iCC 1996

3rd international CAN Conference

in Paris (France)

Sponsored by

Motorola Semiconductor
National Semiconductor
Philips Semiconductors

Organized by

CAN in Automation (CiA)
international users and manufacturers group
Am Weichselgarten 26
D-91058 Erlangen
Phone +49-9131-69086-0
Fax +49-9131-69086-79
Email:headquarters@can-cia.de
URL: http://www.can-cia.de

Dr.-Ing. L. Rauchhaupt, Dipl.-Ing. H. Schultze

Otto-von-Guericke-Universität Magdeburg
Institut für Prozeßmeßtechnik und Elektronik (IPE)

CAN in Industrial Fluidic Systems

As in other branches a central automation concept is the state of the art in industrial fluidic systems. When one wishes an open communication with CAN the problem arises to choose between several major Higher Layer Standards, such as CAL, CANKingdom, DeviceNet or SDS.

In order to make open communication more efficient the idea is to have a Higher Layer independent API for fluidic applications (profile). Due to several Higher Layer Standards for CAN it is advantageous to divide such an API into a Higher Layer determined part and an application determined part. A proposal will be given in the presentation.

We will show, that an open communication which fits the needs of fluidic systems causes a change of its automation structure. It is determined by decentralized devices with processing capabilities (e.g. device configuration and self diagnosis) which can fit to a special application by IEC 1131 compliant configuration facilities.

Automation Concept in Fluidic Systems

Analyzing the communication demands of the sensor/actuator level of industrial fluidic systems the conclusions can be summarized as follows:

- the wish for open communication in fluidic systems depends on whether they are for series machine building or for equipment construction; there is no interest in open systems interconnection for series machines
- no special bus system is preferred; depending on the application and customers demands the use of CAN, INTERBUS-S or PROFIBUS is possible; within those bus systems different protocol versions are known /1/
- besides the wish for open communication it always will be necessary to use vendor specific know-how without making it public

Possible system structures of fluidic systems are shown in Figure 1 by an example of a hydraulic drive /2/. Principally two variants can be differed when planning bus systems. The first possibility is to close the control loop via the bus. That means the control algorithm is executed by a master controller. When using this variant the bus suffers a high load of real time process data, because there are at least two data transmissions per control loop in every sampling cycle. With a growing number of bus nodes the demands to the bus system and to the cycle speed of the master controller, mostly a PLC, are increasing (see structure 1 and structure 3).

The performance requirements to the bus system and the master controller can be considerable reduced by the use of a distributed control (see structure 2). This presupposes intelligent bus nodes which only need to be configured and set with set values.

The bus load can also be decreased for instance by transmitting the sensor signals on a conventional way to the controller. This should only be a temporary solution as long as there are no bus capable

Figure 1: System Structures With Hydraulic Drives Using a Field Bus System

Based on this considerations it can be derived that all shown structures will have their applications in future, too. Fast control tasks, such as valve position control or piston position control for drives with a high process dynamic should be realized decentralized. In comparison with this control loops with lower time requirements or the acquisition of auxiliary information can be done by the master controller.

Communication Model of Hydraulic Devices

Because there was no decision about the use of a certain protocol or bus system in our project we started with a common approach to describe the structure and the behavior of hydraulic devices in open systems interconnection. The main input to the model description came from the communication demands of hydraulic systems. But it considers the definitions specified for example in DRIVECOM (INTERBUS-S) or CANopen, too.

Fehler! Kein Thema angegeben.

Figure 2: Common Structure of a Logical Device

In fluidic systems there is a number of bus participants necessary, which realize simple input/output functionality of binary or analog signals. Thereto belongs the acquisition of flow, pressure, temperature, position and others. The properties which are relevant for the communication of those devices are already described for instance for CANopen in the Device Profile for I/O Modules or in the Function Models of Smart Distributed System (SDS).

It was a basic task to find universal valid function blocks for fluid-specific devices, which can be used to describe sufficing more or less complex fluidic devices. It became very important not to group the vendor specific parameters in a way that would influence the generic description inadmissible.

The concept of modeling is based on a description of a virtual device, which contains several function blocks as shown in Figure 2. The device can be described by parameters for identification, configuration and diagnosis according Table 1.

	explanation		
identification			
device address	address of bus participant	rw[1]	mandatory
device functions	list of communication addressable function blocks of the device	c	mandatory
vendor name	short vendor name in characters	c	optional
device type	vendor specific device name	c	optional
serial number	vendor specific serial number	c	optional
version number	vendor specific version number	c	optional
command			
enable device	enable/disable execution of functions blocks	wo	optional
enable hardware I/O	enable/disable the inputs/outputs of the device.	wo	optional
state			
device state	present state of the device	ro	mandatory
device error	current errors of the device	ro	mandatory

Table 1: Parameters of a Logical Device

The description of the function blocks follows the principle of heritage. First the generic parameters are described and then completed according to the demands of special parameters. A typical example in the area of fluidics is a continuous-action controller with set value call-up. The function model is shown in Figure 3. The connections in the picture are no signals in the sense of the functions models according to IEC 1131-3 but show the parameters relevant to the communication.

Besides the set value, the actual value and the control value the model contains so called ramp parameters. These configure the ramp generator, which smoothly (ramp-like) adjusts the old and the new set value. In addition 16 sets of set values with belonging ramp parameters can be configured in the bus node. The set value sets can be called during operation mode with a single command. Besides the named parameters control and status words for the function blocks are defined.

In coherence with the current project the models for analog and digital inputs/outputs, for continuous-action controllers, for continuous-action controllers with set value call-up and PID controllers are described. So far it is to foresee at the moment all demands of hydraulics can be satisfied.

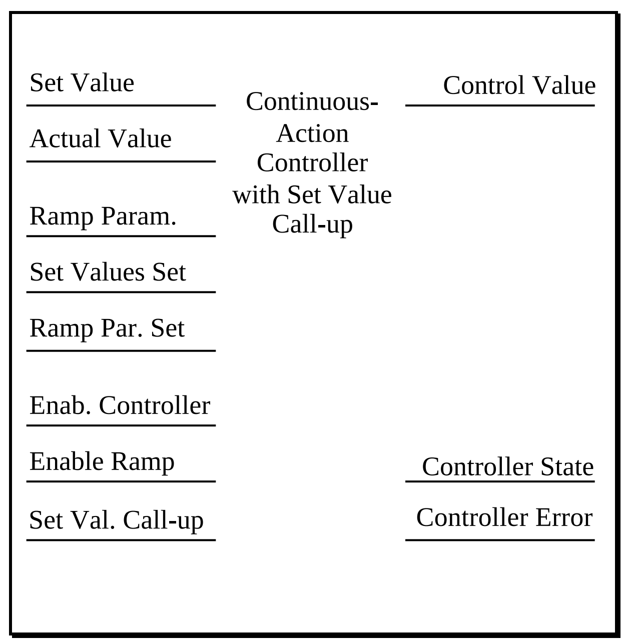

Figure 3: Function Model of a Continuous-Action Controller With Set Value Call-up

signals	explanation		
process data			
set value	Set value transmission is an essential service. The set value can read back via the bus, too.	rw	mandatory
actual value	It is only allowed to read back the actual value via the bus because loops which are closed via the bus will not be used.	ro	optional
control value	If the controller is enabled it sets the control value directly. The control value is only allowed to read back using the bus. It is possible to read and write the control value if the controller is disabled.	rw	optional
parameters			
ramp parameters	The ramp parameter set specifies how a change of set values should be processed. All parameters are allowed to read back. A set consist of parameters for ramp type, ramp time and special ramp parameters.	rw	optional
call-up parameters	***field of max. 16 parameter sets***		
set values set	This is a field of set values which become valid when called-up accordingly.	rw	mandatory
ramp parameters set	This is a field of ramp parameters which become valid when called-up accordingly.	rw	optional
commands			
enable controller	This command enables the controller.	wo	optional
enable ramp	This command specifies whether the ramp parameters should be used.	wo	optional
set value call-up	A complete parameter set is fetched into the controller.	wo	mandatory
state			
controller state	This value informs about the present state of the controller.	ro	mandatory
controller error	This value informs about current errors of the controller.	ro	mandatory

Table 2: Parameters of a Continuous-Action Controller With Set Value Call-up

Application of the Models

The usage of the models is discussed briefly using an example of a continuous-action controller. Starting point is a bus-capable module (similar structure 2 in Figure 1), which controls flow and pressure of a hydraulic drive. The application interface has the structure shown in Figure 4. The set values for flow and pressure are set by a PLC using the bus. Because of the time requirements the actual values are acquired locally. For the purpose of visualization and diagnosis the actual values and control values are reachable using the bus. For the transmission of the actual flow value an additional analog input is provided in order to use the advantages of event-controlled transmission. Therefore trigger events can be selected. The actual value is sent when these events occur and the trigger is enabled.

Whole travel profiles (with different acceleration and retard phases) can be configured in the bus node with the help of the parameter sets. They can be called-up during the operation mode. Flow set values and pressure set values can be called-up either independent or at the same time.

Figure 4: Function Blocks of a Flow/Pressure Controller

For testing and commissioning of hydraulic control devices a PC based program for Windows was created at our institute. At this the protocol independence becomes very clear. For instance the windows for the configuration and operation of a continuous-action controller with set value call-up (Figure 5 and Figure 6) have no relation to the used protocol. Only the parameters of the according model (Figure 3) can be recognized.

Figure 5: Control Setup Dialog

After formal modeling and describing of a protocol independent interface the usability should be proved by implementing on several Higher Layer Protocols. In Table 4 a proposal for the definition of a set value call-up controller for a CAL based device profile is given. It is to decide whether the already existing Device Profile for Drives should be enhanced with fluidic objects or if it is better to create a new device profile. For the integration into the existing Drive Profile stands that a part of the objects is defined already. On the other hand a device profile should not be to broad because of its oversight.

Figure 6: Control Process Window

Table 3 shows the proposal of object models for Smart Distributed System. According to the structure of such a model attributes, actions, and events are defined. During the application of the models the basic differences of both the Higher Layer Protocols became visible. CANopen prefers a broad specification, which covers mostly any thinkable causes in advance by any combination of the objects. The main part of the objects is optional. In opposition to this SDS is much more compact. Here one has to deal with a clearly outlined performance range. Insofar Smart Distributed Systems meets the needs of the modeling described above. Therefore a first realization was done with SDS. It was tested successfully in a hydraulic test environment. At the moment the authors work at the exemplary implementation of the CAL based profiles.

Attribut Set

ID	Name	Datentyp	r/w
...			
18	Set Value	signed 16	rw
19	Actual Value	signed 16	ro
20	Control Value	signed 16	rw
...			
57	Controller State	unsigned 8	ro
58	Controller Error	unsigned 8	ro
...			
61	Controller Configuration	unsigned 8	ro
62	Ramp Time	unsigned 16	rw
63	Ramp Type	unsigned 8	rw
64	Ramp Parameter 1	signed 16	rw
65	Ramp Parameter 2	signed 16	rw
...			
80	Number of Set Value Sets	unsigned 8	rw
81	Set Value Set 0		rw
...			
96	Set Value Set 15	characters[9]	rw

Action Set

ID	Name
...	
54	Enable Controller
55	Set Value Call-up
...	

Event Set

ID	Name
...	
51	Controller Error
52	Set Value Reached
...	

Table 3: Proposal of an SDS Embedded Function Block Definition for fluidic Devices (Selection)

Index [Subindex]	Objekt	Name	Type	Attr.	M/O
Functiongroup Pressure Controller with Set Value Call-up					
6202	VAR	Controller State p	unsigned integer 8	ro	M
6204	VAR	Controller Error p	unsigned integer 8	ro	M
6210	VAR	Set Value p	signed integer 16	rw	M
6220	VAR	Actual Value p	signed integer 16	ro	O
6230	VAR	Control Value p	signed integer 16	rw	O
6240	VAR	Ramp Time p	unsigned integer 16	rw	O
6241	VAR	Ramp Type p	unsigned 8	rw	O
6242	VAR	Ramp Parameter 1 p	signed integer 16	rw	O
6243	VAR	Ramp Parameter 2 p	signed integer 16	rw	O
6250	VAR	Number of Set Value Sets p	unsigned 8	rw	M
6255	VAR	Set Value Call-up p	unsigned 8	rw	M
626x	VAR	Set Value x p	signed integer 16	rw	M
627x	VAR	Ramp Time x p	unsigned integer 16	rw	O
628x	VAR	Ramp Type x p	unsigned 8	rw	O
629x	VAR	Ramp Parameter x.1 p	signed integer 16	rw	O
62Ax	VAR	Ramp Parameter x.2 p	signed integer 16	rw	O
Functiongroup Flow Controller with Set Value Call-up					
6302	VAR	Controller State Q	unsigned integer 16	ro	M
6204	VAR	...	...	...	...
Functiongroup Position Controller with Set Value Call-up					
6402	VAR	Controller State s	unsigned integer 16	ro	M
6404	VAR	...	...	...	...
		x=0..F			

Table 4: Proposal of CAL based Object Definitions for fluidic Devices (Selection)

conclusions

During the implementation of the models it became clear, that it makes a difference whether available fluidic devices are made communication capable or if fluidic devices are conceived bus capable from the start. In the first case it is very difficult to setup the communication according to a generic model. This probably will be the exception in the future.

For the modeling requirements regarding time behavior, the synchronization of objects and so on do not play a role. This behavior is mainly determined by the protocol and the profiles used for implementation. But it is to say that often economic reasons respectively marketing reasons dictate the decision and not technical aspects when choosing a protocol.

The shown way of describing the communication requirements was proofed. But the relationship to the functionality of the systems is not visible immediately. This could lead to acceptance problems. Therefore a form of description is to find which regards both the flow of signals and the communication parameters. There is no binding predefinition in the relevant standards, yet. A great easement can be expected from IEC 1131 installation tools.

References

/2/ **T. Lipsdorf:** CAN-Applikationssystem für fluidische Antriebe. Otto-von-Guericke-Universität Magdeburg, IPE, Diplomarbeit, 1995