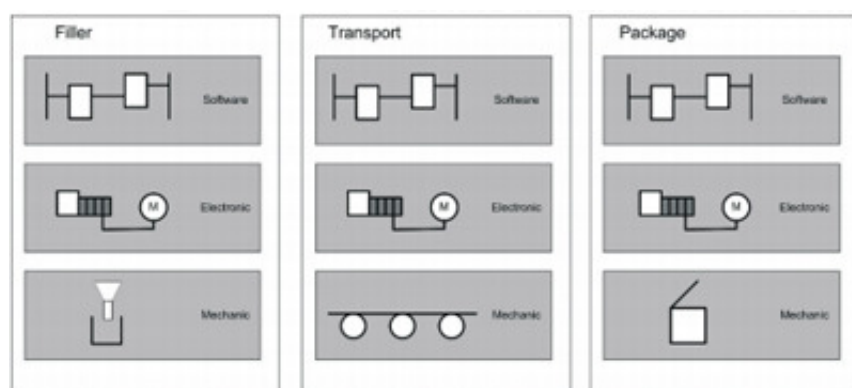


## TECHNICAL ARTICLE: DISTRIBUTED CONTROL PROVIDES PLUG-AND-PLAY FUNCTION BLOCKS

***Mechatronics with distributed processing provides a modern object-oriented application that can help machine builders to reduce costs and give more flexibility in building customer-specific products. For the efficient realisation of creating mechatronic objects - functional assemblies that can slot together to make new systems - the communication, control system and engineering tools must all be optimised for distributed control. SafetyNET ticks all the boxes says Ralf Möbus of Safety Network International***

MACHINE BUILDERS often find themselves in a contradictory situation. On the one hand they have to react flexibly to customer needs with customised machines while on the other, product standardisation reduces costs and engineering effort. Modular machinery design has the potential to serve both aims. This design approach is called mechatronic modularisation. It describes the building of machine modules that include mechanics, electronics and software brought together as a functional block. The modules are only described to the outside world by their interface with which they are connected to each other. Other modules don't need to know about the functions that take place inside the module.

The mechatronic approach cannot be adequately implemented with existing centralised control systems. It requires a different paradigm with a distributed control philosophy aimed squarely at the modularisation of machinery.



**Fig. 1. Mechatronic modules encapsulate software, electronics and mechanics**

With distributed control a fully mechatronic modularisation of machinery is possible where modules can act almost independently from the rest of the machinery. Such an approach offers several advantages for machine builders. If assembled machinery can be broken down into functionally independent modules, then the same modules can always be designed identically - with module design made independent of different module combinations in the complete assembly. Thus if machine builders offer different machine assemblies to individual customers, the modules which make up the machines don't need to be changed. This allows for a high degree of standardisation where costs can be reduced through the stocking of fewer machine variants.

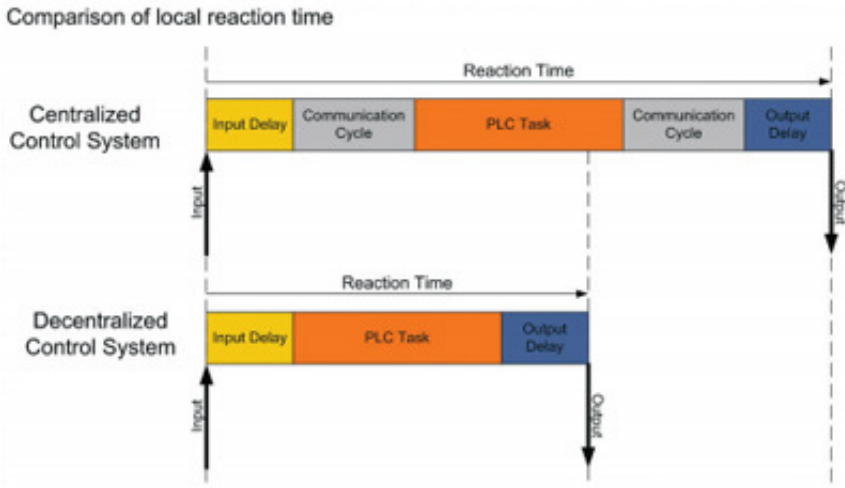
Mechatronic modules can be run up and tested independently of the complete machine assembly permitting functional test at an early stage. Early detection of errors reduces the testing load for the whole assembly.

### Software encapsulation

The supplier of a machine module can encapsulate relevant software into a module such that it will deliver plug and play startup. Integration with and adaptation to other module software - perhaps from other suppliers - in the central PLC is not needed so reducing the effort for system integrators. It also enables module suppliers to protect their know how because the software is already in the module PLC on delivery to customers. If mechanical, electrical and software interfaces are identical, then the exchange of a machine module for that of different function is straight forward - the old module needs only to be replaced by the new module.

### Time-critical process signals

Distributed controllers offer faster local processing of time-critical process signals since local controllers can run their own application program. When a local reaction to a process signal is required, the information stays local and does not need to be transferred to a central PLC via the fieldbus. The possibility to process signals locally on the hardware without additional communication delay can reduce reaction times. An additional advantage is that latency retains independence from communication load on the network thus being more predictable.



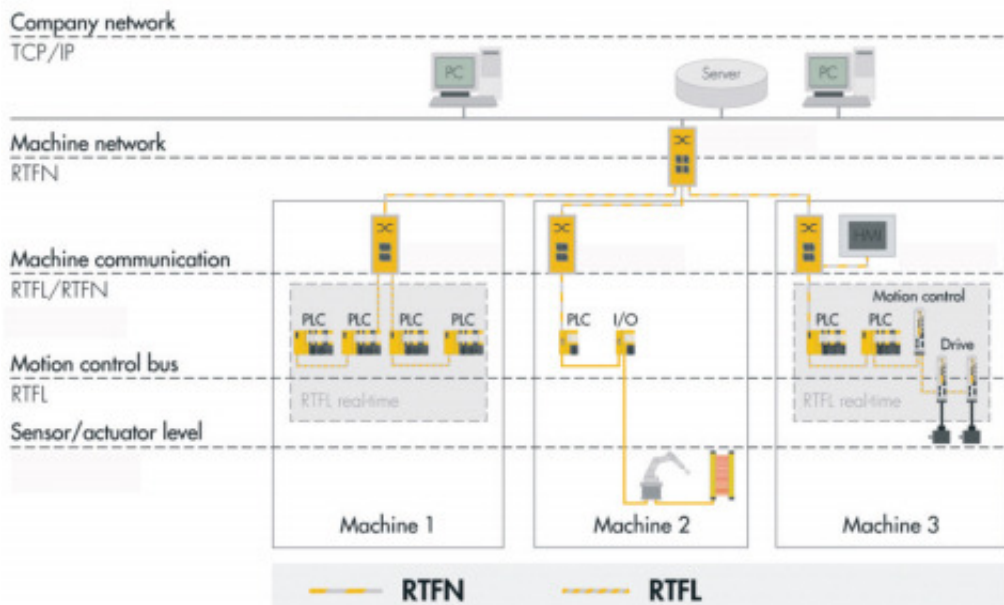
**Fig. 2. Local processing reduces local reaction time**

The failure of a single controller in a distributed control system generally does not affect the whole system as is the case with centralised control: this enables robust distributed systems to be constructed. The availability can additionally be increased by network redundancy functions, for example with ring redundancy schemes. Furthermore a central Master CPU could possibly be designed out of the system altogether lowering the cost of hardware.

**Role for SafetyNET p (SNp)**

The communication system is of prime concern to a distributed control system and should fit a decentralised approach. As the control logic is distributed in the machine assembly and not in a centralised controller the usual master-slave fieldbus systems are not an appropriate solution: such a system requires the complete communication needs to be managed by the master. This is not efficient for a distributed system. Taking this further, intelligent mechatronic modules need direct communication relationships. Also, in a master slave system, the master needs to know about the complete control system and all the network subscribers. For a mechatronic module application, the software in the master would always have to be changed depending on the combination of machine modules. This of course means that plug-and-play standardisation of software would therefore not be possible.

The SNp communication system does not have a centralised master; it uses the so called producer/consumer principle. In such networks every device has the same communication rights: every device can publish data directly to the network. Devices communicate directly with each other, a prime aspect for a distributed control system. Since SNp is based on 100Mbps Ethernet it has enough bandwidth to fulfil the communication needs of automation devices and a number of fieldbuses can be reduced to a single Ethernet cable.



**Fig. 3. SafetyNET p - Ethernet based communication system for distributed control**

To meet the different realtime requirements at different automation devices, two protocols exist. The RTFN protocol uses standard UDP/IP or MAC frames. Therefore standard Ethernet infrastructure such as network interfaces, switches and routers can be used. The RTFL protocol is based on specific hardware and is suitable for highly deterministic communication with cycle times of up to 62.5µs permitting use inside motion control systems.

**Engineering tools**

To reduce the effort required for distributed control system configuration and management Snp's principal benefactor Pilz has designed the PAS (Pilz Automation Suite) engineering tool for the PSS 4000 system. It allows the system to be programmed in IEC 61131 languages and its use is intended to cover the whole machine life cycle. It gives the user a central view of the distributed system.

During programming the user uses symbolic variables in programs without concern about on which hardware the physical IOs exist. The mapping of the variables to the physical PLC and IOs is done at the end of the programming through drag and drop. The required communication relations between the distributed PLCs are setup by the tool in the back ground.

In a decentralised control system the configuration needs to be downloaded to the participating distributed controllers. In principle this requires more effort than would be the case for single controller architecture. But with intelligent tool support this effort can be reduced.

In PAS the download is done during the deployment phase. The information on which hardware the software shall run comes from the user through the act of dragging and dropping software modules onto the hardware resources. In the production and maintenance phase of use, PAS enables a central view of the distributed diagnostic information. Additional diagnosis can also be made available to machine visualisations and control rooms via the standardised OPC interface.

**Ralf Möbus** is with *Safety Network International e.V.*