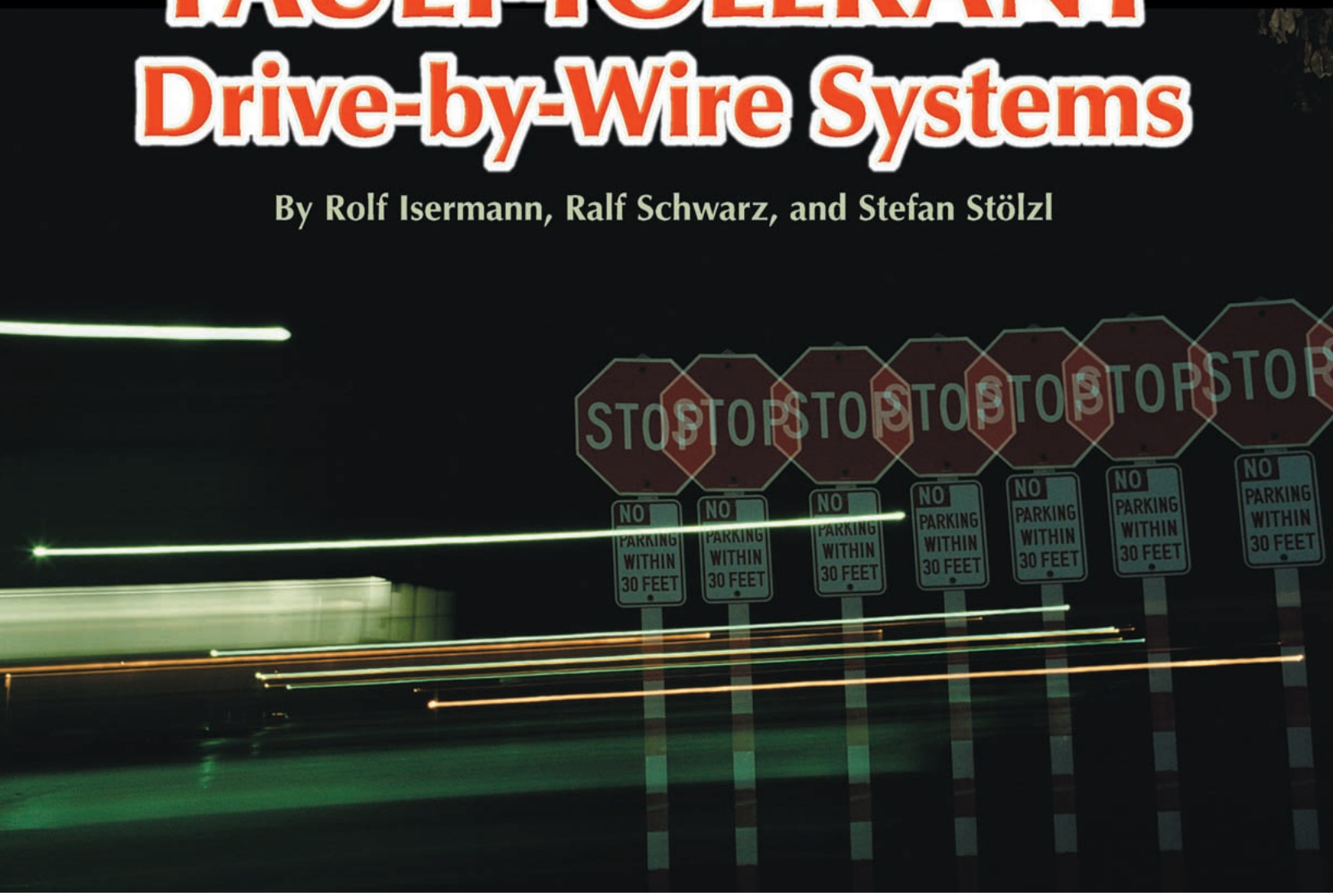
FEATURE

FAULT-TOLERANT Drive-by-Wire Systems

By Rolf Isermann, Ralf Schwarz, and Stefan Stölzl



Automotive systems are increasingly being developed with integrated electronic sensors, actuators, microcomputers, information processing for single components, and engine, drivetrain, suspension, and brake systems. These have matured to mechatronic systems that are characterized by the integration of components (hardware) and signal-based functions (software), resulting in autonomous functionality. Examples of first steps toward mechatronic systems for automobiles were the arrival of digitally controlled combustion engines with fuel injection in 1979 and digitally controlled antilock brake systems (ABS) in 1978. Today's combustion engines are completely microcomputer controlled, typically with five electrical, electrohydraulic, or electropneumatic actuators and several measured output variables, taking into account different operating phases such as startup, warming up, idling, and normal operation. The only input from the driver is generated by the accelerator pedal and transferred to the throttle (spark-ignition engines) or the injection pump (diesel engines)

This article begins with a review of electronic driver assisting systems such as ABS, traction control (TCS), elec-

Isermann (risermann@iat.tu-darmstadt.de) is with the Institut fuer Automatisierungstechnik Technische, Universitaet Darmstadt Landgraf-Georg-Str. 4, D-64283 Darmstadt, Germany. Schwarz is wuth Continental Teves, Frankfurt 60488, Germany. Stölzl is with A.T. Kearney Gmbh, Frankfurt 60327,Germany.

0272-1708/02/$17.00©2002IEEE

tronic stability control (ESP), and brake assistant (BA). We then review drive-by-wire systems with and without mechanical backup. Drive-by-wire systems consist of an operating unit (steering wheel, brake pedal) with an electrical output, haptic feedback to the driver, bus systems, microcomputers, power electronics, and electrical actuators. For their design safety, integrity methods such as reliability, fault tree and hazard analysis, and risk classification are required. Different fault-tolerance principles with various forms of redundancy are considered, resulting in fail-operational, fail-silent, and fail-safe systems. Fault-detection methods are discussed for use in low-cost components, followed by a review of principles for fault-tolerant design of sensors, actuators, and communication. We evaluate these methods and principles and show how they can be applied to low-cost automotive components and drive-by-wire systems. A recently developed brake-by-wire system with electronic pedal and electric brakes is then considered in more detail, showing the design of the components and the overall architecture. Finally, we present conclusions and an outlook for further development of drive-by-wire systems.

Recent Devlopments

Electronic Driver-Assisting Systems

The development of mechatronic systems for the engine and the drive train was paralleled by driver-assisting *electronic braking functions* such as ABS, TCS (1986), ESP [1], and BA; see also [2]. In these cases the hydraulic pressure generated by the driver's brake pedal and pneumatic booster is modulated to control the slip of single wheels (ABS) or to control the drift angle of the vehicle by individual wheel braking (ESP). If the electronic control fails, the brake systems behave like conventional purely hydraulic ones. Since 1945, power steering was realized by hydraulic supporting energy, but electrical power steering for lightweight vehicles that came on the market in 1996 no longer required special hydraulic circuits and auxiliary pumps [3]. In all cases, the direct mechanical linkage with the steering wheel and the driver action was retained.

Drive-by-Wire Systems with Mechanical Backup

Since 1986, the engine has been increasingly manipulated by an electronic pedal and electrically driven throttle or injection [4], [5], representing the first drive-by-wire components. In this case, a "limp home" function after electronic failure is possible because the throttle spring system provides a reduced engine speed (e.g., 1,200 rpm). Hence, the system is rendered fail-safe by the mechanics. Other mechatronic units within the power train were developed for the automatic transmission with hydraulic torque converter and microcomputer-controlled gear shift.

Drive-by-Wire Systems Without Mechanical Backup

The approaches taken for successful fly-by-wire systems and the positive experience with the automotive drive-by-wire systems with mechanical backup for the engine and drive train and the electronic driver assistance systems for braking and power steering are the basis for the development of complete drive-by-wire systems without mechanical backup for braking, steering, and higher-level driver-assisting functions. The disadvantages of mechanical backup systems are that they are costly, heavy, passive safety-critical (steering column, brake pedal), and do not provide enough freedom to tap the potential of the electrical systems. Drive-by-wire systems without mechanical backup generate electrical commands through the driver and transfer them to computer-controlled electromechanical actuators, a scheme that is usually not fail-safe but has fault-tolerant properties.

Higher-Level Automotive Control

A recent development in higher-level control is adaptive cruise control (ACC), also called intelligent cruise control (ICC), where the vehicle either follows a preceding vehicle with distance measurement and control or follows a driver-set reference value as in classical speed control (see [6] and [7]). In this case electrical commands have to be given to an electrical throttle or to the brake booster. Several research automobiles have sensors to assess the road situation (e.g., cameras) and computerized autopilots [8], or for automatic platooning systems, like those pioneered in the California PATH program [9], [10].

This short summary of recent developments shows that from the viewpoints of increased active safety and automated driving, there is a clear demand for drive-by-wire systems, which include the powertrain, braking, and steering. These systems are also called *x*-by-wire systems, where *x* stands for the commanded action. Fig. 1 shows the hazard severity of failures for different electronic (and electrical) driving systems [11]. Clearly, the hazard severity of drive-by-wire control systems for vehicles increases considerably.

Drive-by-Wire Structures with and Without Mechanical Backup

The progression from drive-by-wire systems with mechanical backup to those without mechanical backup or fail-safe by mechanics is a large one because of the lower reliability and different fault behavior of electronic and electrical components compared to mechanical components. Therefore, fault-tolerant electronic systems have to be incorporated to meet the high safety requirements.

Fig. 2 shows the general signal flow diagram of a drive-by-wire system in more detail. The driver's operating unit (steering wheel, braking pedal) has a mechanical input (e.g., torque or force) and an electrical output (e.g., bus protocol). The system contains sensors and switches for position and/or force, microelectronics, and either a passive

(spring-damper) or active (electrical actuator) feedback to give the driver haptic information ("pedal feel") on the action. A bus connects to the brakes or steering control system. This control system consists of power electronics, electrical actuators, brake or steer mechanics, with sensors or reconstructed variables and a microcomputer for actuator control, brake or steer function control, supervision, and various types of management (e.g., fault tolerance with reconfiguration, optimization).

Each of the sensors, electronics, buses, power electronics, high-power actuators, and microcomputers must be fault tolerant with regard to at least one safety-critical failure. Therefore, a safety integrity analysis and methods of fault tolerance are basic issues for drive-by-wire systems.

Safety Integrity Analysis Methods

Drive-by-wire systems are safety-related systems. Therefore, all aspects of reliability, availability, maintainability, and safety (RAMS) have to be considered because they are relevant for manufacturer liability and customer acceptability. To meet safety requirements, special procedures were developed in technical disciplines such as railway, aircraft, space, military, and nuclear systems. These procedures are covered by the terms *system integrity* or *system dependability*.

The various kinds of safety requirements lead to different levels of integrity of safety-related systems, from the lowest to highest requirements. In this context, "integrity" is more precisely termed "safety integrity" with the following definition: "Safety integrity is the probability of a safety-related system satisfactorily performing the required safety functions under all the stated conditions within a stated period of time" [12].

Safety and reliability are generally achieved by a combination of

- fault avoidance
- fault removal
- fault tolerance
- fault detection and diagnosis
- automatic supervision and protection.

Fault avoidance and removal has to be accomplished mainly during the design and testing phase. For investigating the effect of faults on the reliability and safety during the design phase and also for type certification, a range of analysis methods have been developed. They include

- reliability analysis
- event tree analysis (ETA) and fault tree analysis (FTA)
- failure mode and effects analysis (FMEA)
- hazard analysis (HA)
- risk classification.

For details see [12]-[14].

These known methods can now be combined appropriately. Fig. 3 shows an overall scheme. The FMEA identifies all components, failures, causes, and effects. The single failures proceed to an FTA to determine the causes and their logical interconnections on a component level. The failure causes are then used to design the overall reliability. Remaining failures that cannot be avoided are then classified and the maintenance procedure determined.

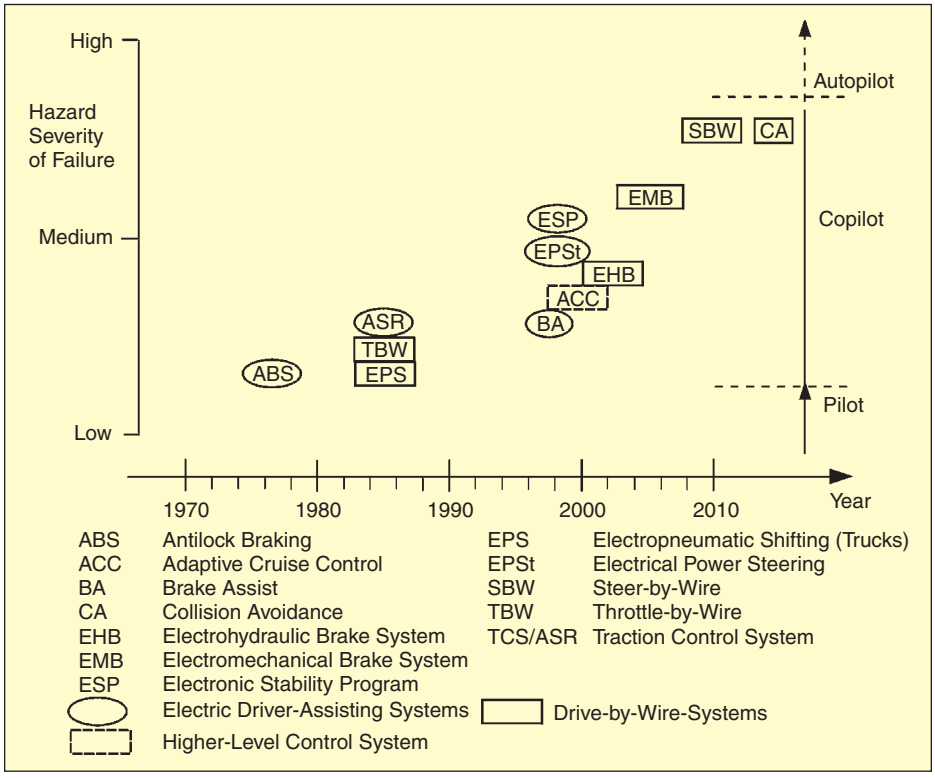


Figure 1. *Hazard severity of failures (qualitative) in electronic driver-assisting systems, drive-by-wire systems, and higher-level control systems.*

Based on the FMEA, the hazard analysis extracts safety-critical failures. Their presentation in a (reduced) fault tree determines the causes with logical interconnections [15] (i.e., dangerous faults leading to hazards). Based on this, the safety system at lower levels can be designed. The remaining dangerous failures then undergo a risk classification and supervision and safety methods to reduce the risk to an acceptable measure are determined.

Herewith a hazard risk number

$$R = C \times F_H \times F_{OP}$$

can be used, where C is the consequence (severity) of hazard, F_H is the frequency (probability) of hazard, and F_{OP} is the frequency of operation state (compare [12], [16]).

In general, fault-tolerance methods have to be implemented

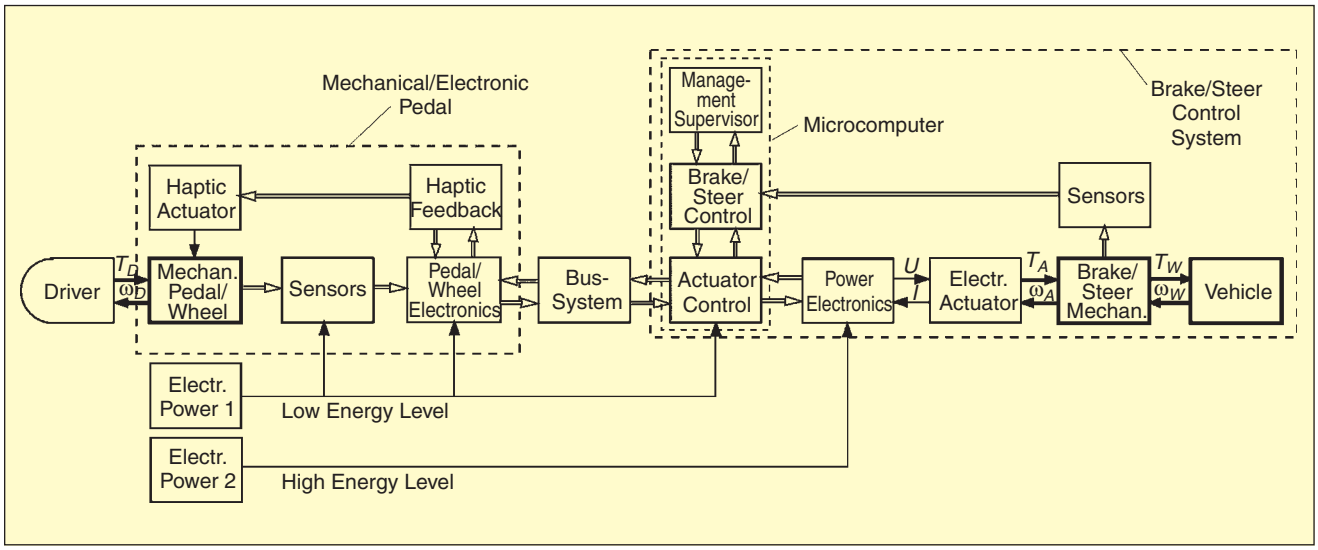


Figure 2. *Basic signal flow diagram of drive-by-wire systems.*

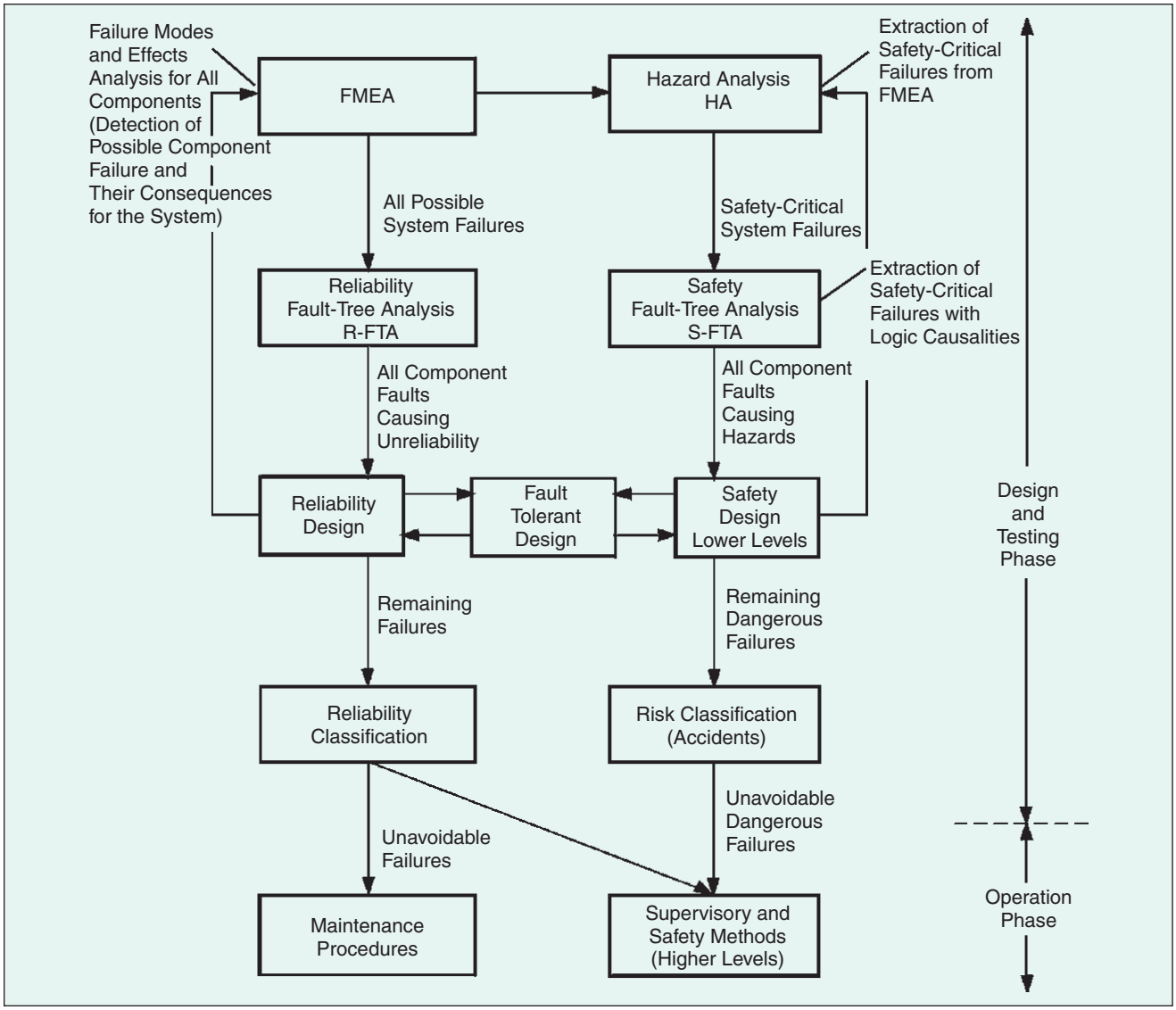


Figure 3. *Integrated design procedures for system reliability and safety to result in high system integrity.*

at the component and unit levels to improve both reliability and safety, especially by reducing F_H. Fig. 3 summarizes the integrated reliability and safety procedure during the design and testing phases.

The unavoidable failures have to be covered by maintenance and online supervision and safety methods during operation, including fault tolerance, protection and supervision with fault detection and diagnosis, and appropriate safety actions. These methods are discussed in the following sections.

Fault-Tolerant Design

After applying reliability and safety analysis methods during design and testing, as well as corresponding quality control methods during manufacturing, certain faults and failures still cannot be avoided totally. Therefore, they should be tolerated by additional design efforts. Hence, high-integrity systems must have, to the extent possible, the ability for fault tolerance. This means that faults are compensated in such a way that they do not lead to system failures. The most obvious way to reach this goal is redundancy in components, units, or subsystems. However, the overall systems then become more complex and costly. In this section various types of fault-tolerant methods are reviewed briefly.

Fault Tolerance for Components

Fault-tolerance methods generally use redundancy. This means that in addition to the considered module, one or more modules are connected, usually in parallel. These redundant modules are either identical or diverse. Such redundant schemes can be designed for hardware, software, information processing, and mechanical and electrical components (sensors, actuators, microcomputers, buses, power supplies, etc.).

Basic Redundant Structures

There are two basic approaches for fault tolerance: static redundancy and dynamic redundancy. The corresponding configurations are first considered for electronic hardware and then for other components.

Fig. 4(a) shows a scheme for static redundancy. It uses three or more parallel modules that have the same input signal and are all active. Their outputs are connected to a voter that compares these signals and decides by majority which signal value is the correct one. If a triple-redundant modular system is applied, and a fault in one of the modules generates a wrong output, this faulty module is masked (i.e., not taken into account) by the two-out-of-three voting. Hence, a single faulty module is tolerated without any effort for specific fault detection. n redundant modules can tolerate $(n-1)/2$ faults (n odd).

Dynamic redundancy requires fewer modules at the cost of more information processing. A minimal configuration consists of two modules, Fig. 4(b) and (c). One module is usually in operation, and if it fails, the standby or backup unit takes over. This requires fault detection to observe if the operational modules become faulty. Simple fault-detection methods use the output signal only for consistency checking (range of the signal), comparison with redundant modules, or use of information redundancy in computers such as parity checking or watchdog timers. After fault detection, it is the task of the reconfiguration to switch to the standby module and to remove the faulty one.

In Fig. 4(b) the standby module is continuously operating, a method called "hot standby." This results in a short transfer time, but at the cost of operational aging (wear-out) of the standby module.

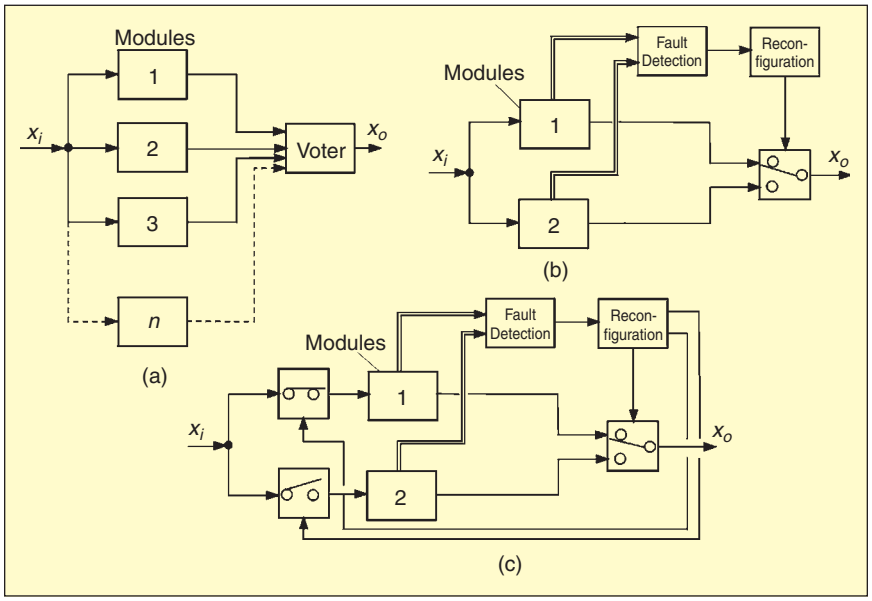


Figure 4. *Fault-tolerant schemes for electronic hardware. (a) Static redundancy: multiple redundant modules with majority voting and fault masking, m out of n systems (all modules are active); (b) dynamic redundancy: standby module that is continuously active (hot standby); and (c) dynamic redundancy: standby module that is inactive (cold standby).*

Dynamic redundancy, where the standby system is out of function and does not wear, is shown in Fig. 4(c) and is called "cold standby." This arrangement requires two additional switches at the input and more transfer time due to a startup procedure. For both schemes, the performance of the fault detection is essential.

Redundant schemes similar to those for electronic hardware exist for software fault tolerance. Here we mean tolerance against mistakes in coding or errors in calculation. The simplest form of static redundancy is repeated running ($n \geq 3$) of the same software and majority voting for the result. However, this is only suitable for some transient faults. As software faults generally are systematic and not random, duplica-

tion of the same software is not sufficient. Therefore, the redundancy must include diversity of software, such as other programming teams, other languages, or other compilers. With $n \geq 3$ diverse programs, a multiple redundant system can be established, followed by majority voting, as in Fig. 4(a). However, if only one processor is used, calculation time is increased, and using n processors may be too costly.

Dynamic redundancy using standby software with diverse programs can be realized by using recovering blocks. This means that, in addition to the main software module, other diverse software modules exist [13], [17].

Fault tolerance can also be designed for purely mechanical and electrical systems. Static redundancy is often used in all types of homogeneous and inhomogeneous materials (e.g., metals and fibers) and in special mechanical constructions such as lattice structures, spoke-wheels, and dual tires, or in electrical components with multiple wiring, multiple coil windings, multiple brushes for dc motors, and multiple contacts for potentiometers. This natural built-in fault tolerance is generally characterized by a parallel configuration. However, the inputs and outputs are not signals but forces, electrical currents, or energy flows, and a voter does not exist. All elements operate in parallel, and if one element fails (e.g., by breakage), the others take over a higher force or current, following the physical laws of compatibility or continuity. Hence, this is a kind of "stressful degradation." Mechanical and electrical systems with dynamic redundancy, as depicted in Fig. 4(b) and (c), can also be built. For the most part, only cold standby is meaningful.

Fault tolerance with dynamic redundancy and cold standby is especially attractive for mechatronic systems where more measured signals and embedded computers are already available, and therefore fault detection can be improved considerably by applying process-model-based approaches. Table 1 summarizes the appropriate fault-tolerance methods for the case of electronic hardware.

Redundant Structures for Drive-by-Wire Components

Mainly because of cost, space, and weight, a suitable compromise between the degree of fault tolerance and the number of redundant components has to be found for automotive drive-by-wire systems. In contrast to fly-by-wire systems, only a single failure must be tolerated (presently) for hazardous cases [18], mainly because a safe state can be reached easier and faster. This means that not all components require stringent fault-tolerance requirements. The following degradation steps are distinguished.

- *Fail-operational* (FO): One failure is tolerated (i.e., the component stays operational after one failure). This is required if no safe state exists immediately after the component fails.
- *Fail-safe* (FS): After one (or several) failure(s), the component directly reaches a safe state (passive fail-safe, without external power) or is brought to a safe state by a special action (active fail-safe, with external power).
- *Fail-silent* (FSIL): After one (or several) failure(s), the component exhibits quiet behavior externally (i.e., stays passive by switching off) and therefore does not wrongly influence other components.

For vehicles, it is proposed to subdivide FO into "long time" and "short time." Considering these degradation steps for various components, one must first check if a safe state exists. For automobiles (usually), a safe state is standstill (or low speed) at a nonhazardous place. For automobile components, a fail-safe status is (usually) a mechanical backup (i.e., a mechanical or hydraulic linkage) for direct manipulation by the driver. Passive fail-safe is then reached after the failure of electronics if independent of the electronics the vehicle comes to a stop (e.g., by a closing spring in the throttle or by actions of the driver via mechanical backup). However, if no mechanical backup exists after failure of the electronics, only an action by other electronics (switch to a still-operating module) can bring the vehicle (in motion) to a safe state (i.e., to reach a stop through active fail-safe). This requires the availability of electric power.

Generally, graceful degradation is envisaged where less critical functions are dropped to maintain the more critical functions, based on priorities [12]. Table 1 shows degradation steps to fail-operational for different redundant structures of electronic hardware. As the fail-safe status depends

Table 1. Fail behavior of electronic hardware for different redundant structures.

		Static Redundancy		Dynamic Redundancy		
Structures	Number of Elements	Tolerated Faults	Fail Behavior	Tolerated Failures	Fault Behavior	Discrepancy Detection
Duplex	2	0	F	0	F	two comparators
				1	FO-F	fault detection
Triplex	3	1	FO-F	2	FO-FO-F	fault detection
Quadruplex	4	1	FO-F	3	FO-FO-FO-F	fault detection
Duo-Duplex	4	1	FO-F	—	—	—

FO: fail-operational; F: fail.

on the controlled system and the kind of components, it is not considered here.

For flight-control computers, a triplex structure with dynamic redundancy (hot standby) is typically used, which leads to FO-FO-FS, such that two failures are tolerated and a third one allows the pilot to operate manually. If the fault tolerance has to cover only one fault to stay fail-operational (FO-F), a triplex system with static redundancy or a duplex system with dynamic redundancy is appropriate. If fail-safe can be reached after one failure (FS), a duplex system with two comparators is sufficient. However, if one fault has to be tolerated to continue fail-operational and after a subsequent fault it is possible to switch to fail-safe (FO-FS), either a triplex system with static redundancy or a duo-duplex system may be used. The duo-duplex system has the advantages of simpler failure detection and modularity.

Fault Tolerance for Control Systems

For automatically controlled systems, the appearance of faults and failures in the actuators, the process, and the sensors will usually affect the operating behavior. With feedforward control, generally all small or large faults influence the output variables and thus the operation.

If the system operates with feedback control, small additive or multiplicative faults in the actuator or process are covered by the controller because of usual robustness properties. This property is therefore a passive controller fault tolerance. However, additive and gain sensor faults will immediately lead to deviations from the reference values. For large changes in actuators, process, and sensors, the dynamic control behavior becomes either too sluggish, too underdamped, or even unstable. Then either a very robust control system or an active fault-tolerant control system is required to save the operation. In the latter case, it consists of fault-detection methods and reconfiguration mechanisms that modify the controller. Depending on the type of faults, the reconfiguration may change the structure and/or parameters of the controller. This can also include changes to other manipulated variables or actuators or sensors, if available.

Examples include fault-tolerant flight control with reconfiguration to other control surfaces after failure of actuators or ailerons, elevators, and rudders (see [19]-[22]). For failures in the satellite altitude control system, see [23]. Failures in heat exchangers are treated in [24] and fault-tolerant control for lateral vehicle control in [25].

Fault Detection for Sensors, Actuators, and Mechatronic Servo Systems

Fault-detection methods based on measured signals can be classified as follows:

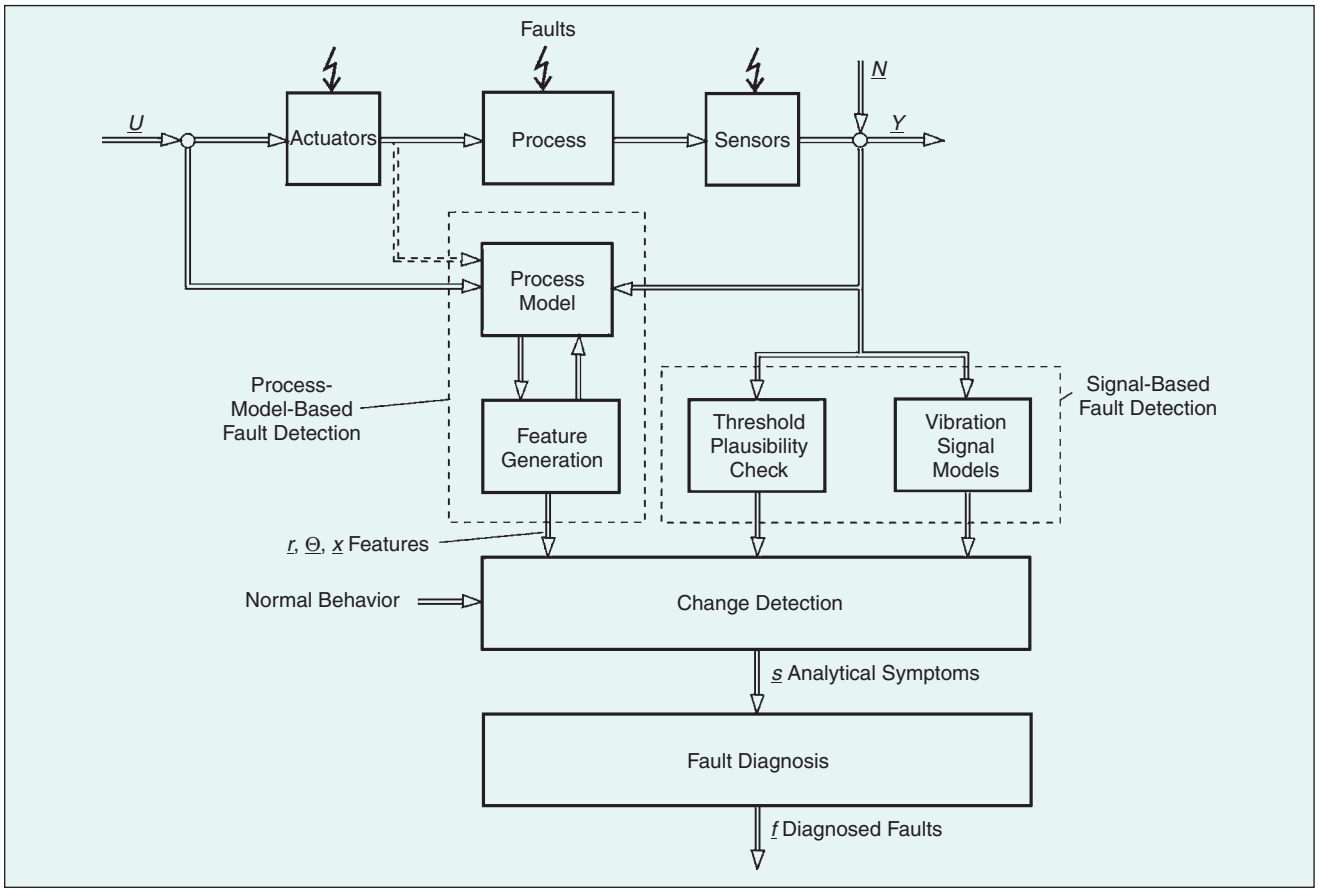


Figure 5. *General scheme of process-model-based and signal-based fault detection.*

- limit value checking (thresholds) and plausibility checks (ranges) of single signals
- signal-model-based methods for single periodic or stochastic signals
- process-model-based methods for two or more related signals.

Fig. 5 shows a scheme for these methods. For a description of the various methods, refer to the literature (e.g., the special section in [26] or [27]-[29]).

To obtain specific symptoms, it is necessary to have more than one input and one output signal for parity equations or output observers. For parameter estimation, one input and one output may be sufficient. Because of the various properties, different methods should be combined to obtain broad fault detection coverage [30], [31]. Some application cases of fault detection and diagnosis are:

1) online testing in manufacturing (quality control)
2) online testing during service
3) online, real-time supervision during operation (on-board).

The first two cases generally require a detailed fault diagnosis with classification or inference methods and can be applied if computationally feasible. However, for the third case, fault-detection capability usually is sufficient for fault-tolerant systems. A fault diagnosis is not necessarily required. However, diagnosis capability is advantageous for general online supervision. Especially for on-board applications in automobiles, the allowable computations are very limited, which restricts the fault detection to methods with fewer computations on microcomputers; see [32]. Furthermore, the fault-detection methods must be transparent and easy to understand, must function reliably for the different operating conditions, must use only few measured signals, and must require little effort for modeling. Maintenance effort and easy transfer to modified components are also important issues.

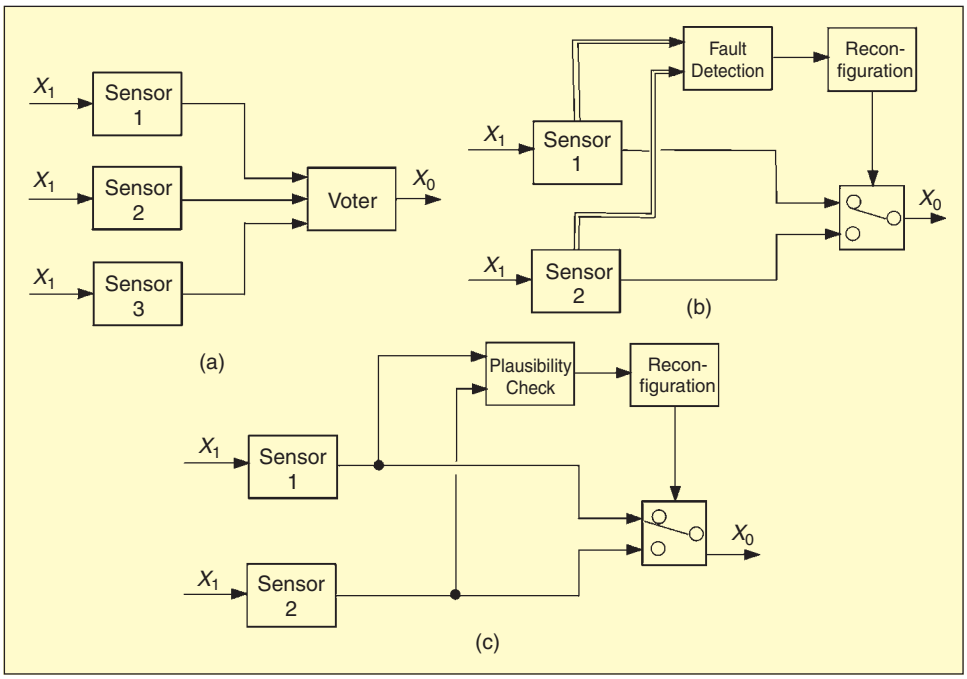


Figure 6. *Fault-tolerant sensors with hardware redundancy. (a) Triplex system with static redundancy and hot standby, (b) duplex system with dynamic redundancy, and (c) duplex system with dynamic redundancy, hot standby, and plausibility checks.*

Fault-Tolerant Components for Drive-by-Wire Systems

The discussion of high-integrity systems and drive-by-wire systems shows that a comprehensive overall fault-tolerance design can be obtained by fault-tolerant components and fault-tolerant control. This means designing fault-tolerant

- sensors
- actuators
- process parts
- computers
- communication (bus systems)
- control algorithms.

Examples of components with multiple redundancy exist for aircraft, space, and nuclear power systems. However,

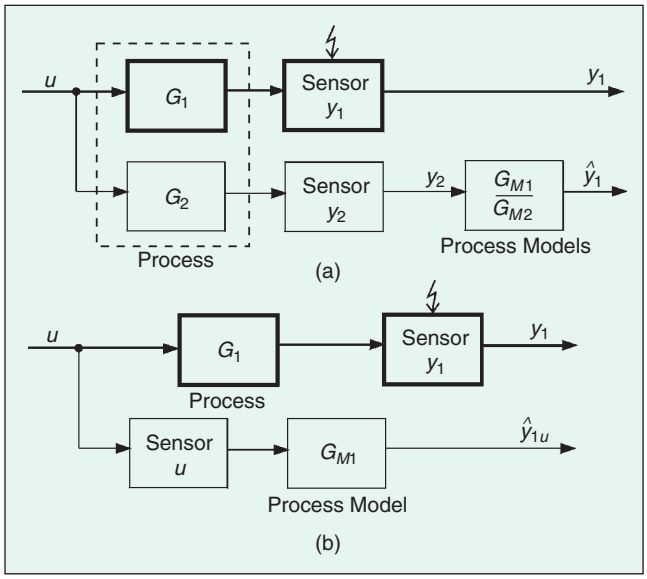


Figure 7. *Sensor fault tolerance for one output signal y_1 (main sensor) through analytical redundancy by process models (basic schemes): (a) two measured outputs, no measured input and (b) one measured input and one measured output.*

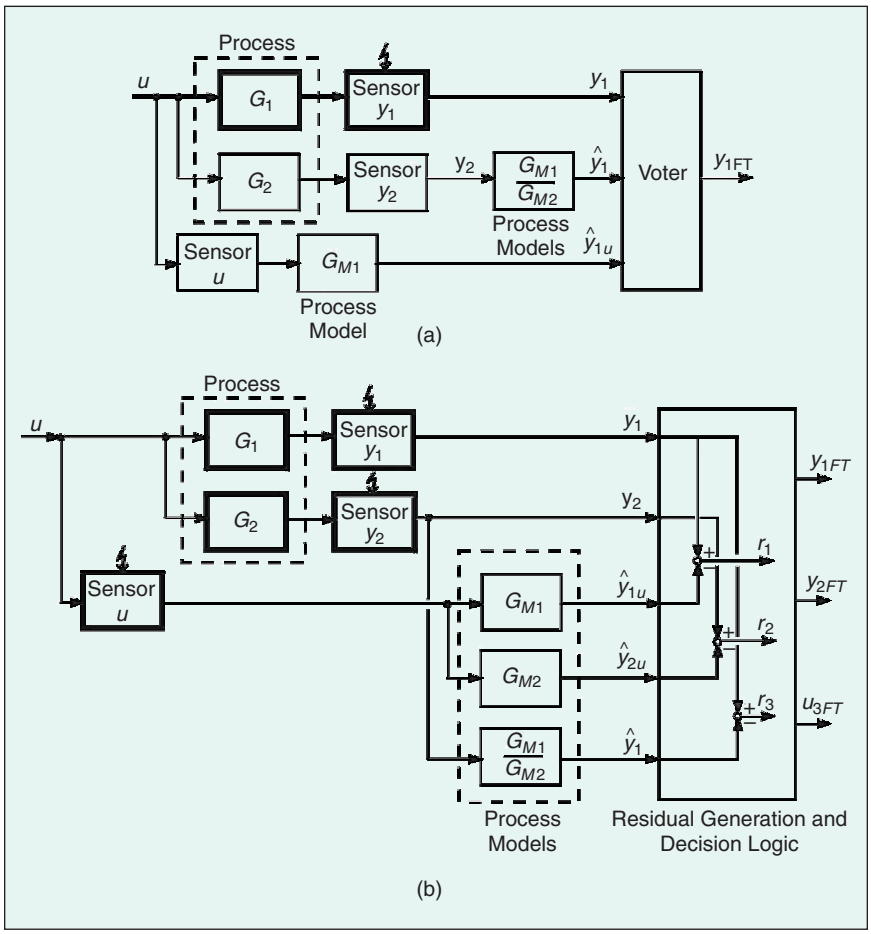


Figure 8. *Fault-tolerant sensors with combined analytical redundancy for two measured outputs and one measured input through (analytical) process models: (a) y_1 is main measurement, y_2, u are auxiliary measurements (combination of Fig. 7(a) and (b)) and (b) y_1, y_2, and u are measurements of same quality (parity equation approach).*

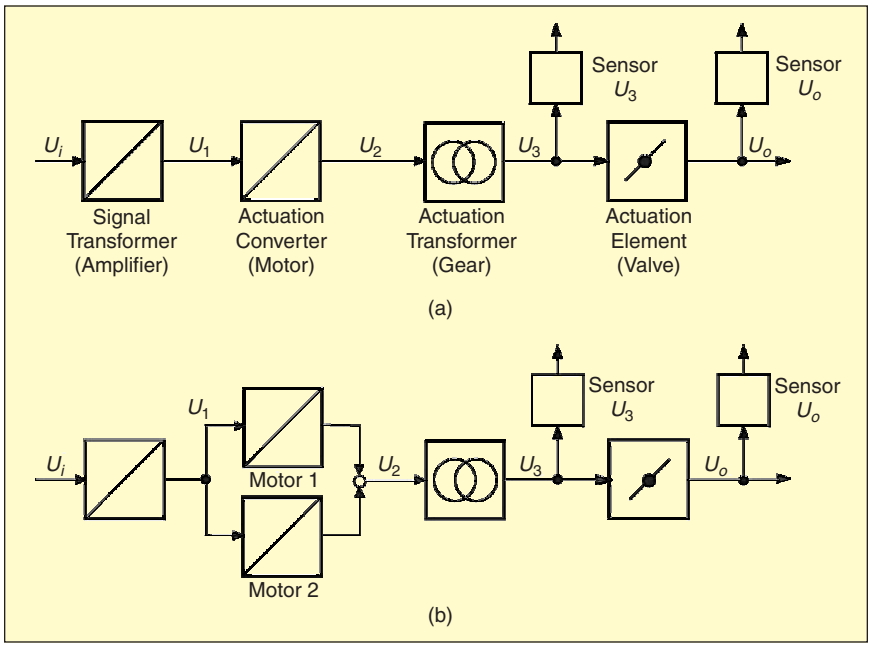


Figure 9. *Fault-tolerant actuator: (a) common actuator and (b) actuator with duplex drive.*

lower cost components with built-in fault tolerance have to be developed. The following sections describe examples from the automotive area, in addition to examples from other fields.

Fault-Tolerant Sensors

A fault-tolerant sensor configuration should be at least fail-operational (FO) for one sensor fault. This can be obtained by hardware redundancy with the same type of sensors or by analytical redundancy with different sensors and process models.

Hardware Sensor Redundancy

Sensor systems with static redundancy are realized with a triplex system and a voter (Fig. 6(a)). A configuration with dynamic redundancy needs at least two sensors and a fault detection for each sensor (Fig. 6(b)). Usually only hot standby is feasible. Another less powerful possibility is to use plausibility checks for two sensors, as well as using signal models (e.g., variance) to select the more plausible one.

The fault detection can be performed by self-tests (e.g., by applying a known measurement value to the sensor). Another way is to use self-validating sensors [33], [34], where the sensor, transducer, and a microprocessor form an integrated, decentralized unit with self-diagnostic capability. The self-diagnosis takes place within the sensor or transducer and uses several internal measurements. The output consists of the sensor's best estimate of the measurement and a validity status, such as good, suspect, impaired, bad, or critical.

Analytical Sensor Redundancy

As a simple example, we consider a process with one input and one main output y_1 and an auxiliary output y_2 (see Fig. 7(a)). Assuming the process input signal u is not available but two output signals y_1 and y_2, which both depend on u, are available, one of the signals (e.g., $\hat{y}_1$) can be reconstructed and used as a redundant signal if process models G_{M1} and G_{M2} are known and significant disturbances do not appear (ideal cases).

For a process with only one output sensor y_1 and one input sensor u, the output $\hat{y}_1$ can be reconstructed if the process model G_{M1} is known (Fig. 7(b)). In both cases, the relationship between the signals of the process are used and expressed in the form of analytical models.

To obtain one usable fault-tolerant measurement value y_{1FT}, at least three different values for y (e.g., the measured one and two reconstructed ones) must be available. This can be obtained by combining the schemes of Fig. 7(a) and (b) as shown in Fig. 8(a). A sensor fault y_1 is then detected and masked by a majority voter, and either $\hat{y}_1$ or $\hat{y}_{1u}$ is used as a replacement, depending on a further decision. (Also, single sensor faults in y_2 or u are tolerated with this scheme.)

One example of this combined analytical redundancy is the yaw rate sensor for the ESP, where the steering wheel angle is also used as input to reconstruct the yaw rate through a vehicle model, as in Fig. 7(b), and the lateral acceleration and the wheel speed difference between the right and left wheels (no slip) are used to reconstruct the yaw rate according to Fig. 7(a) [35].

A more general fault-tolerant sensor system can be designed if two output sensors and one input sensor yield measurements of same quality. Then, according to the scheme shown in Fig. 8(b), three residuals can be generated, and by decision logic, fault-tolerant outputs can be obtained in the case of single faults of any of the three sensors. The residuals are generated based on parity equations. In this case, state observers can be used for residual generation; compare, for example, the dedicated observers of [36]. (Note that all schemes assume ideal cases. For realizibility constraints of the inverted models, additional filters have to be considered.)

If possible, a faulty sensor should be fail-silent (i.e., should be switched off); however, this requires additional switches, which lowers the reliability. For both hardware and analytical sensor redundancy without fault detection for individual sensors, at least three measurements must be available to make one sensor fail-operational. However, if the sensor (system) has built-in fault detection (integrated self-test or self-validation), two measurements are enough and a scheme like the one in Fig. 6(b) can be applied. (This means that by methods of fault detection, one element can be saved.)

Fault-Tolerant Actuators

Actuators generally consist of different parts: input transformer, actuation converter, actuation transformer, and actuation element (e.g., dc amplifier, dc motor, gear and valve, as shown in Fig. 9(a)). The actuation converter converts one type

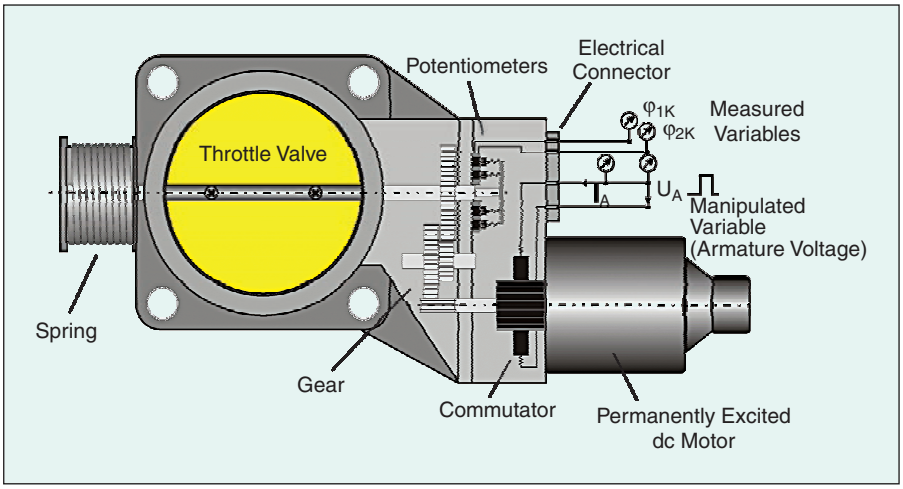


Figure 10. *Scheme of an electromechanical throttle valve actuator.*

Table 2. Process parameter deviations for different actuator faults.

	Features→	Parameter Estimates						
	Faults↓	R_A	Ψ	c_{0e}	J	c_f	M_{R1}	M_{R0}
F1	incr. spring pretension	0	0	0	0	0	0	+
F2	decr. spring pretension	0	0	0	0	0	0	–
F3	commutator shortcut	–	–	0	+	+	+	0
F4	arm. winding shortcut	0	–	0	+	+	+	0
F5	arm. winding break	+	–	0	0	+	+	+
F6	add. serial resistance	+	0	0	0	0	0	0
F7	add. parallel resistance	–	–	0	0	+	+	0
F8	increased gear friction	0	0	0	+	+	+	0
F9	offset fault U_A	0	0	+/–	0	0	0	0
F10	offset fault I_A	0	0	+/–	0	0	0	–/+
F11	offset fault φ_k	0	0	0	0	0	0	–/+
F12	scale fault U_A	+/–	+/–	+/–	+/–	+/–	+/–	+/–
F13	scale fault I_A	–/+	0	0	+/–	+/–	+/–	+/–
F14	scale fault φ_k	0	–/+	0	–/+	–/+	–/+	–/+

0: no significant change; +: large increase; –: large decrease.

of energy (e.g., electrical or pneumatic) into another (e.g., mechanical or hydraulic). Available measurements are frequently the input signal U_i, manipulated variable U_0, and intermediate signal U_3.

Fault-tolerant actuators can be designed by using multiple complete actuators in parallel, with either static redundancy or dynamic redundancy with cold or hot standby (Fig. 4). One example of static redundancy is hydraulic actuators for fly-by-wire aircraft, where at least two independent actuators operate with two independent hydraulic energy circuits.

Another possibility is to limit the redundancy to parts of the actuator that have the lowest reliability. Fig. 9(b) shows a scheme where the actuation converter (motor) is split into separate parallel parts. Examples with static redundancy are two servo valves for hydraulic actuators [37] or three windings of an electrical motor (including power electronics) [38]. Within electromotor-driven throttles for spark-ignition engines, only the slider is doubled to make the potentiometer position sensor static redundant.

An example of dynamic redundancy with cold standby is the cabin pressure flap actuator in aircraft, where two independent dc motors exist and act on one planetary gear [32].

As cost and weight generally are higher for them than for sensors, actuators with fail-operational duplex configuration are preferred. Then either static redundant structures, where both parts operate continuously (Fig. 4(a)), or dynamic redundant structures with hot (Fig. 4(b)) or cold standby (Fig. 4(c)) can be chosen. For dynamic redundancy, fault-detection methods for the actuator parts are required [39]. One goal should always be that the faulty part of the actuator fails silent, i.e., has no influence on the redundant parts. Fault-tolerant communication systems are treated in [40]-[43].

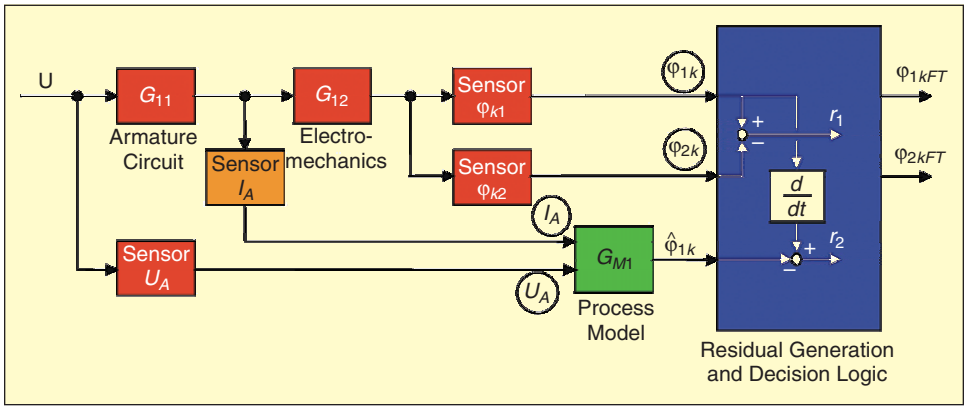


Figure 11. *Fault-tolerant double position sensors for an electromechanical throttle valve.*

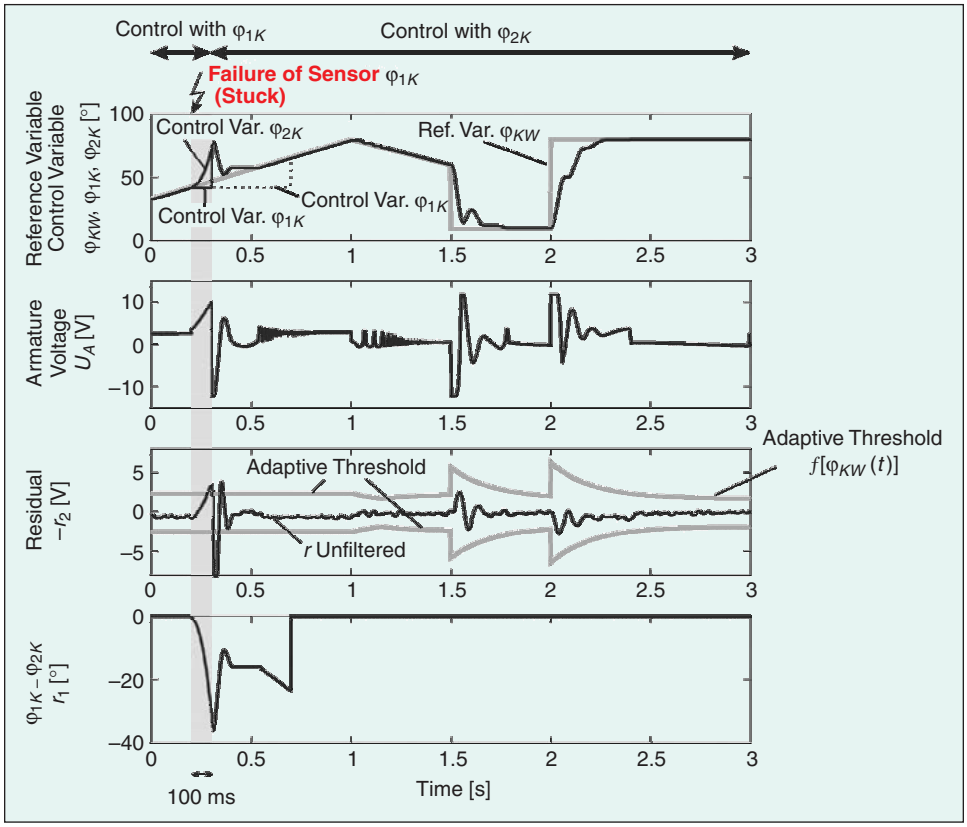


Figure 12. *Closed-loop position control behavior after stuck fault of the first sensor φ_{1k}, fault detection, and reconfiguration with the second sensor φ_{2k}.*

Example of the Diagnosis and Sensor Fault Tolerance of an Electrical Throttle Actuator

The described methodology for fault detection and fault tolerance will now be demonstrated for an electromechanical throttle valve, one of the first drive-by-wire components introduced.

Fault Detection and Diagnosis of the Actuator

Fig. 10 shows the scheme of a throttle-valve actuator. A permanently excited dc motor with gear acts on the throttle against a spring. The motor is driven by a pulsewidth-modulated (PWM) armature voltage U_A and armature current I_A. The angular position φ_k is re-

dundantly measured by two potentiometers, φ_{K1} and φ_{K2}, within 0-90°. Three measured variables $U_A(t)$, $I_A(t)$, and $\varphi_K(t)$ are available. Two differential equations are used.

Armature circuit:

$$U_A(t) = R_A I_A(t) + \Psi\omega_A(t) \tag{1}$$

Electromechanical part:

$$\nu J\dot{\omega}_k(t) = \Psi I_A(t) - \frac{1}{\nu}(c_F\varphi_k(t) + M_0) - M_{R1}\omega_A. \tag{2}$$

The symbols used are

$\omega_k = \dot{\varphi}_k$: throttle angular speed

J: ratio of motor inertia

R_A: armature resistance

c_F: spring constant

Ψ: flux linkage

M_0: Coulomb friction torque

M_{R1}: viscous friction torque.

By applying continuous-time parameter estimation with state-variable filtering to obtain the signal derivatives, the following model parameters were estimated:

$$\underline{p} = [R_A, \Psi, c_{oe}, J, c_F, M_{R1}, M_0], \tag{3}$$

where c_{oe} is a dc value.

In addition to fault detection by parameter estimation, two parity equations are used for fault detection of the sensors φ_{1k} and φ_{2k}:

$$r_1(t) = \varphi_{1k}(t) - \varphi_{2k}(t) \tag{4}$$

$$\begin{aligned} r_{2(t)} &= \Psi\nu[\dot{\varphi}_{1k}(t) - \hat{\dot{\varphi}}_{1k}(t)] \\ &= \Psi\nu\dot{\varphi}_{1k}(t) - U_A(t) + R_A I_A(t). \end{aligned} \tag{5}$$

The first residual r_1 directly indicates discrepancies between the potentiometer outputs, and r_2 uses analytical redundancy through the electromagnetic dc motor model to predict the output $\hat{\dot{\varphi}}_{1k}$ based on measurements $U_A(t)$ and $I_A(t)$.

Table 2 shows the obtained parameter deviations as symptoms through parameter estimation with excitation of the closed-loop position control using a pseudo-random binary signal (PRBS). Nearly all faults generate different symptom patterns, except F11 and F1, F2. The symptom patterns for the process parameter faults F1 through F8 show better isolation than for the offset sensor faults. Therefore, faults F1 through F8 are detected by parameter estimation and sensor faults F9 through F14 by parity equations. Hence, by combining both fault-detection methods, many different faults are detectable; see [31].

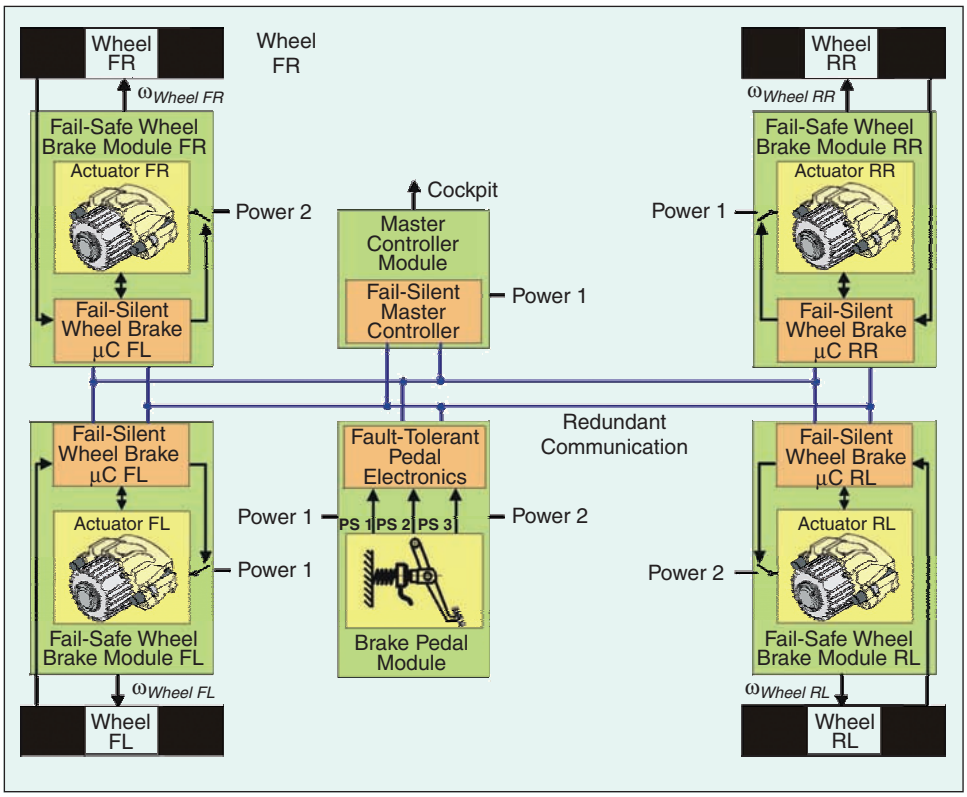


Figure 13. *Brake-by-wire architecture.*

For fault diagnosis, a fault-symptom tree based on fuzzy-logic decision was developed. A special test signal with full-range slow inputs and fast input changes (PRBS) is used to test all types of faults within a time period of about 8 s [44].

Fault-Tolerant Position Sensors

The two parity equations are now used to obtain a fault-tolerant position sensor system for the actuator. Fig. 11 depicts the applied fault detection and sensor fault-tolerance scheme, which corresponds to Fig. 8(b). Based on the residual deviations indicated in Table 3, sensor offset faults of both potentiometers can be isolated. This is used to connect the healthy sensor to the position controller. (Note that here, in contrast to Fig. 6(a), two redundant sensors are used and a third sensor output is calculated from the sensor signals $U_A(t)$ and $I_A(t)$ of the dc motor applying analytical

Table 3. Residual deviations for parity equations.

Residuals→ / Faults↓	r_1	r_2
Offset sensor $+\Delta\varphi_{1k}$	+	+
Offset sensor $+\Delta\varphi_{2k}$	–	0
Faults in armature circuit	0	+/–

redundancy.) Table 3 shows that if only r_2 changes, faults in the armature circuit can be detected.

In Fig. 12, the closed-loop position control behavior with the throttle valve is shown for the case where sensor φ_{1k} sticks at $t = 0.2$ s during a ramp change of the position reference variable. Residuals r_1 and r_2 both deviate, indicating a fault in sensor φ_{1k}. After 100 ms, the residual r_2 exceeds the (adaptive) threshold, the controller is switched to sensor φ_{2k}, and after an additioanl 200 ms, the throttle angle reaches its reference variable again [44]. In practice, this would mean a short additional increase in vehicle speed and an alarm to the driver with further hints (e.g., to visit a service station).

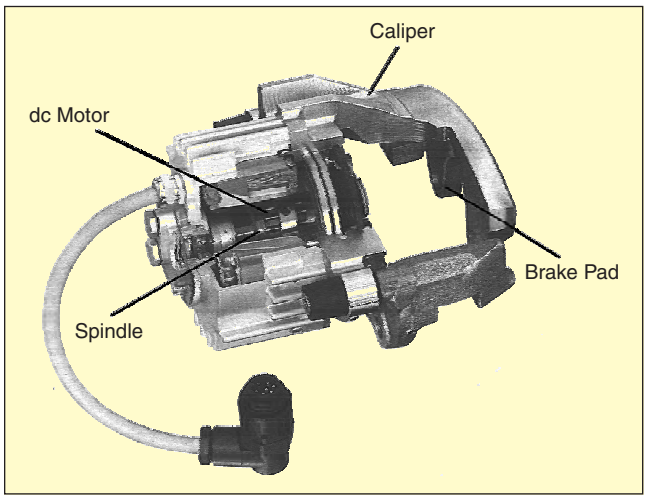


Figure 14. *Prototype of an electromechanically actuated wheel brake by Continental Teves.*

An Example Brake-by-Wire System

A brake-by-wire system without mechanical backup is described as an example of the development of fault tolerance and supervisory functions for a drive-by-wire system. The version illustrated is a prototype electrical brake system, recently developed by Continental Teves, Frankfurt, Germany, partially in cooperation with the authors [45], [46]; see also [47] and [48].

The brake-by-wire system consists of four electromechanical wheel brake modules with local microcomputers, an electromechanical brake pedal module, a duplex communication bus system, and a central brake management computer (Fig. 13).

The chosen overall structure is the result of an FMEA and hazard analysis. Fault tolerance with duplex systems is implemented for the wheel brake controllers, the bus communication, the brake management computer, and the power supply; however, the brake pedal contains internally higher redundancy because of its central function.

Electromechanical Brake

The design of the electromechanical wheel brake is driven by the demand for high electromechanical efficiency, minimized space, lightweight construction, and robustness against rough environmental conditions. Fig. 14 shows the basic construction [45].

A generalized four-pole model of the electromechanical brake (EMB) is depicted in Fig. 15. The current is controlled by power electronics. The dc motor's torque is converted by the spindle gear into a friction force at the disk, which results in a braking force. To compensate for the large parame-

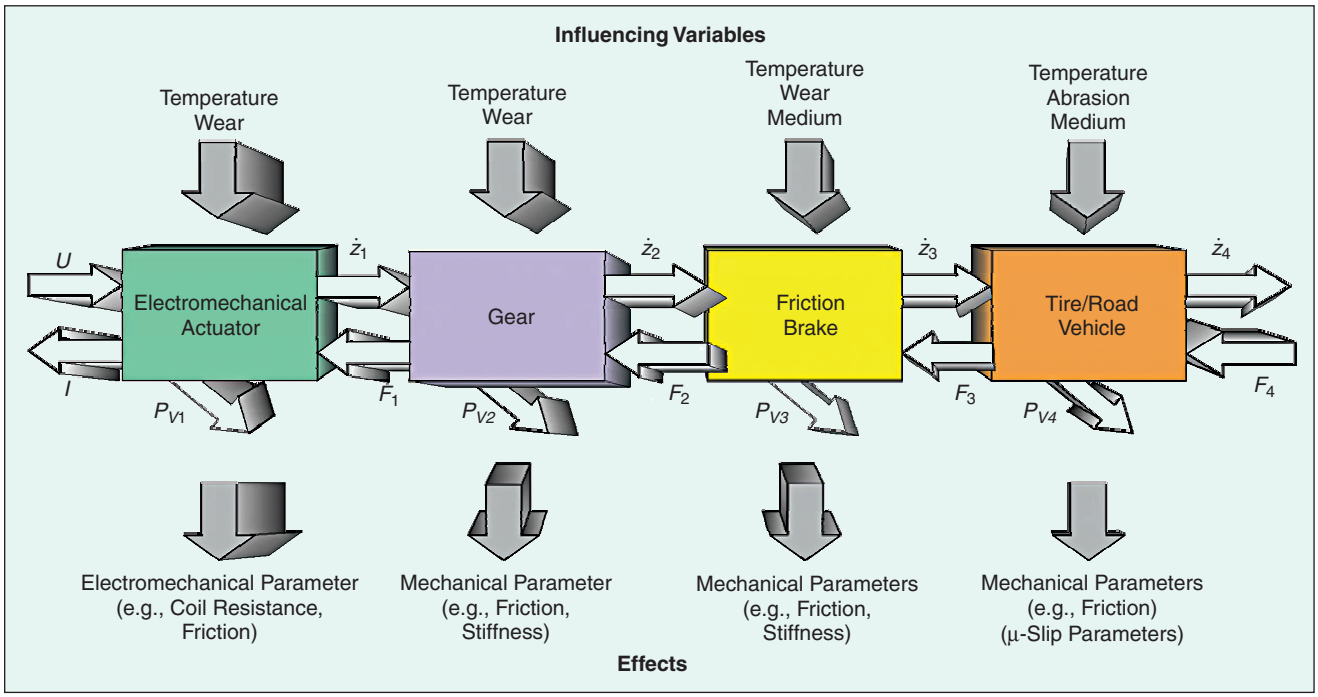


Figure 15. *Generalized four-pole models of an electromechanical disk brake. U, I: voltage, current; P_{vi}: power losses; $\dot{z}, F$: speed, force or torque.*

ter variations (and changing efficiency) in the whole EMB, a closed-loop control of the clamping force or brake torque is used in a cascaded loop system, with current and speed controllers as slave controllers. However, this requires special sensors. As an alternative, the clamping force or the braking torque can be reconstructed by measuring only voltage, current, and position of the dc motor using adaptive dynamic models of the EMB [46].

The corresponding control and other algorithms, such as parameter estimation, clamping force reconstruction, and clearance-contact point detection, are implemented in the brake module microcomputer system. It is designed as a duplex system and connected to the brake control management computer and the brake pedal via a duplex control area network (CAN)-bus system.

Electromechanical Brake Pedal

The driver's input to the pedal is measured by a proper combination of position (and force) sensors. The measured analog signals are then transmitted to microcontrollers. After signal processing, the brake pedal information is given to the CAN-bus system. Because the pedal represents a central part of the brake system, its design must be highly fault tolerant.

An integrated FMEA and hazard analysis as summarized in Fig. 3 was therefore applied to the pedal unit. Major hazards of the EMB determined in [15] and [47] are

1) no braking after brake command from the driver
2) braking without brake command from the driver
3) braking with wrong deceleration
4) one-sided braking
5) braking with unacceptable time delay.

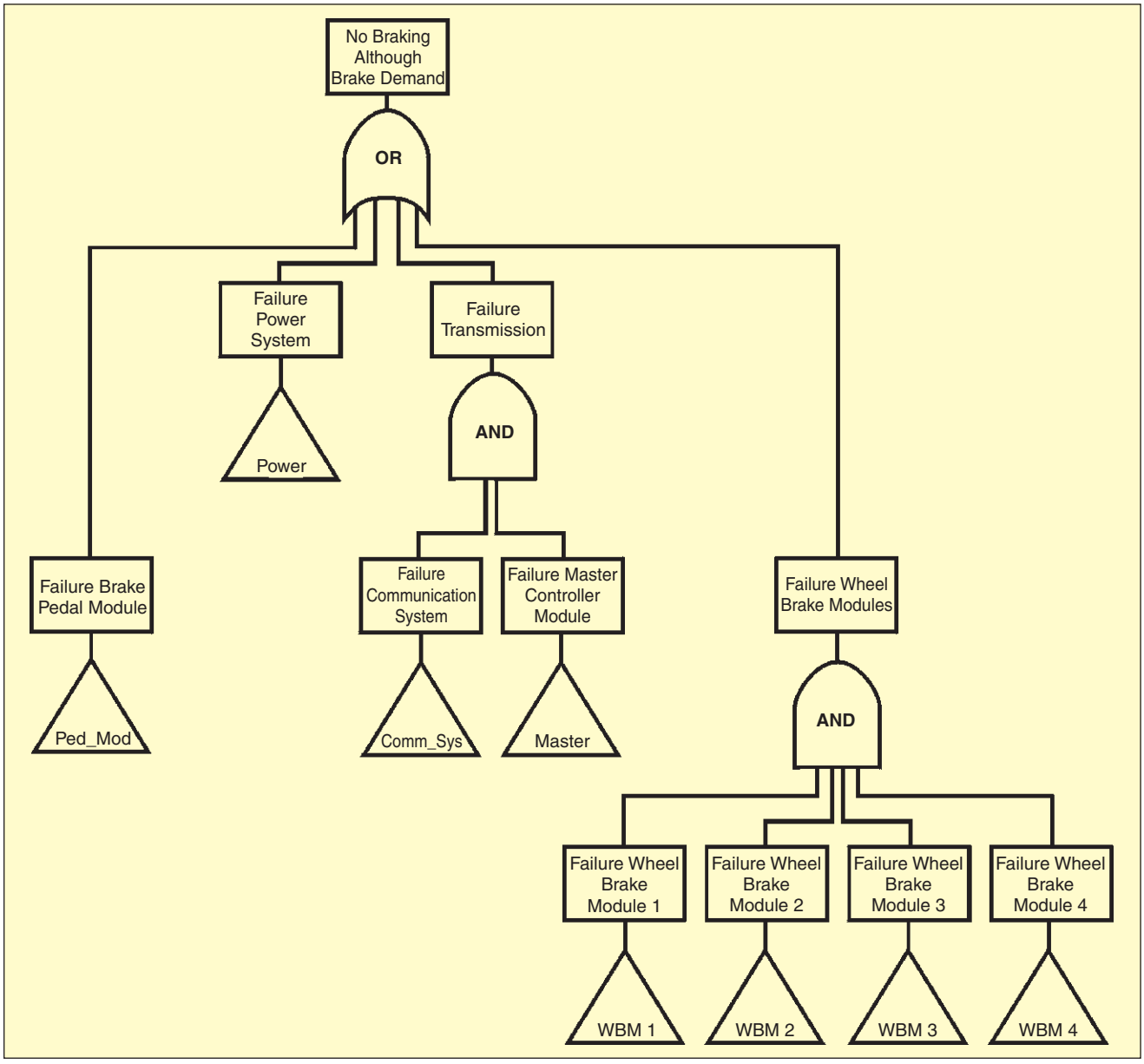


Figure 16. *A fault tree for hazard 1 (no braking after brake command).*

Fig. 16 shows the hazard analysis of the electromechanical brake for hazard 1 (no braking after brake command) by using a safety fault tree. The fault tree transfer symbols (Δ) represent further fault trees (subtrees). The top events of the subtrees in Fig. 16 are the causes of hazard 1: failed brake pedal module, failed power system, failed transmission, or failed wheel brake modules. Therefore, all of the four wheel brake modules have to fail before the hazard "No braking although brake command" occurs.

The subtrees are constructed until basic or primary events are reached. The circle symbols in Fig. 17 represent the basic events. In this example, each of the events (Failure Sensor Electronics 1, Failure Sensor Cable 1, Failure Power Supply 1, Failure Sensor Element 1, or Failure Sensor Connection 1) results directly in the event "Failure Clamping-Force Sensor 1" because of the OR-connection, and together they represent the causes of that event. However, all of the three wheel brake sensors have to fail before event "Failure Wheel Brake Module 1" occurs because the brake control operates with only one of these sensors.

From this FMEA and hazard analysis the following recommendations were derived:

- overdimensioned pedal mechanics
- fault-tolerant pedal electronics
- fault-tolerant touchless pedal sensors
- two independent power supplies
- two independent plug connectors for communication
- two separate boxes with sufficient protection (EMC) and cooling
- avoidance of common-mode failures.

The most sensitive parts, such as electronics and sensors, were found to have triple or quadruple modular redundancy.

Fig. 18 shows the design of the resulting overall structure. It represents a fault-tolerant and distributed real-time system that consists of three different kinds of modules: one brake pedal module, one central controller module, and four wheel brake modules. The real-time communication system and the power system have dynamic redundancy with hot standby. The pedal module must be fail-operational after one failure in the sensors, electronics, or plug connections. The higher-level brake functions such as ABS, TCS, ESP, and the master supervision functionality of the brake-by-wire system are mainly implemented in the integrated software of the fail-silent central controller module. The wheel brake modules can

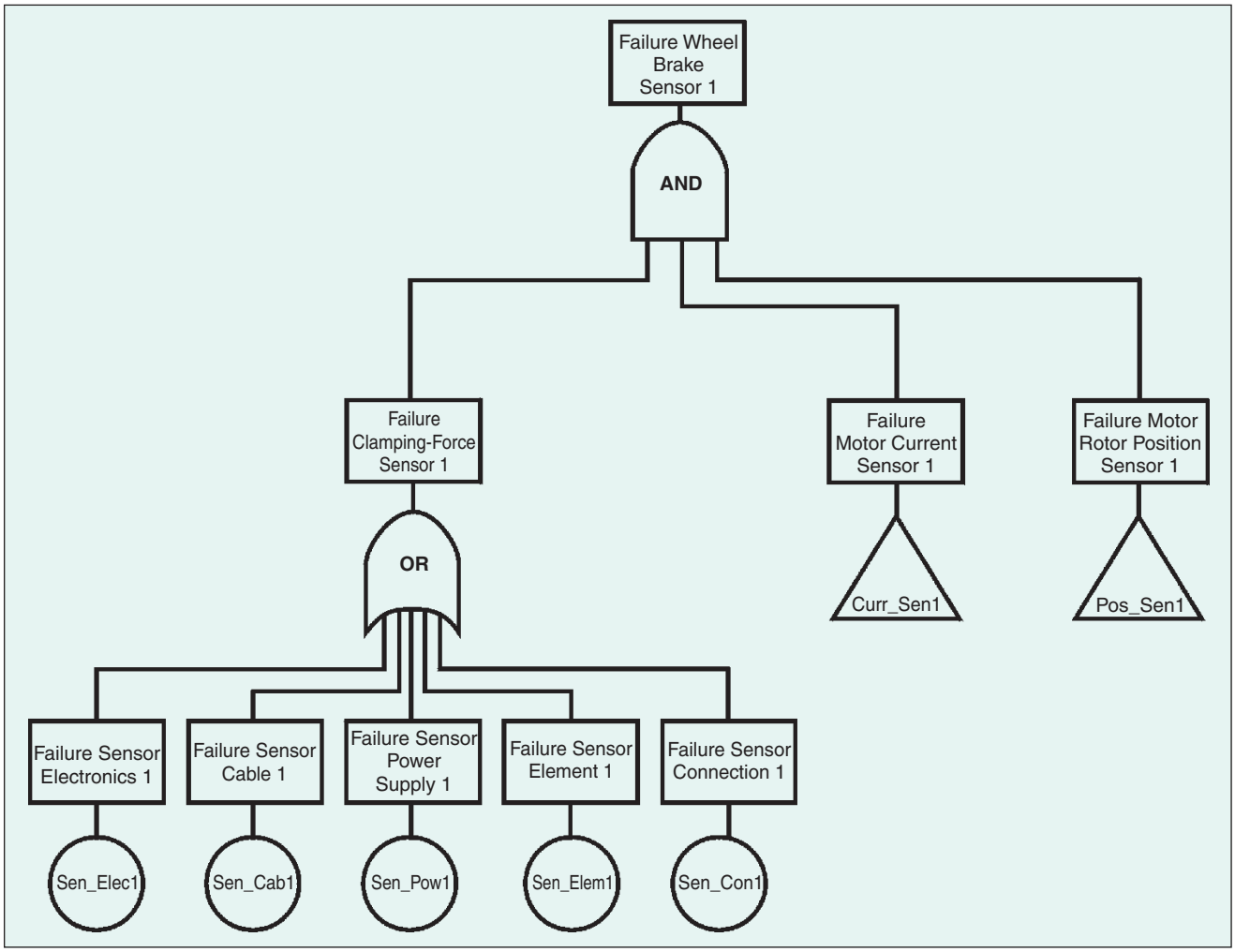


Figure 17. *Sub-fault-tree of hazard 1 (Fig. 16) with sub-top-event "failure pedal sensor."*

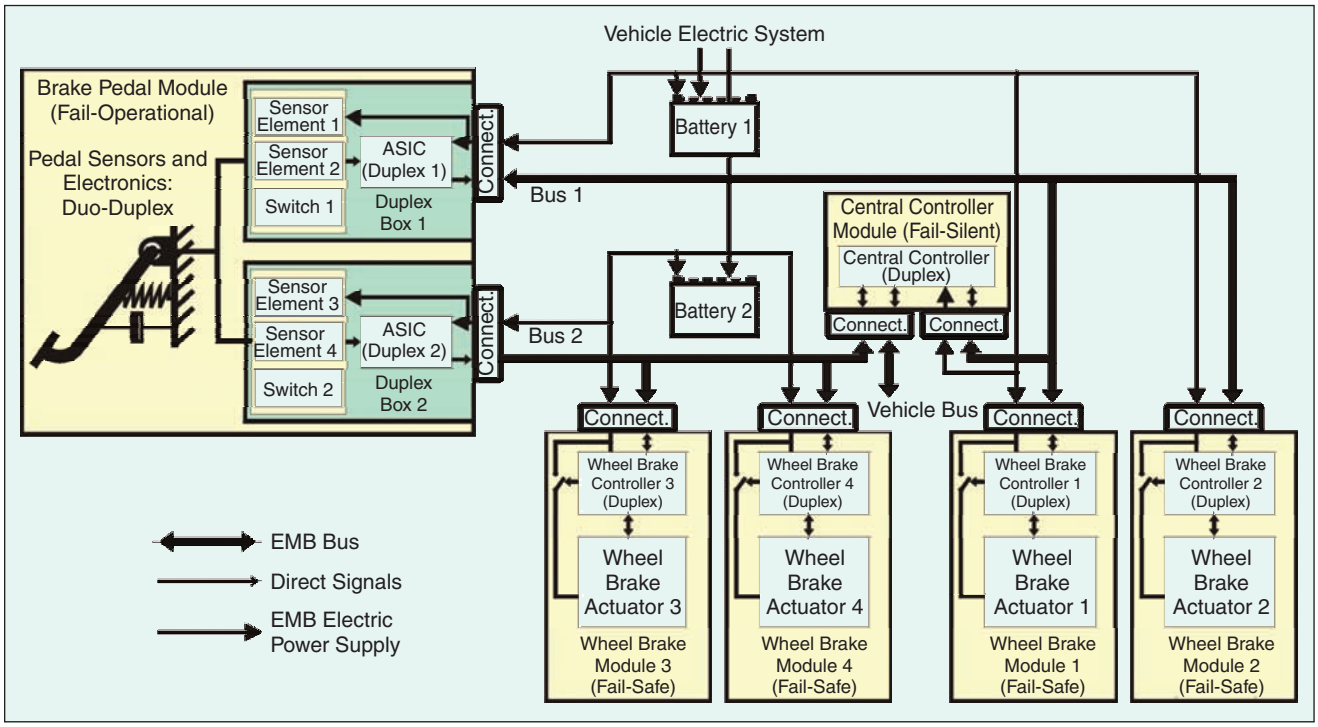


Figure 18. *Scheme of the fault-tolerant electromechanical brake system.*

detect whether the brake management controller is working correctly or has transferred to a silent state after a failure in this unit has occurred.

Fig. 19 shows the electronic hardware architecture of the pedal module. To obtain the fail-operational behavior, one triplex system or a duo-duplex system are alternatives. The duo-duplex system was chosen because it is easier to realize and can be contained in two different housings. Each sensor duplex unit contains two diverse angle sensor elements; thus, four sensor signals are used to identify the driver's brake command and to supervise the brake pedal module. Electronic hardware or sensor failures are detected within a duplex unit with a redundant voter, which compares two application-specific integrated circuit (ASIC) outputs (including their two sensor signals). If the outputs differ too much, a fault is assumed and one duplex unit is switched off (fails silent).

A further task of the central controller module is to detect and locate pedal sensor failures for the case that the ASICs in the pedal unit and two buses work correctly. Here, a model-based fault detection with parity equations is implemented. Fig. 20 shows a brake pedal sensor configuration with three different pedal sensors and three residuals. The pedal sensors may be diverse (e.g., one angle sensor, one travel sensor, and one force sensor). The residuals are the difference between one sensor signal value and the reconstructed value from another sensor through an analytical pedal model. If using four brake pedal sensors, this parity equation is expanded to calculate six residuals [15].

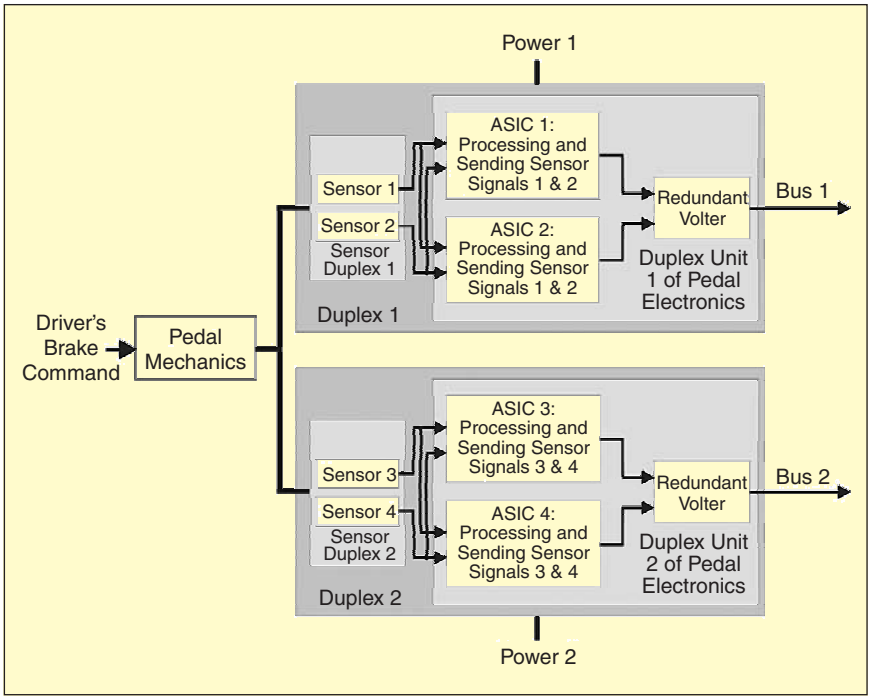


Figure 19. *Duo-duplex architecture for the pedal module.*

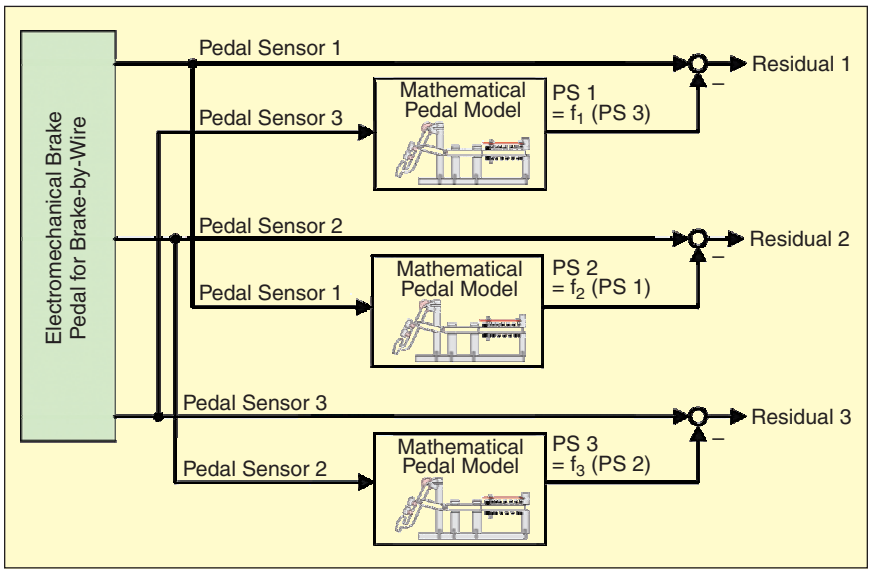


Figure 20. *Sensor fault detection with parity equations for three different sensor signals.*

Conclusion and Outlook

Reliability and safety are of major importance to the introduction of drive-by-wire systems. Their required high safety integrity necessitates that all electronic and electromechanical components, units, and subsystems be fault tolerant with regard to failures in electronic hardware, software, and electrical and mechanical parts. Fault-tolerant properties can be obtained primarily by static or dynamic redundancy, the latter with cold or hot standby, leading to systems that are fail-operational for at least one failure. Fault detection is a basic issue for fault-tolerant systems with redundant modules. However, at present only computationally simple and reliable methods can be used, for reasons of software reliability and testability and the limitations of small microcontrollers. Compared to static redundancy, the use of fault-detection methods for dynamic redundancy saves at least one module.

An engineering challenge is to design mass-producible fault-tolerant sensors, actuators, microcomputers, and bus communication systems with hard-real-time requirements for a reasonable cost. Especially attractive are components with built-in redundancy for mass production. For several years, throttle-by-wire, shift-by-wire, and electronic driver-assisting systems have proven to be highly reliable and safe. Drive-by-wire systems with higher hazard severity for failures are currently being developed. Electrohydraulic brake systems came on the market in 2001, and it is expected that electromechanical brakes will follow in about six to eight years. Steer-by-wire will take longer because of the higher hazard severity and the reduced inherent fault-tolerance possibilities through two front wheels (instead of four wheels, as for braking). The development of drive-by-wire systems will therefore proceed in steps to achieve highly reliable and fault-tolerant sensors, microcomputers, and electromechanical components or, in other words, to build extremely safe mechatronic systems.

Acknowledgments

We appreciate the cooperation of Continental Teves, Frankfurt, during the development of the brake-by-wire system. We are also indebted to the support of research projects on fault diagnosis of actuators, hydraulic brakes, and vehicles through Deutsche Forschungsgemeinschaft (DFG) and Deutsche Forschungsgesellschaft für die Anwendung der Mikroelektronik e.V. (DFAM).

References

[1] H.-M. Streib and H. Bischof, "Electronic throttle control (ETC): A cost effective system for improved emissions, fuel economy, and driveability," *SAE Technical Paper Series*, no. 960338, in R.K. Jurgen, Ed., "Electronic engine control technologies," Warrendale, PA, Society of Automotive Engineers, PT-73, pp. 165-172, 1998.

[2] R.K. Jurgen, Ed., *Electronic Engine Control Technologies*. Warrendale, PA: Soc. Automotive Engineers, SAE PT-73, 1998.

[3] A.T. van Zanten, R. Erhardt, and G. Pfaff, "VDC, the vehicle dynamics control system of Bosch," *SAE Technical Paper Series*, no. 950759, in R.K. Jurgen, Ed., "Electronic braking, traction, and stability control," Warrendale, PA: Soc. Automotive Engineers, SAE: PT-76, pp. 395-412, 1999.

[4] R.K. Jurgen, Ed., "Electronic braking, traction, and stability control," Society of Automotive Engineers, Warrendale, PA, SAE: PT-76, 1999.

[5] B. Connor, "Electric power assisted steering—An alternative to hydraulic and electro-hydraulic systems," *Automobiltechnische Zeitschrift*, vol. 98, nos. 7/8, p. 407, 1996.

[6] S. Germann and R. Isermann, "Nonlinear distance and cruise control for passenger cars," in *First IFAC-Workshop Advances in Automotive Control*, Ascona, Switzerland, 1995, pp. 203-208.

[7] H. Fritz, "Model-based neural speed control of vehicles" (in German), *Automatisierungstechnik*, vol. 44, no. 5, pp. 252-257, 1996.

[8] E.D. Dickmanns, "Road vehicle eyes for high precision navigation," in *High Precision Navigation*, Linkwitz, Ed. Bonn, Germany: Dümmler Verlag, 1995, pp. 329-336.

[9] W.B. Stevens, "The automated highway system program: A progress report," in *Proc. 13th World Congr. IFAC*, San Francisco, CA, 1996, pp. 25-33.

[10] M. Tomizuka, "Automated highway systems—An intelligent transportation system for the next century," in *Proc. IEEE/ASME Int. Conf. Advanced Intelligent Mechatronics '97*, Tokyo, Japan, 1997, p. 1.

[11] P. Rieth, "Electronic driver assistance" (in German), in *VDA-Technischer Kongress,* Frankfurt, Germany, 1999, pp. 119-136.

[12] *Functional Safety of Electrical/Electronic/Programmable Electronic Systems, Version 4.0, Part 1-7,* Standard IEC 61508, 1997.

[13] N. Storey, *Safety-Critical Computer Systems*. Essex, U.K.: Addison Wesley Longman Ltd., 1996.

[14] M.J. Schneider, "Use of a hazard and operability study for evaluation of ABS control logic," *SAE Technical Paper Series*, no. 970815, in *Electronic Braking, Traction, and Stability Control*, R.K. Jurgen, Ed. Warrendale, PA: Society of Automotive Engineers, SAE: PT-76, 1988, pp. 65-93

[15] S. Stölzl, "Fault-tolerant pedal unit of an electromechanical brake system" (in German), Fortschr.-Ber. VDI-Verlag, Düsseldorf, Germany, Reihe 12, no. 426, 2000.

[16] Prometheus, "Report on recommended practice of safety and reliability engineering of future automotive system," Eureka Prometheus European Project, Germany, 1998.

[17] N. Leveson, *Safeware. System Safety and Computer.* Reading, MA: Addison-Wesley, 1995.

[18] W.-D. Jonner, H. Winner, L. Dreilich, and E. Schunck, "Electrohydraulic brake system—The first approach to brake-by-wire technology," *SAE Technical Paper Series*, no. 960991, in *Electronic Braking, Traction, and Stability Control*, R.K. Jurgen, Ed. Warrendale, PA: Society of Automotive Engineers, SAE: PT-76, pp. 221-228, 1999.
[19] H.E. Rauch, "Autonomous control reconfiguration," *IEEE Control Syst. Mag.*, vol. 15, no. 6, pp. 37-48, 1995.
[20] P.R. Chandler, "Reconfigurable flight control at Wright Laboratory," Neuilly-sur-Seine, France, vol. III, AGARD Advisory Rep. 360, Aerospace 2020, 1997.
[21] R.J. Patton, "Fault-tolerant control: The 1997 situation," in *Proc. IFAC Symp. Fault Detection, Supervision and Safety for Technical Processes (SAFEPROCESS)*, vol. 2. Kingston Upon Hull, U.K.: Elsevier, 1997, pp. 1033-1055.
[22] J. Chen, R.J. Patton, and Z. Chen, "Active fault-tolerant flight control systems design using the linear matrix inequality method," *Trans. Inst. Meas. Control*, vol. 21, no. 2/3, pp. 77-84, 1999.
[23] M. Blanke, R. Izadi-Zamanabadhi, S.A. Bogh, and C.P. Lunau, "Fault-tolerant control systems—A holistic view," *Control Eng. Practice*, vol. 5, no. 5, pp. 693-702, 1997.
[24] P. Ballé, M. Fischer, D. Füssel, O. Nelles, and R. Isermann, "Integrated control, diagnosis and reconfiguration of a heat exchanger," *IEEE Control Syst. Mag.*, vol. 18, no. 3, pp. 52-63, 1998.
[25] S. Suryanaryanan and M. Tomizuka, "Fault-tolerant lateral control of automated vehicles based on simultaneous stabilization," in *1st IFAC Conf. Mechatronic Systems*, Darmstadt, Germany, 2000, pp. 899-923.
[26] R. Isermann, Ed."Tutorial Workshop IFAC Congress 1996," *IFAC J. Control Eng. Practice,* Special Section on Supervision, Fault Detection and Diagnosis of Technical Processes, vol. 5, no. 5, pp. 637-719, 1996.
[27] J.J. Gertler, *Fault Detection and Diagnosis on Engineering Systems*. New York: Marcel Dekker, 1999.
[28] J. Chen and R.J. Patton, *Robust Model-Based Fault Diagnosis for Dynamic Systems*. Boston, MA: Kluwer, 1999.
[29] R. Isermann, *Supervision and Fault Diagnosis* (in German). Düsseldorf, Germany: Springer/VDI-Verlag, 1994.
[30] R. Isermann, "Integration of fault detection and diagnosis methods," in *IFAC Symp. Fault Detection, Supervision and Safety for Technical Processes (SAFEPROCESS)*, Espoo, Finland, 1994, pp. 597-612.
[31] T. Pfeufer, "Application of model-based fault detection and diagnosis to the quality assurance of an automotive actuator," *Control Eng. Practice*, vol. 5, no. 5, pp. 703-708, 1997.
[32] O. Moseler, T. Heller, and R. Isermann, "Model-based fault detection for an actuator driven by a brushless dc motor," in *Proc. 14th IFAC World Congr.,* vol. P. Beijing, China: 1999, pp. 193-198.
[33] M.P. Henry and D.W. Clarke, "The self-validating sensor: Rationale, definitions, and examples," *Control Eng. Practice*, vol. 1, no. 2, pp. 585-610, 1993.
[34] D.W. Clarke, "Sensor, actuator, and loop validation," *IEE Control Syst.*, vol. 15, pp. 39-45, Aug. 1995.
[35] A.T. van Zanten, R. Erhardt, K. Landesfeind, and G. Pfaff, "VDC systems development and perspective," *SAE Technical Paper Series*, no. 980235, in *Electronic Braking, Traction, and Stability Control*, R.K. Jurgen, Ed. Warrendale, PA: Society of Automotive Engineers, SAE: PT: PT-76, pp. 373-394, 1999.
[36] R.N. Clark, "State estimation schemes for instrument fault detection," in *Fault Diagnosis in Dynamic Systems*, R.J. Patton, J. Chen, and R.N. Clark, Eds. New York: Prentice-Hall, 1989.
[37] R. Oehler, A. Schoenhoff, and M. Schreiber, "Online model-based fault detection and diagnosis for a smart aircraft actuator," in *Proc. IFAC Symp. Fault Detection, Supervision and Safety for Technical Processes (SAFEPROCESS)*, vol. 2. Kingston upon Hull, U.K., 1997, pp. 591-596.
[38] A. Krautstrunk and P. Mutschler, "Remedial strategy for a permanent magnet synchronous motor drive," in *Proc. EPE'99*, Lausanne, Switzerland, 1999.
[39] R. Isermann and U. Raab, "Intelligent actuators—Ways to autonomous actuating systems," *Automatica*, vol. 29, no. 5, pp. 1315-1331, 1993.
[40] G. Heiner and T. Thurner, "Time-triggered architecture for safety related distributed real-time systems in transportation systems," in *Fault-Tolerant Computing (FTCD 28)*, Munich, Germany, 1998.
[41] H. Kopetz, *Real-Time Systems*. Boston, MA: Kluwer, 1997.
[42] "X-by-wire, safety related fault-tolerant systems in vehicles," Final report of EU Project Brite-EuRam III, 3.B.5, 3.B.6, Prog. no. BE 95/1329, Contract BRPR-CT95-0032, 1998.
[43] S. Poledna and G. Kroiss, "TTD: Towards drive-by-wire," *Elektonik*, no. 14, pp. 36-43, 1999.
[44] T. Pfeufer, "Model-based fault detection and diagnosis for an automotive actuator" (in German), Düsseldorf, Germany, VDI-Verlag, Fortschr.-Ber. VDI Reihe 8, no. 764, 1999.
[45] R. Schwarz, R. Isermann, J. Böhm, J. Nell, and P. Rieth, "Modeling and control of an electro-mechanical disk brake," *SAE Technical Paper Series*, vol. SP-1339, no. 980600 pp. 177-189, 1998.
[46] R. Schwarz, "Rekonstruktion der Bremskraft bei Fahrzeugen mit elektromechanisch betätigten Radbremsen," Düsseldorf, Germany, VDI-Verlag, Fortschr.-Ber. VDI Reihe 12, no. 393, 1999.
[47] S. Stölzl, R. Schwarz, R. Isermann, J. Böhm, J. Nell, and P. Rieth, "Control and supervision of an electromechanical brake system," in *FISITA World Automotive Congr., 2nd Century of the Automobile*, Paris, France, 1998, pp. 154-155 .
[48] J. Balz, K. Bill, J. Böhm, P. Scheerer, and M. Semsch, "Advanced brake system with highest flexibility," *Automobiltechnische Zeitschrift*, vol. 98, no. 6, pp. 328-333, 1996.

Rolf Isermann received the Dipl.-Ing degree in mechanical engineering in 1962 from the University of Stuttgart and the Dr.-Ing. degree in 1965. In 1968, he became "Privatdozent" for automatic control at the University of Stuttgart and, since 1972, has been a Professor in Control Engineering. Since 1977 he has been a Professor at the Darmstadt University of Technology and head of the Laboratory of Control Engineering and Process Automation at the Institute of Automatic Control. He received the Dr. h.c. (honoris causa) from L'Université Libre de Bruxelles and from the Polytechnic University in Bucharest. In 1996, he was awarded the VDE-Ehrenring, the highest scientific award from Verband der Elektrotechnik und Informationstechnik. His current research concentrates in the fields of identification with neural networks, nonlinear digital control, intelligent control, and model based methods of process fault diagnosis with applications for hydraulic and pneumatic servo systems, combustion engines, automobiles, and mechatronic systems.

Ralf Schwarz received the Dipl.-Ing. degree in electrical engineering in 1994. He received the Dr.-Ing. degree in electronic mechanical brakes in 1998 from the Darmstadt University of Technology. In 1999, he worked in the ABS/ESP development for electromechanical brakes at Continental Teves AG, Frankfurt. Since 2000, he has been responsible for global chassis control development.

Stefan Stölzl studied electrical engineering at the University of Karlsruhe to obtain the Dipl.-Ing. degree in 1994. From 1995 to 2000, he was with Continental Teves AG and was responsible for program management for ABS and ESP. Then he joined the Darmstadt University of Technology to perform research on a fault-tolerant pedal unit for brake-by-wire systems. After obtaining the Dr.-Ing. degree, he returned to Continental Theves AG. Since 2001, he has been a manager in the global automotive praxis at A.T. Kearney GmbH.