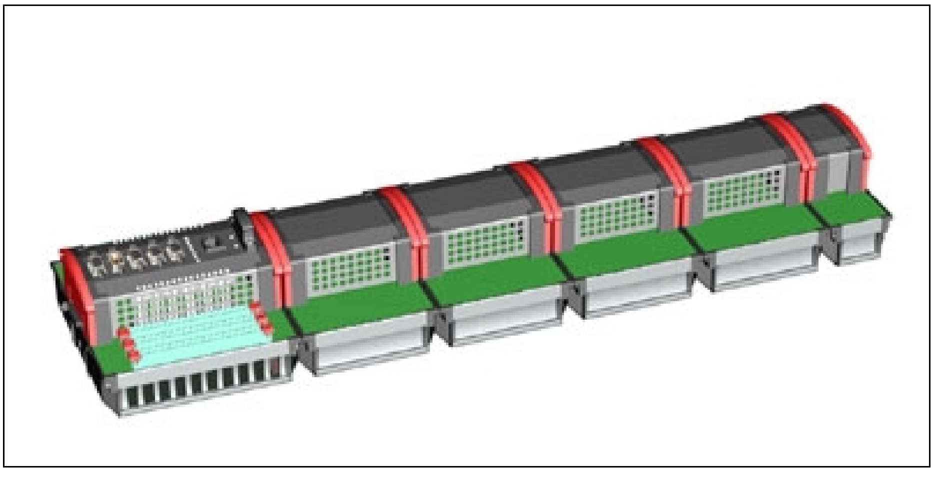

Figure 9: Modular hardware

A modular hardware shown in Figure 9, based on a high level of connectivity between modules is one of the key features of this new approach. At the heart of the system is the integrated closed loop controller, which is a stand alone multi axis control module with enough performance to handle the most complex control loops with a fast sample frequency. This system includes all the needed hardware to realize effective, modular high performance motion control systems, including power supply, inputs/outputs, sensor interfaces, serial interfaces, fieldbusses and ethernet link.

4.2 Interfaces

Systems for automation without effective communication are nowadays out of the question. More and more information at a higher and higher level and sometime with all the small details included are required. That is why this new approach has been developed to also take care of these factors.

4.2.1 Analog

Flexible 16 bit inputs and outputs (6 in/2out) are available and can be extended through extension modules. All this is software configurable to make it as generic and user friendly as possible. Also fast sample times with analog interfaces have been taken into account to make sure system performance is optimized to the highest possible level.

4.2.2 Digital

The basic controller has 8 digital inputs or outputs. These are also software configurable. The system can be extended with additional digital input/output modules through the internal bus or fieldbus.

4.2.3 Sensor

Two channels of sensor interface, either encoder or SSI are available. These channels can read very high frequencies to make for fast and precise machine control.

4.2.4 Serial/Fieldbus/Ethernet

Serial, fieldbus and etherent interfaces are available as described in the software description earlier (3.3.2 to 3.3.4)

4.3 Monitoring and Fail Safe Functions

The system has features designed with machine safety in mind. For example, power supply monitoring, cable monitoring, a hardware watchdog, and definition of inputs and outputs whenever the system recognizes a problem. These and some other details inbuilt in the system may sound simple enough, but these features are the key to preventing damage, either to operators or to the machine itself. This is an area where years of experience count!

5 APPLYING THE PRODUCT IN PRACTICE

5.1 Use of Simulation

In many cases to develop a machine with high performance at reasonable cost is nowadays getting more and more difficult. Simulation is becoming a key factor in the development of new solutions. This new approach is supporting and helping this trend to proceed making it more available for the typical applications engineer, who is not normally a software guru.

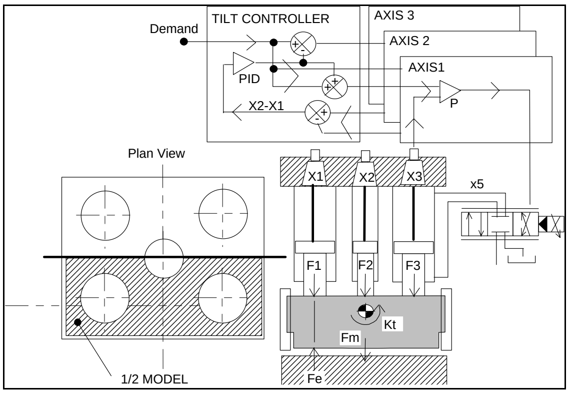

Figure 10: Typical simulation model of hydraulic system

Developed models, like shown in Figure 10, using simulation tools are easily changed into practical application with industry standard tools, like IEC61131. In this described approach it is easy to use graphical elements to describe a model and even in the case of some tools it is possible to use code generated by the simulation tool as function block or executable code.

Using this approach it is possible to take a model from the simulation system and easily transfer this to software, which can be used to control the machine. Important also is that this is done with hardware and software at an affordable price for normal industrial use.

5.2 Application example

An application example is shown here to give a practical feeling about what it means to use the described method for a real life application.

5.2.1 Control structure

In the example described here, the problem statement is typical for hydraulic systems. The system condition and parameters are changing, depending on the operating conditions. Also the mode of operation, means that the machine needs to do changes, depending what product is going to be made or what part of the process the machine is doing. Using the M3000 approach it is easy to create specific control structures, Figure 11, to take care of these influences and even change structures or parameters online depending on the system requirements.

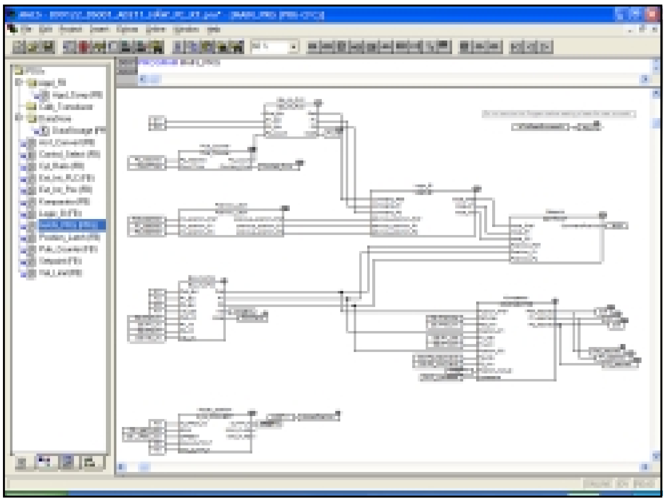

Figure 11: Control structure developed with MACS software environment

If required, the system can be changed easily during commissioning, which especially in the development phase could save a lot of travel, communication and time. In the end this equates to money saved.

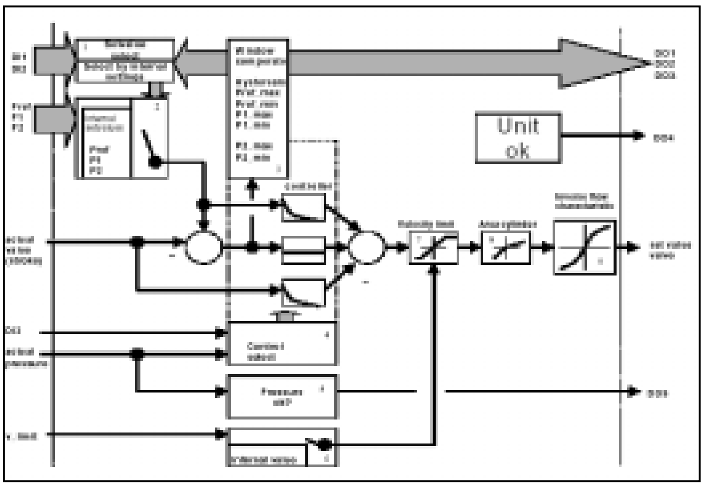

Figure 12: Control structure block diagram of example system

To improve system performance special characteristics of the servo valve are used (e.g. dual gain), which in other ways has non-linear effects for the controls. These effects, as an example, can be compensated by the system in a comfortable manner.

5.2.2 Products used in this application

In this case products used in the machine are, MACS development suite including visualization, MSC controller, I/O extension modules, servo cartridge valve and a radial piston pump, see Figure 13.

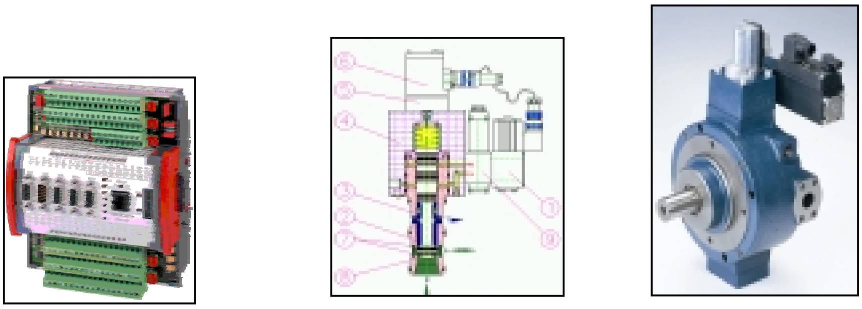

Figure 13: Products used in application example

5.2.3 Logical functionality (sequencing)

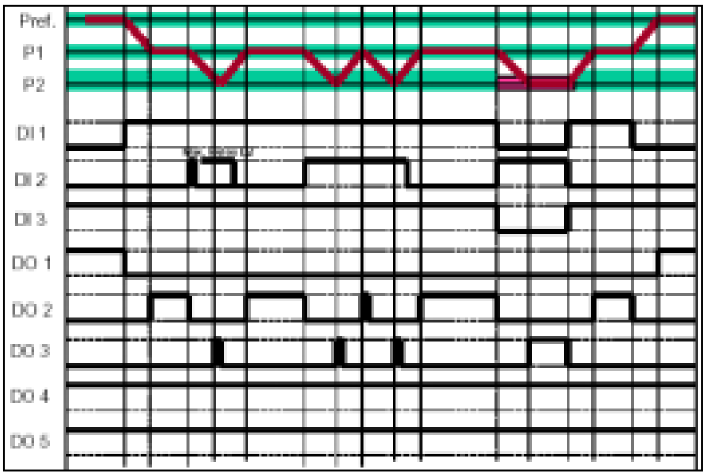

Figure 14: Sequence chart for example application

Logical functions of the machine and sequencing of the machine are programmed using any of the IEC61131 languages, Figure 14. As mentioned before, the programmer is able to select any of 5 available languages, from simple ladder diagrams to the functional object orientated programming language. Example shown in Figure 15, where function block diagram is used.

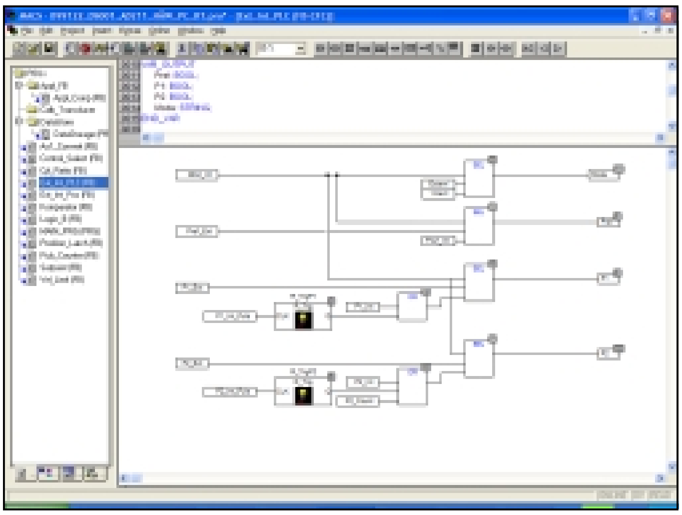

Figure 15: Function block diagram for machine sequencing and logics

5.2.4 Visualization, HMI

In the end there needs to be tools so that engineering can develop a new system and also the user can use and make required changes for the system. All this is included in the M3000 suite of software and hardware – one environment many possibilities.

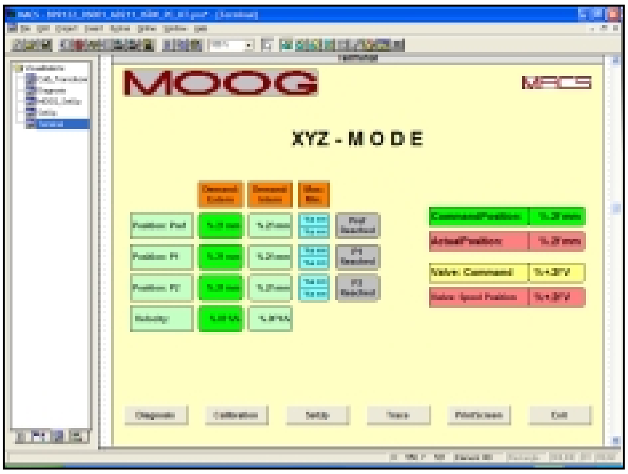

Figure 16: Machine visualization, HMI

5.2.5 Result of the project

As a result, what counts is how the machine is performing. Is the closed loop response fast enough and without overshoot? The laws of physics still describes the limit of what can be done, but modern control technology can nowadays do quite a bit more to get the best possible performance. For example see the machine response data in certain operation conditions.

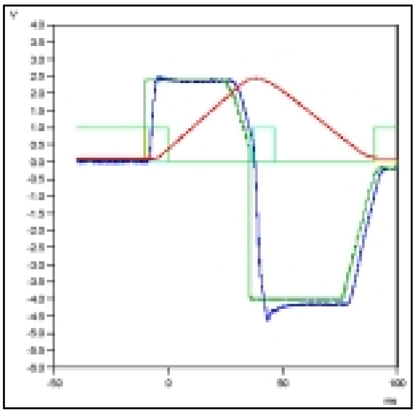

Figure 17: Example of command and actual response of the system

In this case the flexibility that the machine builder was looking for was reached and compared to previous systems additional flexibility of the system level was achieved. In this case the machine builder acquired the development tool and was able to make any changes needed by himself. In many cases a machine builder can develop systems with enough control knowledge.

5.3 Advantages

Use of standards which are widely accepted, makes it easy to learn and use the software tools. Instead the focus can be on the machine or output product optimization. Here, high performance closed loop controls, the environment and tools, ensures a quality solution.

Use of cooperation and alliances that fit industry requirements, helps to provide a scope of supply and tools that make it easier to develop machine control functionality. One tool for everything is a direction that everybody wants to go in. Software is getting more and more rich with functionality and features and to be able to use them effectively needs more and more training. All together, to have an easy to use, high performance package will give value for the machine and help industry to develop itself.

6 CONCLUSIONS

With this development it has been shown that it is possible to develop a new system by using the available technology, a technology that uses a standard environment and produces state of the art close loop control performance. It is also proven, that one system can be used for multiple tasks, which have different requirements running in one environment.

On the one hand high performance complex control algorithms are allowed and on the other hand relatively slow PLC type sequencing tasks, which are complex due to the multitude and structure.

By selecting the right hardware and software architectures it can be realized cost effectively. By using the platform strategy and developing a layered software concept it has also been made portable for multiple products and applications.

7 REFERENCES

Moog Inc. – Technical Bulletin TB126 - 126 - Performance Estimation for Electrohydraulic Control Systems – www.servovalve.com

Moog Inc. - Practical Considerations When Laying Out Electro-Hydraulic Control Systems – www.servovalve.com

I M Whiting, BSc(Hons), PhD and N P Cottell, CEng, MIMechE - Moog Controls Limited, Tewkesbury, Glos, UK - PROGRAMMABLE SERVO CONTROL - A NEW APPROACH [1] – www.servovalve.com

Moog Inc., Closed Loop Motion Control Machine Control Technologies: Improving Control of the Extrusion Blow-Molding Process, March 2000, www.servovalve.com

Dipl.-Ing. Walter Lenz, MOOG GmbH, Developments in High Performance Proportional Valves with CANopen Fieldbus Interface

8 DEFINITIONS, ACRONYMS, ABBREVIATIONS

OPC: OLE for Process Control

DDE: Dynamic Data Exchange

IEC61131-3: The standard IEC 61131-3 is an international standard for programming languages of Programmable Logic Controllers

SCADA: Supervisory Control and Data Acquisition

FLASH: = non-volatile, in-system reprogrammable memory

PLC: Programmable Logic Controller

M3000: Moog motion control concept

MACS: Moog axis control software

MSC: Moog servo controller